HydroFunctions Documentation

Release 0.2.4

Martin Roberge

Jun 14, 2023

CONTENTS

CHAPTER

ONE

INTRODUCTION

Hydrofunctions is an open source, free-to-use Python package containing tools for downloading, plotting, analyzing, and storing hydrology data. It is designed to be simple enough for Python beginners, yet powerful enough for scientific research.

Hydrofunctions accesses the the US Geological Survey's National Water Information System (NWIS), which includes data from thousands of active sensors around the US and Territories. The internal design preserves important metadata while facilitating rapid analysis. Detailed error messages help users figure out why a request went wrong. Sensible defaults make it possible to get started right away while still maintaining your access to all of the NWIS's sophisticated functionality.

Hydrofunctions is typically used with Jupyter Notebooks, and it incorporates the functionality of Pandas, Numpy, and matplotlib.

Please give it a try!

1.1 Features

- Retrieves stream data from the USGS NWIS service
- Select data using multiple site numbers, by state, county codes, or a boundary box
- Preserves NWIS metadata, including NoData values
- Helpful error messages to help you write valid requests
- Extracts data into a Pandas dataframe, json, or dict
- **Plot beautiful graphs in Jupyter Notebooks**
  - hydrographs (or time series of any data)
  - flow duration charts
  - cycle plots to illustrate annual or diurnal cycles
  - Interactive map for finding stream gauge ID numbers
- Plotting and manipulation through Pandas dataframes
- Retrieve peak discharges, rating curves, field notes, and statistics for sites
- Saves data in compact, easy-to-use parquet files instead of requesting the same dataset repeatedly

1.2 Quick start

Hydrofunctions can be used from the Python command line, but its graphing capabilities require the use of Jupyter, which is easily downloaded and installed from Anaconda.org.

Before we use Hydrofunctions, we must download and install it on our local computer. You can do this from your computer's command line prompt, which you can access from your recently installed Anaconda distribution:

```
(base) C:\> pip install hydrofunctions
```

Pip will make sure that you have everything that you need on your computer, and load everything you don't have.

Now, open a new Jupyter notebook. You can find an icon for Jupyter within the Anaconda Navigator application.

In the first cell, import hydrofunctions for use on your page:

```
>>> import hydrofunctions as hf
```

To enable chart plotting in Jupyter, add:

```
>>> %matplotlib inline
```

Next, request ten days of data from the USGS National Water Information System (NWIS):

```
>>> site = '01570500'
>>> harrisburg = hf.NWIS(site, 'iv', period='P10D')
Requested data from https://waterservices.usgs.gov/nwis/iv/?format=json%2C1.1&
↪sites=01570500&period=P10D
```

Use the 'ok' attribute to check that the transfer went okay:

```
>>> harrisburg.ok
True
```

Find out about what we collected:

```
>>> harrisburg
USGS:01570500: Susquehanna River at Harrisburg, PA
    00045: <30 * Minutes> Precipitation, total, inches
    00060: <30 * Minutes> Discharge, cubic feet per second
    00065: <30 * Minutes> Gage height, feet
Start: 2019-04-06 00:30:00+00:00
End:   2019-04-15 23:00:00+00:00
```

This listing reports the site ID and name for the site we requested, followed by a list of all of the parameters collected at this site. For each parameter, it lists the parameter code, how frequently the data are collected for this parameter, and the name of the parameter written out with units. The start and end times of the dataset are given in Universal Time (UTC).

You can output the data as a JSON dict using the *.json* attribute, or as a Pandas dataframe by using the .df() method. The .df() method takes parameters to limit what data will go into the dataframe.

View the first five rows of a dataframe that only contains the discharge data:

```
>>> harrisburg.df('discharge').head()
```

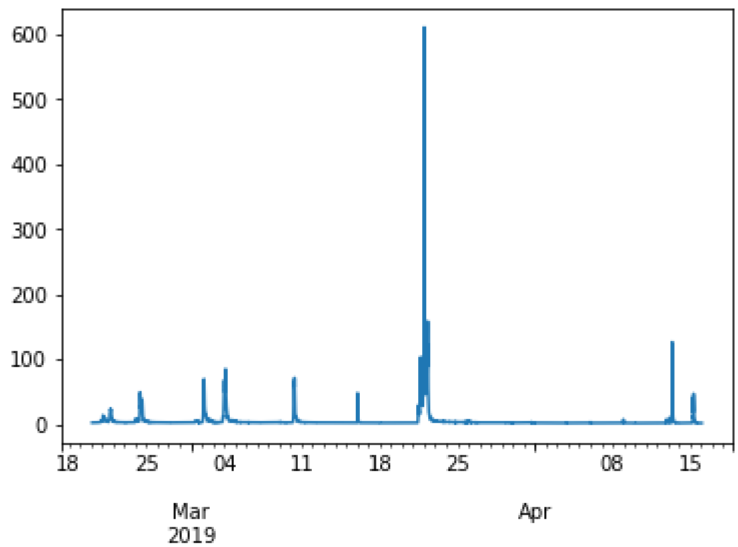
Our data appears in a table…

datetimeUTC	USGS:01570500:00060:00000
2019-04-06 00:30:00+00:00	44200.0
2019-04-06 01:00:00+00:00	44000.0
2019-04-06 01:30:00+00:00	44000.0
2019-04-06 02:00:00+00:00	43700.0
2019-04-06 02:30:00+00:00	43700.0

Because the .df() method returns a dataframe, you have access to all of the methods associated with Pandas, including .plot(), .describe(), and .info() !

Plot the data using Pandas and mathplotlib:

```
>>> harrisburg.df('q').plot()
```



As long as you had *%matplotlib inline* enabled earlier, you will get a graph.

To learn more about hydrofunctions, try using:

```
>>> help(hf)
```

and:

```
>>> dir(hf.NWIS)
```

to list all of the methods available.

CHAPTER

TWO

INSTALLATION

2.1 Easy install

To install HydroFunctions, run this command in your terminal:

```
$ pip install hydrofunctions
```

This is the preferred method to install HydroFunctions, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

2.2 Why use Anaconda?

Anaconda is a free, open-source Python distribution. It bundles important scientific software like Jupyter notebooks, VS Code, and the Spyder IDE all into one pre-configured, pre-compiled download, along with a huge number of scientific libraries (packages). It also includes a tool, *conda*, which manages and updates these packages. Conda also creates 'environments', which are isolated installations of Python, just in case your ArcGIS software (for example) uses a different version of Python than some other piece of software.

If you want to install Jupyter notebooks at the same time that you install hydrofunctions, or if you want to use Python 2 and 3 at the same time, or if you had problems when you tried the Easy Install instructions above. . . then Download and install Anaconda.

Choose the version of Anaconda that is right for your operating system. I advise using the latest version of Python 3.

Once you have Anaconda installed, much of the following will be done from the command line. Anaconda will install a shortcut called 'Anaconda Prompt'. Use this.

2.3 Easy Anaconda install

For people who are using Anaconda because they just wanted an easy way to get all of the software installed at once.

1. From the 'Anaconda Prompt', install hydrofunctions with Pip.

```
$ pip install hydrofunctions
```

That's it!

2.4 Safe Anaconda install for people having problems

For people who like to write their own code, or have Python already installed for something else, or who had difficulties.

1. From the command prompt, create a new environment called 'myenv' with packages for Python 3.9 and jupyter notebooks:

```
> conda create -n myenv python=3.7 jupyternb nb_conda
```

2. List all of the environments that you have available:

```
> conda info -e
```

3. The active environment will have a star next to it. To activate a different environment, such as the one you just created, type:

```
> activate name_of_environment
```

4. To test that you have the correct version of python:

```
> python --version
```

5. install hydrofunctions using the pip tool:

```
> pip install hydrofunctions
```

2.5 Getting started once everything is installed

You now have several ways to run Python. If you installed Anaconda, then you will have icons for the Anaconda Navigator, which you can use to launch your applications, and you may have additional icons for launching Spyder, Jupyter, and other bundled software directly. If you would rather just launch python directly from the command line, just open a command prompt and type the following:

- for a command line interface: *python*
- for an enhanced command line: *ipython*
- to use the Spyder IDE: *spyder*
- or to use Jupyter Notebook: *jupyter lab*

To use hydrofunctions in your python code, make sure that one of your first lines says this:

```
> import hydrofunctions as hf
```

2.6 Installations for people wanting to contribute code to Hydrofunctions

You are a brave and special person indeed. WE SALUTE YOU:

==> Follow the directions in the Contributors Guide

CHAPTER

THREE

WRITING VALID REQUESTS FOR NWIS

The USGS National Water Information System (NWIS) is capable of handling a wide range of requests. A few features in Hydrofunctions are set up to help you write a successful request.

[1]:
```
# First, import hydrofunctions.
import hydrofunctions as hf
```

3.1 What can we specify?

The NWIS can handle data requests that specify:

- Where: we need to specify which stations we are interested in.
- Service: the NWIS provides daily averages ('dv') and 'instantaneous values' ('iv')
- When: we can specify a range of dates, a period of time before now, or just get the most recent observation.
- What: we can specify which parameter we want, or just get everything collected at the site.
- the data service we want.

The only **required** element is a station:

[2]:
```
minimum_request = hf.NWIS('01585200')
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01585200
```

Since we only specified the *where*, the NWIS will assume the following elements:

- **Service**: if not specified, provide the daily average value ('dv')
- **When**: if a `start_date` or `period` is not given, then provide the most recent reading.
- **What**: if you don't ask for a specific parameter (`parameterCd`), you will get everything.

Let's see what our request came back with:

[3]:
```
minimum_request
```

[3]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <0 * Minutes>  Discharge, cubic feet per second
Start: 2019-07-22 00:00:00+00:00
End:   2019-07-22 00:00:00+00:00
```

Here's what the data look like in table form:

[4]:
```
minimum_request.df()
```

[4]:
```
                          USGS:01585200:00060:00003_qualifiers  \
datetimeUTC
2019-07-22 00:00:00+00:00                                    P

                           USGS:01585200:00060:00003
datetimeUTC
2019-07-22 00:00:00+00:00                       1.39
```

3.2 Different ways to specify which site you want

You can specify a site four different ways:

- as a number or list of site numbers
- using `stateCd` and a two letter postal code to retrieve every site in the state
- using `countyCd` and a FIPS code to retrieve every site in a county or list of counties
- using `bBox` to retrieve everything inside of a bounding box of latitudes and longitudes.

You are required to set **one** of these parameters, but only one.

All of these parameters are demonstrated in *Selecting Sites*

3.3 Different ways to specify time

You can specify time in three different ways:

- if you specify nothing, you'll get the most recent reading.
- `period` will return up to 999 days of the most recent data: `period='P11D'`
- `start_date` will return all of the data starting at this date: `start_date='2014-12-31'`

If you specify a `start_date`, you can also specify an `end_date`, which is given in the same format.

3.4 What happens when you make a bad request?

The power of the NWIS also makes it easy to make mistakes. So, we've added a series of helpful error messages to let you know when something went wrong, and why it went wrong.

[6]:
```
# For example, let's mistpye one of our parameters that worked so well above:
notSoGoodNWIS = hf.NWIS('01585200', 'xx', period='P200D')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-6-cbd93523cd36> in <module>
      1 # For example, let's mistpye one of our parameters that worked so well above:
----> 2 notSoGoodNWIS = hf.NWIS('01585200', 'xx', period='P200D').get_data()

c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\station.py in __init_
```

(continues on next page)

(continued from previous page)

```
↪_(self, site, service, start_date, end_date, stateCd, countyCd, bBox, parameterCd,␣
↪period, file)
    129                                         bBox=bBox,
    130                                         parameterCd=parameterCd,
--> 131                                         period=period
    132                                         )
    133             try:

c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\hydrofunctions.py in␣
↪get_nwis(site, service, start_date, end_date, stateCd, countyCd, bBox, parameterCd,␣
↪period)
    217     """
    218
--> 219     service = typing.check_NWIS_service(service)
    220
    221     if (parameterCd == 'all'):

c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\typing.py in check_
↪NWIS_service(input)
    114         raise TypeError("The NWIS service type accepts 'dv' for daily values, "
--> 116                         "{}".format(input))
    115                         "or 'iv' for instantaneous values. Actual value: "
    117
    118

TypeError: The NWIS service type accepts 'dv' for daily values, or 'iv' for␣
↪instantaneous values. Actual value: xx
```

Okay, maybe I shouldn't have typed 'xx' for our service.

Some errors get caught by hydrofunctions, but some don't. Sometimes we end up asking NWIS for something that doesn't make sense, or something that it doesn't have, or maybe NWIS isn't available. In this case, hydrofunctions will receive an error message from NWIS and help you figure out what went wrong.

```
[7]: # Let's ask for the impossible: the start date is AFTER the end date:
badRequest = hf.get_nwis('01585200', 'dv', '2017-12-31', '2017-01-01')
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01585200&startDT=2017-12-31&endDT=2017-01-01
```

```
c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\hydrofunctions.py:
↪627: SyntaxWarning: The NWIS returned a code of 400.
400 Bad Request - This often occurs if the URL arguments are inconsistent. For example,␣
↪if you submit a request using a startDT and an endDT with the period argument. An␣
↪accompanying error should describe why the request was bad.
Error message from NWIS: Bad Request

URL used in this request: https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01585200&startDT=2017-12-31&endDT=2017-01-01
  warnings.warn(msg, SyntaxWarning)
```

3.5 Getting help

I probably shouldn't have started with all of the things that go wrong! My point is that we've got ya.

Where can you go to learn how to do things the RIGHT way?

- The User's Guide
- The USGS guide to their waterservices

But we also have a few built-in helpers that you can use right here, right now:

- help() and ? will list the docstring for whatever object you are curious about
- dir() and .<TAB> will tell you about available methods.

[8]:
```
# Use the help() function to see all of the parameters for a function, their default␣
↪values,
# and a short explanation of what it all means. Or you can type ?hf.NWIS to access the␣
↪same information.
help(hf.NWIS)
```

```
Help on class NWIS in module hydrofunctions.station:

class NWIS(Station)
 |  NWIS(site=None, service='dv', start_date=None, end_date=None, stateCd=None,␣
↪countyCd=None, bBox=None, parameterCd='all', period=None, file=None)
 |
 |  A class for working with data from the USGS NWIS service.
 |
 |  description
 |
 |  Args:
 |      site (str or list of strings):
 |          a valid site is '01585200' or ['01585200', '01646502']. Default is
 |          None. If site is not specified, you will need to select sites using
 |          stateCd or countyCd.
 |
 |      service (str):
 |          can either be 'iv' or 'dv' for instantaneous or daily data.
 |          'dv'(default): daily values. Mean value for an entire day.
 |          'iv': instantaneous value measured at this time. Also known
 |          as 'Real-time data'. Can be measured as often as every
 |          five minutes by the USGS. 15 minutes is more typical.
 |
 |      start_date (str):
 |         should take on the form 'yyyy-mm-dd'
 |
 |      end_date (str):
 |          should take on the form 'yyyy-mm-dd'
 |
 |      stateCd (str):
 |          a valid two-letter state postal abbreviation, such as 'MD'. Default
 |          is None. Selects all stations in this state. Because this type of
 |          site selection returns a large number of sites, you should limit
 |          the amount of data requested for each site.
```

(continues on next page)

(continued from previous page)

```
 |
 |      countyCd (str or list of strings):
 |          a valid county FIPS code. Default is None. Requests all stations
 |          within the county or list of counties. See https://en.wikipedia.org/wiki/
↪FIPS_county_code
 |          for an explanation of FIPS codes.
 |
 |      bBox (str, list, or tuple):
 |          a set of coordinates that defines a bounding box.
 |              * Coordinates are in decimal degrees.
 |              * Longitude values are negative (west of the prime meridian).
 |              * Latitude values are positive (north of the equator).
 |              * comma-delimited, no spaces, if provided as a string.
 |              * The order of the boundaries should be: "West,South,East,North"
 |              * Example: "-83.000000,36.500000,-81.000000,38.500000"
 |
 |      parameterCd (str or list of strings):
 |          NWIS parameter code. Usually a five digit code. Default is 'all'.
 |          A valid code can also be given as a list: parameterCd=['00060','00065']
 |          This will request data for this parameter.
 |
 |              * if value is 'all', or no value is submitted, then NWIS will ␣
↪     return every parameter collected at this site. (default option)
 |              * stage: '00065'
 |              * discharge: '00060'
 |              * not all sites collect all parameters!
 |              * See https://nwis.waterdata.usgs.gov/usa/nwis/pmcodes for full list
 |
 |      period (str):
 |          NWIS period code. Default is None.
 |              * Format is "PxxD", where xx is the number of days before ␣
↪ today, with a maximum of 999 days accepted.
 |              * Either use start_date or period, but not both.
 |
 |  Method resolution order:
 |      NWIS
 |      Station
 |      builtins.object
 |
 |  Methods defined here:
 |
 |  __init__(self, site=None, service='dv', start_date=None, end_date=None, stateCd=None,
↪ countyCd=None, bBox=None, parameterCd='all', period=None, file=None)
 |      Initialize self.  See help(type(self)) for accurate signature.
 |
 |  __repr__(self)
 |      Return repr(self).
 |
 |  df(self, *args)
 |      Return a subset of columns from the dataframe.
 |
 |      Args:
```

(continues on next page)

(continued from previous page)

```
|           '': If no args are provided, the entire dataframe will be returned.
|
|           str 'all': the entire dataframe will be returned.
|
|           str 'data': all of the parameters will be returned, with no flags.
|
|           str 'flags': Only the _qualifier flags will be returned. Unless the
↪   flags arg is provided, only data columns will be returned. Visit             https:/
↪/waterdata.usgs.gov/usa/nwis/uv?codes_help#dv_cd1 to see a             more complete
↪listing of possible codes.
|
|           str 'discharge' or 'q': discharge columns ('00060') will be returned.
|
|           str 'stage': Gauge height columns ('00065') will be returned.
|
|           int any five digit number: any matching parameter columns will be returned.
↪'00065' returns stage, for example.
|
|           int any eight to twelve digit number: any matching stations will be returned.
|
|   get_data(self)
|
|   read(self, file)
|
|   save(self, file)
|
|   ----------------------------------------------------------------------
|   Data descriptors inherited from Station:
|
|   __dict__
|       dictionary for instance variables (if defined)
|
|   __weakref__
|       list of weak references to the object (if defined)
|
|   ----------------------------------------------------------------------
|   Data and other attributes inherited from Station:
|
|   station_dict = {}
```

[9]:

```
# Use the dir() function to see what sort of methods you have available to you,
# or type hf.NWIS.<TAB> to see the same list.
dir(hf.NWIS)
```

[9]:

```
['__class__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
```

(continues on next page)

(continued from previous page)

```
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__weakref__',
 'df',
 'get_data',
 'read',
 'save',
 'station_dict']
```

CHAPTER FOUR

SAVING DATA

Once you request data, Hydrofunctions can automatically save the JSON in a compact zip file. The next time that you re-run your request, the data are retrieved automatically from the local file. Using a data cache like this saves on internet traffic, speeds up your code, and prevents spamming the NWIS just because you are making minor changes to your code. As an alternative to zipped JSON, Hydrofunctions also makes it easy to use Parquet, a compact file format for storing large datasets. Parquet is efficient: file sizes are small and can be read quickly. Parquet is great for large datasets, because it is possible to access parts of the file without reading the entire file.

To save your data, simply provide a filename as a parameter to the NWIS object. If you supply a .parquet file extension, Hydrofunctions will save a parquet file; otherwise it will supply a .json.gz extension and save it in that format.

[1]:
```
import hydrofunctions as hf
```

[2]:
```
new = hf.NWIS('01585200', 'dv', start_date='2018-01-01', end_date='2019-01-01', file=
↪'save_example.json.gz')
new
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01585200&startDT=2018-01-01&endDT=2019-01-01
Saving data to save_example.json.gz
```

[2]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2018-01-01 00:00:00+00:00
End:   2019-01-01 00:00:00+00:00
```

4.1 Automatic file reading & writing

The first time that you make the request, hydrofunctions will save the incoming data into a new file, and you will get a message, `Saving data to filename`.

The second time that you make the request, hydrofunctions will read the data from the file instead of requesting it, and you will get a message, `Reading data from filename`.

[3]:
```
new = hf.NWIS('01585200', 'dv', start_date='2018-01-01', end_date='2019-01-01', file=
↪'save_example.json.gz')
new
```

```
Reading data from save_example.json.gz
```

[3]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <Day>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
Start: 2018-01-01 00:00:00+00:00
End:   2019-01-01 00:00:00+00:00
```

In effect, the local file will act as a cache for your data, reducing your network traffic.

4.2 Manual file reading & writing

It is also possible to force hydrofunctions to read or write a file by using the `NWIS.read()` and `NWIS.save()` methods.

[4]:
```
new.save('save_example.parquet')
```

[4]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2018-01-01 00:00:00+00:00
End:   2019-01-01 00:00:00+00:00
```

[5]:
```
new.read('save_example.parquet')
new
```

[5]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2018-01-01 00:00:00+00:00
End:   2019-01-01 00:00:00+00:00
```

CHAPTER

FIVE

SELECTING SITES BY LOCATION

The National Water Information System (NWIS) makes data available for approximately 1.9 Million different locations in the US and Territories. Finding the data you need within this collection can be a challenge!

There are four methods for selecting sites by location:

- Request data for a site or a list of sites
- Request data for all sites in a state
- Request data for all sites in a county or list of counties
- Request data for all sites inside of bounding box

We'll give examples for each of these methods below.

The following examples are requesting sites, but not specifying a time or parameter of interest. When time is not specified, the NWIS will return only the most recent reading for the site- even if that was fifty years ago! If we don't specify a parameter of interest, NWIS will return all of the parameters measured at that site.

[1]:
```
# First things first
import hydrofunctions as hf
```

5.1 Requesting data for a site or a list of sites

Most USGS site names are between 8-11 digits long. You can use the `draw_map()` function to create an interactive map with 8,000 active stream gages from the Gages-II dataset.

[2]:
```
hf.draw_map()
```

[2]:
```
<IPython.core.display.HTML object>
```

5.1.1 Select a single site

[3]:
```
Beetree = hf.NWIS('01581960')
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01581960
```

```
c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\hydrofunctions.py:
↪103: HydroUserWarning: It is not possible to determine the frequency for one of the␣
↪datasets in this request. This dataset will be set to a frequency of 0 minutes
  "0 minutes", exceptions.HydroUserWarning)
```

[4]:
```
Beetree.df()
```

[4]:
```
                          USGS:01581960:00060:00003_qualifiers  \
datetimeUTC
2019-07-22 00:00:00+00:00                                    P

                           USGS:01581960:00060:00003
datetimeUTC
2019-07-22 00:00:00+00:00                       22.9
```

5.1.2 Select a list of sites

[6]:
```
sites = ['01580000', '01585500', '01589330']
Baltimore = hf.NWIS(sites)
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01580000%2C01585500%2C01589330
```

[7]:
```
Baltimore.df()
```

[7]:
```
                          USGS:01580000:00060:00003_qualifiers  \
datetimeUTC
2019-07-22 00:00:00+00:00                                    P

                           USGS:01580000:00060:00003  \
datetimeUTC
2019-07-22 00:00:00+00:00                      253.0

                          USGS:01585500:00060:00003_qualifiers  \
datetimeUTC
2019-07-22 00:00:00+00:00                                    P

                           USGS:01585500:00060:00003  \
datetimeUTC
2019-07-22 00:00:00+00:00                       12.4

                          USGS:01589330:00060:00003_qualifiers  \
datetimeUTC
2019-07-22 00:00:00+00:00                                    P

                           USGS:01589330:00060:00003
datetimeUTC
2019-07-22 00:00:00+00:00                       1.42
```

5.2 Request data by state or territory

Use the two-letter state postal code to retrieve all of the stations inside of a state. You can only request one state at a time. Lists are not accepted.

[8]:
```
# Request data for all stations in Puerto Rico.
puerto_rico = hf.NWIS(stateCd='PR')
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&stateCd=PR
```

[9]:
```
# List the names for all of the sites in PR
puerto_rico
```

[9]:
```
USGS:175708066163000: PIEZOMETER ERMITANO CENTRAL, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175711066143600: PIEZOMETER JBNERR EAST 1 SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175711066143601: PIEZOMETER JBNERR EAST 2 SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175719066085500: PHILPET 13 WELL, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175721066151400: PIEZOMETER JBNERR WEST 1 SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175721066151401: PIEZOMETER JBNERR WEST 2 SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175728066072200: BARRANCA DUG WELL, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175734066233300: ALOMAR OESTE WELL, SANTA ISABEL, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175735066151800: PIEZOMETER C RASA, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175814066102200: JOBOS WELL, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175829066232200: ALOMAR 1 WELL, SANTA ISABEL, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175833066145800: PIEZOMETER A RASA, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175843066244100: JOBITOS BTR WELL, SANTA ISABEL, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175848066170700: PIEZOMETER RASA G SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175855066023100: FID 2 WELL, ARROYO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175855066050500: ALGARROBOS DOMESTIC WELL, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175858066100200: JUA 5 WELL, GUAYAMA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175903066024100: CMOR WELL, ARROYO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175910066155500: PIEZOMETER RASA D, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175934066364800: CONSTANCIA 3 WELL, PONCE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
```

(continues on next page)

(continued from previous page)

```
USGS:175939066121400: PIEZOMETER AGUIRRE HW 8, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175943066224800: PASO SECO 7 WELL, SANTA ISABEL, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175946066102000: PIEZOMETER POZO HONDO HW 14, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175947066130601: PIEZOMETER AGUIRRE HW 5B, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175950066125200: PIEZOMETER AGUIRRE HW 10, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175950066354200: RESTAURADA P8A WELL, PONCE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175955066103000: PIEZOMETER POZO HONDO HW 15, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:175957066123400: PIEZOMETER AGUIRRE HW 13, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180000066125200: PIEZOMETER AGUIRRE HW 4, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180001066122000: PIEZOMETER AGUIRRE HW 3, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180001066122002: PIEZOMETER AGUIRRE HW 3C, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180001066122004: PIEZOMETER AGUIRRE HW 3E, GUAYAMA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180002066132200: PIEZOMETER AGUIRRE HW 1, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180006066123700: PIEZOMETER AGUIRRE HW 7, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180012066125500: PIEZOMETER AGUIRRE HW 11, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180012066125502: PIEZOMETER AGUIRRE HW 11C, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180017066132100: PIEZOMETER AGUIRRE HW 2, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180020066261500: CAB 1 WELL, JUANA DIAZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180048066301800: PIEZOMETER FA-1, JUANA DIAZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180052066471000: MER 3 WELL, GUAYANILLA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180057066311300: PIEZOMETER JAC-6, JUANA DIAZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180104066152300: PIEZOMETER RM 10, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180122066560300: ARENAS 1 WELL, GUANICA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180132067033800: VIV COLONIA AMISTAD WELL, LAJAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180133066503300: PPGIND 4 WELL, YAUCO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180138066323600: COLONES WELL, PONCE, PR
```

(continues on next page)

(continued from previous page)

```
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180140066542600: ARENAS OBSERVATION WELL, SABANA GRANDE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180141066324701: MON WEST WELL, PONCE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180153066302800: PIEZOMETER JAC-1, JUANA DIAZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180156066434000: LVE WELL, PENUELAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180206066135500: PIEZOMETER RM 5, SALINAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180219066222500: BANOS DE COAMO SPRING, COAMO, PR
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
USGS:180358065503700: PIEZOMETER YABUCOA USGS BRACKISH, YABUCOA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180415065513900: YABUCOA 7 WELL, YABUCOA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180444067555900: MONA AIRPORT WELL, MAYAGUEZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180542067084000: CABO ROJO P1 WELL, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180547067073100: PIEZOMETER CABO ROJO 10, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180547067084800: PIEZOMETER CABO ROJO 8, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180557067083100: PIEZOMETER CABO ROJO 5, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180559065280501: VIEQUES A1 WELL, VIEQUES, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180604067085100: PIEZOMETER CABO ROJO 7, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180621065274902: VIEQUES P2B WELL, VIEQUES, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180627067080600: PIEZOMETER CABO ROJO 1, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180628067075801: PIEZOMETER CABO ROJO 2B, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180628067075802: PIEZOMETER CABO ROJO 2C, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180628067084301: PIEZOMETER CABO ROJO 9B, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180643067080400: PIEZOMETER CABO ROJO 3, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180650067073700: PIEZOMETER CABO ROJO 4, CABO ROJO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180850065493700: RIO HUMACAO WELL, HUMACAO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:180908065475000: SQB 3 WELL, HUMACAO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181023066095700: CIDRA SITE, CIDRA, PR
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:181026066100300: BO. RABANAL RAINGAGE AT CIDRA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:181217065453000: PIEZOMETER CA-1, NAGUABO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181232067083700: CIND 4 WELL, MAYAGUEZ, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181301067081900: CE-RUM2 OBSERVATION WELL, MAYAGUEZ CAMPUS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181311066022500: PIEZOMETER CAGUAS-JUNCOS 11, CAGUAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181346066021500: PIEZOMETER CJ 9, CAGUAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181352066025300: PIEZOMETER CAGUAS-JUNCOS 19A, CAGUAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181446066013400: PIEZOMETER CJ 20, CAGUAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181513065554601: PIEZOMETER CJ 3B, JUNCOS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181529065575200: GURABO RAINGAGE AT GURABO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:181539066014500: PIEZOMETER CAGUAS-JUNCOS 15, CAGUAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181540065580300: PIEZOMETER CAGUAS-JUNCOS 18, GURABO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181708066152400: BO. ANONES RAINGAGE NEAR NARANJITO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:181823065401900: PIEZOMETER USGS RF 4, FAJARDO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:181917065382701: PIEZOMETER RIO FAJARDO 12 FAJARDO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182017067143300: RINCON 4 WELL, RINCON, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182018066593200: SAN SEBASTIAN WELL, SAN SEBASTIAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182131065421100: PIEZOMETER RIO PITAHAYA 4 LUQUILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182133066342800: FLORIDA 7 WELL, FLORIDA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182134066544600: BO GUAJATACA RAINGAGE ABV LAGO GUAJATACA AT PR 455
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:182138065431800: PIEZOMETER USGS RS 2, LUQUILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182209066340600: FLORIDA 1 WELL, FLORIDA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182223065455900: PIEZOMETER RIO MAMEYES 2, RIO GRANDE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182224065430300: MW 11 WELL, LUQUILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182234065440000: PIEZOMETER QDA. MATA DE PLATANOS 1, LUQUILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
```

(continues on next page)

(continued from previous page)

```
USGS:182306065491500: PIEZOMETER USGS RE 1, RIO GRANDE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182308066260400: GMAR WELL, VEGA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182330066185700: PAMPANO 2 WELL, VEGA ALTA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182344065490801: PIEZOMETER USGS RE 2A, RIO GRANDE, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182349066032600: PIEZOMETER JARDIN BOTANICO I-13, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182350066063700: RAINGAGE NEAR ALTAMIRA, GUAYNABO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:182406066034700: PIEZOMETER JARDIN BOTANICO III-19 SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182417066042700: PIEZOMETER LAS AMERICAS I-10, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182422067015100: SALTOS 1 WELL, ISABELA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182431066463400: CAMPO ALEGRE 4 WELL, HATILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182435066052700: PIEZOMETER SALUD MENTAL I-5 SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182436066031200: HPAR WELL, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182437066040500: PIEZOMETER PARQUE FUENTES I-7, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182437066040501: PIEZOMETER PARQUE FUENTES II-7, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182441066082600: BUCHANAN PARK WELL, BAYAMON, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182442067091700: AGUADILLA CEMENT NORTH WELL, AGUADILLA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182443066041500: PIEZOMETER MUNOZ MARIN 1A-8, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182443066041502: PIEZOMETER MUNOZ MARIN 1C-8, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182445066043400: PIEZOMETER ALSACIA I-6, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182445066043401: PIEZOMETER ALSACIA II-6 SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182451066080200: PIEZOMETER FORT BUCHANAN 1, BAYAMON, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182506066280200: PIEZOMETER HILL 2, MANATI, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182511066045401: PIEZOMETER LA ESPERANZA II-2, SAN JUAN, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182515065594100: PIEZOMETER CAMPO RICO 1, CAROLINA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182515066194000: PONDEROSA 1 WELL, VEGA ALTA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182526066165001: PIEZOMETER SANTA ROSA 2 DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
```

(continues on next page)

(continued from previous page)

```
USGS:182530066135400: CAMPANILLA NAVY WELL, TOA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182531066075900: PIEZOMETER BLDG 652 GUAYNABO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182541066313500: OJO DE GUILLO SPRING, MANATI, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:182544066341500: CRUCE DAVILA NC-5 WELL, BARCELONETA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182546066271200: COTO SUR 5 WELL, MANATI, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182548066164401: PIEZOMETER MAGUAYO 2 DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182548066265700: COTO SUR 7 WELL, MANATI, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182549066304300: USGS 166 OBSERVATION WELL, MANATI, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182603066333601: FLORIDA AFUERA 2 WELL, BARCELONETA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182614066261500: PALO ALTO 2 WELL, VEGA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182615066235300: ROSARIO 2 WELL, VEGA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182616066364100: ENCANTADO WELL, ARECIBO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182620066163400: PIEZOMETER HIGUILLAR USGS 1, DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182620066163403: PIEZOMETER HIGUILLAR 4 DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182621066343300: LEDERLE PRLA WELL, BARCELONETA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182626066345100: TIBURONES WELL, BARCELONETA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182626067023800: MALDONADO WELL, ISABELA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182637066475900: PALOMA 3 WELL, HATILLO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182639066385200: SANTANA 1 WELL, ARECIBO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182647066201700: SABANA HOYOS 2 WELL, VEGA ALTA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182647066552400: CBGAR WELL, QUEBRADILLAS, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182654066150600: PIEZOMETER 1 TOA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182655066142400: PIEZOMETER MONSERRATE 2, TOA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182657066162700: PIEZOMETER SAN ANTONIO USGS 1, DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182657066162701: PIEZOMETER SAN ANTONIO USGS 3, DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182657066250600: OJO DE AGUA SPRING, VEGA BAJA, PR
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:182710066303700: CANTITO LA LUISA WELL, MANATI, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182712066251700: PIEZOMETER TORTUGUERO 3, VEGA BAJA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182723066511200: ZANJA 4 WELL, CAMUY, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182737066370900: GRIV WELL, ARECIBO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182743067004200: CANAL PRINCIPAL DIVERSION BLW FOREBAY, ISABELA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:182746066170800: DORADO BEACH 7 WELL, DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182756066454700: BAR 1 WELL, ARECIBO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182757066325600: PLAZUELA 2 WELL, BARCELONETA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182804066173500: DORADO AIRPORT WELL, DORADO, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182856066510900: VAMA WELL, CAMUY, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:182925067031100: OTILIO WELL, ISABELA, PR
    72019: <0 * Minutes>  Depth to water level, feet below land surface
USGS:50010500: RIO GUAJATACA AT LARES, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50010600: RIO GUAJATACA ABV LAGO DE GUAJATACA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50010800: LAGO GUAJATACA AT DAMSITE NR QUEBRADILLAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50011000: CANAL DE DERIVACION AT LAGO DE GUAJATACA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50011080: CANAL MOCA BLW FOREBAY, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50011085: CANAL DE MOCA ABV LAGO REGULADOR (E13) ISABELA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50011088: LAGO REGULADOR DE ISABELA NR HWY 112 ISABELA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50011095: CANAL MOCA BLW HWY 112, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50011128: CANAL MOCA ABV AGUADILLA PLANT, AGUADILLA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50011180: CANAL DE AGUADILLA ABV LAGO RAMEY, AGUADILLA PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50011190: CANAL AGUADILLA BLW LAGO RAMEY, AGUADILLA PR
```

(continues on next page)

(continued from previous page)

```
    00065: <0 * Minutes>  Gage height, feet
USGS:50011200: RIO GUAJATACA BLW LAGO GUAJATACA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50011400: RIO GUAJATACA ABV MOUTH NR QUEBRADILLAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50013000: RIO CAMUY NR LARES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50014000: RIO CRIMINALES NR LARES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50014600: RIO CAMUY AT TRES PUEBLOS SINKHOLE, LARES PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50014800: RIO CAMUY NR BAYANEY, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50015700: RIO CAMUY NR HATILLO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50016000: RIO CAMUY NR CAMUY, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50020100: LAGO GARZAS NR ADJUNTAS, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50020500: RIO GRANDE DE ARECIBO NR ADJUNTAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50020550: LAGO ADJUNTAS NR ADJUNTAS, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50021000: RIO PELLEJAS AT CENTRAL PELLEJAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50021030: RIO PELLEJAS ABV CENTRAL PELLEJAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50021050: RIO PELLEJAS BLW CENTRAL PELLEJAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50021500: RIO PELLEJAS NR UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50021700: RIO GRANDE DE ARECIBO ABV UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50022810: RIO VIVI BLW HACIENDA EL PROGRESO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50023000: RIO VIVI NR CENTRAL PELLEJAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
```

(continues on next page)

(continued from previous page)

```
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50023110: LAGO VIVI NR UTUADO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50024400: RIO VIVI NR UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50024950: RIO GRANDE DE ARECIBO BLW UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50025155: RIO SALIENTE AT COABEY NR JAYUYA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50025850: RIO JAUCA AT PASO PALMA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50026025: RIO CAONILLAS AT PASO PALMA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50026100: RIO CAONILLAS NR UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50026140: LAGO CAONILLAS AT DAMSITE NR UTUADO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50026200: RIO CAONILLAS BLW LAGO CAONILLAS TUNNEL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50026400: RIO YUNES AT HWY 140 NR FLORIDA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50027000: RIO LIMON ABV LAGO DOS BOCAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50027100: LAGO DOS BOCAS AT DAMSITE NR UTUADO, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50027200: RIO GRANDE DE ARECIBO BLW LAGO DOS BOCAS DAM, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50027250: RIO GDE. DE ARECIBO BLW LAGO DOS BOCAS NR FLORIDA
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50027600: RIO GRANDE DE ARECIBO NR SAN PEDRO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50027750: RIO GRANDE DE ARECIBO ABV ARECIBO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50027850: RIO TANAMA NR TANAMA, UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50027880: RIO GUAONICA NEAR UTUADO, UTUADO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50027920: RIO TANAMA ABOVE QDA. PALMA, UTUADO, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50028000: RIO TANAMA NR UTUADO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50028400: RIO TANAMA AT CHARCO HONDO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50029000: RIO GRANDE DE ARECIBO AT CENTRAL CAMBALACHE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50030460: RIO OROCOVIS AT OROCOVIS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50031000: RIO GRANDE DE MANATI AT HWY 155 NR MOROVIS, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50031200: RIO GRANDE DE MANATI NR MOROVIS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50031500: RIO SANA MUERTO NR OROCOVIS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50032290: LAGO EL GUINEO AT DAMSITE NR VILLALBA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50032590: LAGO DE MATRULLAS AT DAMSITE NR OROCOVIS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50033500: RIO BAUTA NR DIVISORIA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50034000: RIO BAUTA NR OROCOVIS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50035000: RIO GRANDE DE MANATI AT CIALES, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50035200: RIO GRANDE DE MANATI AT HWY 145 AT CIALES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50035950: RIO CIALITOS AT HWY 649 AT CIALES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50038100: RIO GRANDE DE MANATI AT HWY 2 NR MANATI, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50038300: RIO COROZAL AT COROZAL, PR
```

(continues on next page)

(continued from previous page)

```
    00045: <0 * Minutes>  Precipitation, total, inches
    00065: <0 * Minutes>  Gage height, feet
USGS:50038320: RIO CIBUCO BLW COROZAL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50038360: RIO MAVILLA NR COROZAL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50038600: RIO UNIBON NR MOROVIS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50039500: RIO CIBUCO AT VEGA BAJA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50039600: RIO CIBUCO AT CENTRAL SAN VICENTE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50039990: LAGO CARITE AT GATE TOWER NR CAYEY, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50039995: LAGO CARITE AT SPILLWAY, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50040000: RIO DE LA PLATA AT LAGO CARITE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50043000: RIO DE LA PLATA AT PROYECTO LA PLATA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50043197: RIO USABON AT HWY 162 NR BARRANQUITAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50043800: RIO DE LA PLATA AT COMERIO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50043980: RIO ARROYATA AT BO. NARANJO, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50044100: REPRESA DE COMERIO 2 BLW COMERIO, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50044810: RIO GUADIANA NR GUADIANA, NARANJITO PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50044830: RIO GUADIANA AT GUADIANA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50045000: LAGO LA PLATA AT DAMSITE NR TOA ALTA, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50045010: RIO DE LA PLATA BLW LA PLATA DAMSITE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50045700: RIO LAJAS AT TOA ALTA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:50046000: RIO DE LA PLATA AT HWY 2 NR TOA ALTA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50047535: RIO DE BAYAMON AT ARENAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50047540: RIO SABANA AT VISTA MONTE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50047545: QUEBRADA PRIETA NR CIDRA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50047550: LAGO DE CIDRA AT DAMSITE NR CIDRA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50047560: RIO DE BAYAMON BLW LAGO DE CIDRA DAM, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50047820: RIO DE BAYAMON AT HWY 174 NR BAYAMON, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50047850: RIO DE BAYAMON NR BAYAMON, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50048000: RIO DE BAYAMON AT BAYAMON, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50048565: QUEBRADA SANTA CATALINA NEAR GUAYNABO, PR
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00065: <0 * Minutes>  Gage height, feet
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:50048580: UNNAMED CREEK AT FORT BUCHANAN, GUAYNABO, PR
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:50048680: LAGO LAS CURIAS AT DAMSITE NR RIO PIEDRAS, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50048690: QUEBRADA LAS CURIAS BLW LAS CURIAS OUTFLOW, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50048700: QUEBRADA LAS CURIAS NR RIO PIEDRAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50048750: QUEBRADA LAS CURIAS TRIBUTARY NR CAIMITO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50048770: RIO PIEDRAS AT EL SENORIAL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50048800: RIO PIEDRAS NR RIO PIEDRAS, PR
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50049000: RIO PIEDRAS AT RIO PIEDRAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50049100: RIO PIEDRAS AT HATO REY, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50049300: QUEBRADA JOSEFINA AT PUERTO NUEVO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50049310: QUEBRADA JOSEFINA AT PINERO AVENUE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50049600: QUEBRADA MARGARITA AT QUEBRADA HEIGHTS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50049620: QDA. MARGARITA AT CAPARRA INTER. NR GUAYNABO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50049800: LAGUNA SAN JOSE AT SAN JUAN, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50050330: LAGUNA PINONES NR CAROLINA, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50050900: RIO GRANDE DE LOIZA AT QUEBRADA ARENAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50051150: QUEBRADA BLANCA AT EL JAGUAL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50051180: QUEBRADA SALVATIERRA NR SAN LORENZO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50051310: RIO CAYAGUAS AT CERRO GORDO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50051800: RIO GRANDE DE LOIZA AT HWY 183 SAN LORENZO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50053025: RIO TURABO ABV BORINQUEN, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50053050: RIO TURABO AT BORINQUEN, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
```

(continues on next page)

(continued from previous page)

```
USGS:50055000: RIO GRANDE DE LOIZA AT CAGUAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
    80225: <0 * Minutes>  Bedload sediment discharge, short tons per day
USGS:50055100: RIO CAGUITAS NR AGUAS BUENAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50055170: RIO CAGUITAS NR CAGUAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50055225: RIO CAGUITAS AT VILLA BLANCA AT CAGUAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50055380: RIO BAIROA ABV BAIROA, CAGUAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50055390: RIO BAIROA AT BAIROA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50055650: QUEBRADA CAIMITO NR JUNCOS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50055750: RIO GURABO BLW EL MANGO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50056400: RIO VALENCIANO NR JUNCOS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50056900: QUEBRADA MAMEY NR GURABO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50057000: RIO GURABO AT GURABO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
    80225: <0 * Minutes>  Bedload sediment discharge, short tons per day
USGS:50058350: RIO CANAS AT RIO CANAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50059000: LAGO LOIZA AT DAMSITE NR TRUJILLO ALTO, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
```

(continues on next page)

(continued from previous page)

```
USGS:50059050: RIO GRANDE DE LOIZA BLW LOIZA DAMSITE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50059210: QUEBRADA GRANDE AT BO. DOS BOCAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50061000: RIO GRANDE DE LOIZA AT CAROLINA, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50061300: RIO CANOVANILLAS NR LOIZA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50061800: RIO CANOVANAS NR CAMPO RICO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50062500: RIO HERRERA NR COLONIA DOLORES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50063250: RIO ESPIRITU SANTO ABV EL VERDE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50063300: RIO ESPIRITU SANTO NR EL VERDE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50063500: QUEBRADA TORONJA AT EL VERDE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50063800: RIO ESPIRITU SANTO NR RIO GRANDE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50064200: RIO GRANDE NR EL VERDE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50065500: RIO MAMEYES NR SABANA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50065700: RIO MAMEYES AT HWY 191 AT MAMEYES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50066000: RIO MAMEYES AT MAMEYES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50067000: RIO SABANA AT SABANA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50070500: RIO FAJARDO ABV FAJARDO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50070900: RIO FAJARDO AT PARAISO NR FAJARDO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50071000: RIO FAJARDO NR FAJARDO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
```

(continues on next page)

(continued from previous page)

```
USGS:50071225: LAGO FAJARDO NEAR VAPOR, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50072000: RIO FAJARDO AT FAJARDO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50073400: QUEBRADA PALMA AT DAGUAO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50074950: QUEBRADA GUABA NR NAGUABO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50075000: RIO ICACOS NR NAGUABO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50075500: RIO BLANCO AT FLORIDA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50075550: LAGO ICACOS AT DAMSITE, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50076000: RIO BLANCO NR FLORIDA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50076800: LAGO BLANCO NEAR NAGUABO, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50077000: RIO BLANCO AT RIO BLANCO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50077500: RIO BLANCO BLW LA FE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50081000: RIO HUMACAO AT LAS PIEDRAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50081500: RIO HUMACAO NR HUMACAO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50082000: RIO HUMACAO AT HWY 3 AT HUMACAO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50082800: RIO GUAYANES NR COLONIA LAURA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50083500: RIO GUAYANES NR YABUCOA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50084000: RIO LIMONES NR YABUCOA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50085100: RIO GUAYANES AT CENTRAL ROIG, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50090500: RIO MAUNABO AT LIZAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50091200: RIO MAUNABO NR MAUNABO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50092000: RIO GRANDE DE PATILLAS NR PATILLAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50093000: RIO MARIN NR PATILLAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50093045: LAGO PATILLAS AT DAMSITE NR PATILLAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
```

(continues on next page)

(continued from previous page)

```
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50093050: RIO GRANDE DE PATILLAS AT LAGO PATILLAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50093053: CANAL DE PATILLAS AT FOREBAY, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093075: CANAL DE PATILLAS ABV GUAYAMA FILTRATION PLANT, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093078: CANAL DE PATILLAS BLW GUAYAMA FILTRATION PLANT, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093083: CANAL DE PATILLAS ABV AES INTAKE AT GUAYAMA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093084: AES INTAKE CANAL AT CANAL DE PATILLAS, GUAYAMA, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50093090: CANAL DE PATILLAS ABV EL LEGADO INTAKE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093110: CANAL DE PATILLAS AT INTAKE 113 SALINAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093115: CANAL DE PATILLAS AT INTAKE 123, SALINAS PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50093120: RIO GRANDE DE PATILLAS BLW LAGO PATILLAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50094545: CANAL GUAMANI AT CARITE FOREBAY
    00065: <0 * Minutes>  Gage height, feet
USGS:50095000: CANAL DE GUAMANI OESTE AT HWY 15 GUAYAMA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50095200: RIO GUAMANI AT GUAYAMA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50095500: RIO GUAMANI NR GUAYAMA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50095800: LAGO MELANIA NR GUAYAMA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50098500: QUEBRADA SECA NR GUAYAMA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50099000: QUEBRADA AGUAS VERDES NR SALINAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50100200: RIO LAPA NR RABO DEL BUEY, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50100450: RIO MAJADA AT LA PLENA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50106100: RIO COAMO AT HWY 14 AT COAMO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:50106500: RIO COAMO NR COAMO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50106850: LAGO COAMO NR LOS LLANOS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50108000: RIO DESCALABRADO NR LOS LLANOS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50108500: RIO DESCALABRADO NR SANTA ISABEL, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50109400: RIO CANAS NR PASTILLO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50110200: CANAL DE ACEITUNA AT VILLALBA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50110650: RIO JACAGUAS ABV LAGO GUAYABAL
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50110900: RIO TOA VACA ABV LAGO TOA VACA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50111000: RIO TOA VACA AT HWY 150 NR VILLALBA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50111200: RIO TOA VACA NR VILLALBA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50111210: LAGO TOA VACA AT DAMSITE NR VILLALBA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50111300: LAGO GUAYABAL AT DAMSITE NR JUANA DIAZ, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50111320: CANAL DE JUANA DIAZ AT PIEDRA AGUZADA NR JUANA DIA
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50111330: CANAL DE JUANA DIAZ AT PASO SECO NR SANTA ISABEL
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50111340: CANAL DE JUANA DIAZ AT BO. PENUELAS NR SALINAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50111500: RIO JACAGUAS AT JUANA DIAZ, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50111700: RIO JACAGUAS NR JUANA DIAZ, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50111750: RIO JACAGUAS BLW QUEBRADA GUANABANA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50112100: RIO JACAGUAS AT MANZANILLO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50112500: RIO INABON AT REAL ABAJO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:50112525: LAGO ANA MARIA AT HACIENDA ANA MARIA, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50113800: RIO CERRILLOS ABV LAGO CERRILLOS NR PONCE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50113950: LAGO CERRILLOS AT DAMSITE NR PONCE, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00065: <0 * Minutes>  Gage height, feet
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50114000: RIO CERRILLOS BLW LAGO CERRILLOS NR PONCE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50114390: RIO BUCANA AT HWY 14 BRIDGE NR PONCE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50114700: RIO BUCANA AT PLAYA DE PONCE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50114900: RIO PORTUGUES NR TIBES, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50115000: RIO PORTUGUES NR PONCE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50115230: RIO PORTUGUES BLW PORTUGUES DAMSITE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50115240: RIO PORTUGUES AT PARQUE CEREMONIAL TIBES NR PONCE
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50115900: RIO PORTUGUES AT HWY 14 AT PONCE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50121000: RIO TALLABOA AT PENUELAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50122000: RIO TALLABOA AT HWY 127, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50124000: RIO GUAYANILLA NR GUAYANILLA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50124200: RIO GUAYANILLA NEAR GUAYANILLA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50124500: RIO GUAYANILLA AT GUAYANILLA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50125780: LAGO LUCCHETTI AT DAMSITE NR YAUCO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50125790: RIO YAUCO AT LUCCHETTI DAMSITE, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50125900: RIO DUEY ABV DIVERSION NR YAUCO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50126150: RIO YAUCO ABV DIVERSION MONSERRATE NR YAUCO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50127000: RIO YAUCO AT YAUCO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50128000: RIO YAUCO NR YAUCO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128900: LAGO LOCO AT DAMSITE NR YAUCO, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50128905: CANAL DE RIEGO DE LAJAS BLW LAGO LOCO DAM YAUCO PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128907: LATERAL M5L, CANAL DE RIEGO DE LAJAS SABANA GRANDE
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128920: CANAL DE RIEGO DE LAJAS ABV MAJINAS FILT. PLANT PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128925: CANAL DE RIEGO DE LAJAS BLW MAJINAS FILT. PLANT PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128933: CANAL RIEGO DE LAJAS ABV LAJAS FILTRATION PLANT
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128935: CANAL DE RIEGO DE LAJAS ABV LAJAS FILT PLANT LAJAS
    00065: <0 * Minutes>  Gage height, feet
USGS:50128940: CANAL DE RIEGO DE LAJAS BLW LAJAS FILT PLANT LAJAS
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128945: CANAL DE RIEGO DE LAJAS AT BO. PALMAREJO NR LAJAS
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128948: CANAL DE RIEGO DE LAJAS AT BO. BETANCES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50128950: CANAL DE RIEGO DE LAJAS NR BOQUERON, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129000: RIO LOCO NR YAUCO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129254: RIO LOCO AT LAS LATAS NR LA JOYA NR GUANICA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129300: LAJAS EAST DRAINAGE CANAL NR ENSENADA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129500: RIO LOCO NR GUANICA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129900: LAGUNA CARTAGENA OUTFLOW NR BOQUERON, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50129935: CANO BOQUERON NO.3 AT BOQUERON, CABO ROJO, PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50130320: QUEBRADA MAMEY AT JOYUDA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50133800: RIO DUEY NR ROSARIO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:50136000: RIO ROSARIO AT ROSARIO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50136400: RIO ROSARIO NR HORMIGUEROS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50138000: RIO GUANAJIBO NR HORMIGUEROS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50141000: RIO BLANCO NR ADJUNTAS, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50141100: LAGO YAHUECAS NR ADJUNTAS, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50141500: LAGO GUAYO AT DAMSITE NR CASTANER, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50141600: RIO GUAYO AT MOUTH NR LARES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50142500: LAGO PRIETO NR ADJUNTAS, PR
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50142700: RIO PRIETO NR LARES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50143930: RIO GRANDE DE ANASCO AT BO. GUACIO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50144000: RIO GRANDE DE ANASCO NR SAN SEBASTIAN, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50145000: RIO GRANDE DE ANASCO AT EL ESPINO, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50145395: RIO CASEI ABV HACIENDA CASEI, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50146073: LAGO DAGUEY ABV ANASCO, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    62614: <0 * Minutes>  Lake or reservoir water surface elevation above NGVD 1929, feet
USGS:50147800: RIO CULEBRINAS AT HWY 404 NR MOCA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:50148890: RIO CULEBRINAS AT MARGARITA DAMSITE NR AGUADA, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:50214500: QUEBRADA RESACA NR MONTE RESACA, CULEBRA PR
    00065: <0 * Minutes>  Gage height, feet
USGS:50215000: DRAINGAGE CANAL AT CULEBRA AIRPORT, CULEBRA PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50231000: QDA COFRESI TRIBUTARY NR ISABEL SEGUNDA VIEQUES PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
    00065: <0 * Minutes>  Gage height, feet
USGS:50231500: LAGUNA BAHIA MOSQUITO NO.1 VIEQUES PR
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00045: <0 * Minutes>  Precipitation, total, inches
    00065: <0 * Minutes>  Gage height, feet
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
    63680: <0 * Minutes>  Turbidity, water, unfiltered, monochrome near infra-red LED␣
↪light, 780-900 nm, detection angle 90 +-2.5 degrees, formazin nephelometric units (FNU)
USGS:50232000: QUEBRADA LA MINA NR LA ESPERANZA, VIEQUES, PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50233000: QUEBRADA PILON AT COLONIA PUERTO REAL, VIEQUES PR
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00065: <0 * Minutes>  Gage height, feet
USGS:50999954: QUEBRADA SALVATIERRA RAINGAGE, SAN LORENZO PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999956: QUEBRADA BLANCA RAINGAGE, SAN LORENZO PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999958: PUEBLITO DEL RIO RAINGAGE, LAS PIEDRAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999959: GURABO ABAJO RAINGAGE, GURABO PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999960: QUEBRADA ARENAS RAINGAGE, SAN LORENZO PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999961: LA PLAZA RAINGAGE, CAGUAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999962: CANABONCITO RAINGAGE, AGUAS BUENAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999963: JAGUEYES ABAJO RAINGAGE, AGUAS BUENAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999964: BAIROA ARRIBA RAINGAGE, AGUAS BUENAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999965: VAQUERIA EL MIMO RAINGAGE, CAGUAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999966: BO. BEATRIZ RAINGAGE, CAGUAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999967: BO. MONTONES RAINGAGE, LAS PIEDRAS PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999968: LAS PIEDRAS CONSTRUCTION RAINGAGE, LAS PIEDRAS, PR
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:50999970: BO. APEADERO RAINGAGE NR VILLALBA, PR
    00045: <0 * Minutes>  Precipitation, total, inches
Start: 1948-12-31 00:00:00+00:00
End:   2019-07-23 00:00:00+00:00
```

5.3 Request data by county or list of counties

Use the five digit FIPS code for each county.

[10]:
```
# Mills, Iowa: 19129; Maui, Hawaii: 15009
counties = hf.NWIS(countyCd = ['19129', '15009'])
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪countyCd=19129%2C15009
```

[11]:
```
counties
```

[11]:
```
USGS:06805850: Keg Creek at Epperson Avenue near Glenwood, IA
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06808000: Mule Creek near Malvern, IA
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:06808200: Spring Valley Creek near Tabor, IA
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16400000: Halawa Stream near Halawa, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16401000: PAPALAUA STREAM NEAR PUKOO, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16402000: PULENA STREAM NEAR WAILAU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16403000: WAIAKEAKUA STREAM NEAR WAILAU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16403600: KAPUHI STREAM NR PELEKUNU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16403900: KAWAINUI STREAM NR PELEKUNU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16404000: PELEKUNU STREAM NR PELEKUNU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16404200: PILIPILILAU STREAM NR PELEKUNU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16405000: LANIPUNI STREAM NEAR PELEKUNU, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16405300: Molokai Tunnel at West Portal, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16410000: KEOLEWA STREAM NEAR KALAE, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16411000: WAIALALA SPRINGS NR KALAE, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16411400: Kakaako Gulch near Mauna Loa, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16412000: MOKOMOKO GULCH NR KALAE, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16413000: KAPUNA STREAM NEAR KALAE, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16414000: Kaunakakai Gulch at Kaunakakai, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16414200: Kaunakakai Gulch at altitude 75 feet, Molokai, HI
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16415000: EF Kawela Gulch nr Kamalo, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16415600: Kawela Gulch near Moku, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:16416000: PUNAULA GULCH NR PUKOO, MOLOKAI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16419500: Papio Gulch at Halawa, Molokai, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16500100: Kepuni Gulch near Kahikinui House, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16500800: Kukuiula Gulch near Kipahulu, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16501000: PALIKEA STREAM BL DIV DAM NR KIPAHULU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16501200: Oheo Gulch at dam near Kipahulu, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16502000: HAHALAWE GULCH NEAR KIPAHULU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16503000: KAELEKU FLUME NR KAELEKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16504000: HANA FLUME NR HANA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16506000: MAKAPIPI DITCH NEAR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16506500: WEST MAKAPIPI SPRING NR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16507000: MAKAPIPI STREAM NEAR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16508000: Hanawi Stream near Nahiku, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16509000: HANAWI STREAM BL GOVT RD NR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16510000: Kapaula Gulch near Nahiku, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16511000: KAPAULA GULCH BL GOVT RD NR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16512000: KOOLAU DITCH AT NAHIKU WEIR NR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16513000: Waiaka Stream near Nahiku, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16514000: PAAKEA GULCH NEAR NAHIKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16515000: Waiohue Gulch near Nahiku, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16516000: Kopiliula Stream near Keanae, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16517000: East Wailuaiki Stream near Keanae, Maui, HI
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16518000: West Wailuaiki Stream near Keanae, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16519000: WEST WAILUANUI STREAM NEAR KEANAE, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16520000: East Wailuanui Stream near Keanae, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16521000: Wailuanui Stream near Keanae, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16521300: Waiokamilo Stream at Dam 3 near Keanae, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16522000: TARO PATCH FEEDER DITCH AT KEANAE, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16523000: Koolau Ditch near Keanae, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16524000: HONOMANU STR AT HAIKU-UKA BDRY NR KAILIILI,MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16527000: Honomanu Stream near Keanae, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16531000: KULA DIV FROM HAIPUAENA STR NR OLINDA,MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16531099: HAIPUAENA STREAM AT KULA DIVERSION
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16531100: HAIPUAENA STR AT KULA PL INTK NR OLINDA,MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16532000: HAIPUAENA STR AT HAIKU-UKA BDY NR KAILIILI, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16535000: HAIPUAENA DIV DITCH AT KOLEA GL NR KEANAE, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16536000: HAIPUAENA STR AB SPRECKELS DITCH NR HUELO, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16538000: SPRECKELS DITCH AT HAIPUAENA WEIR NR HUELO, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16541000: KOOLAU DITCH AT HAIPUAENA NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16541500: Manuel Luis Ditch at Puohokamoa Stream, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16542000: E B PUOHOKAMOA STR AT HAIKU-UKA BDRY NR KAILIILI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16543000: M B PUOHOKAMOA STR AT HAIKU-UKA BDRY NR KAILIILI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16544000: W B PUOHOKAMOA STR AT HAIKU-UKA BDRY NR KAILIILI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16545000: PUOHOKAMOA STR AB SPRECKELS DITCH NR HUELO, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16551000: KOOLAU DITCH AT WAHINEPEE NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16552000: SPRECKELS DITCH AT WAHINEPEE NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16552500: MANUEL LUIS DI W OF PUOHOKAMOA STR NR HUELO,MAUI
```

(continues on next page)

(continued from previous page)

```
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16552600: Waikamoi Stream at Puu Luau nr Olinda, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16552800: Waikamoi Str abv Kula PL intake nr Olinda, Maui,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16554000: WAIKAMOI STR AT HAIKU-UKA BDRY NR KAILIILI, MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16554500: E BR WAIKAMOI STR AT HAIKU-UKA BDRY NR KAILIILI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16555000: WAIKAMOI STR AB WAILOA DITCH NR HUELO, MAUI,HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16556000: WAIKAMOI STREAM NEAR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16557000: ALO STREAM NEAR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16561000: CENTER DITCH BELOW KOLEA RESERVOIR NR HUELO,MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16565000: KAAIEA GULCH NEAR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16565500: SPRECKELS DITCH BELOW KAAIEA GULCH NR HUELO,MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16566000: OOPUOLA STREAM NEAR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16567000: OOPUOLA STR AB SPRECKELS DITCH CRSNG NR HUELO,MA
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16570000: Nailiilihaele Stream near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16574000: KAILUA STR AT HAIKU-UKA BDRY NR KAILIILI,MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16574500: KAILUA STREAM NR KAILIILI, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16577000: Kailua Stream near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16582000: NEW HAMAKUA DITCH AT STATION 5 NR HUELO,MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16585000: HOOLAWANUI STREAM NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16586000: HOOLAWALIILII STREAM NEAR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16587000: Honopou Stream near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16588000: Wailoa Ditch at Honopou near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16589000: New Hamakua Ditch at Honopou near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16590000: OLD HAMAKUA DITCH AT HONOPOU NR HUELO, MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16591000: HONOPOU STR AT LOWRIE DITCH SIPHON NR HUELO,MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16592000: Lowrie Ditch at Honopou Gulch near Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:16593000: HONOPOU STR AB HAIKU DITCH NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16594000: Haiku Ditch at Honopou Gulch near Kailua, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16595000: HONOPOU STR BELOW HAIKU DITCH NR HUELO,MAUI,HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16595100: Div. Ditch at intake, Honopou Str nr Huelo,Maui,HI
    00011: <0 * Minutes>  Temperature, water, degrees Fahrenheit
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16595200: Honopou Stream blw. div. ditch nr Huelo, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16596000: NEW HAMAKUA DITCH AT HALEHAKU WEIR NR HUELO,MAUI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16596200: Halehaku Gulch near Kailiili, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16599500: Opana Tunnel near Kailiili, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16602000: KAUHIKOA DITCH AT OPANA WEIR NR HUELO, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16602400: AWALAU GULCH AT KAILIILI, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16604500: Wailuku River at Kepaniwai Park, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16610000: SOUTH WAIEHU STREAM NR WAILUKU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16612000: WAIHEE RIVER NR WAIHEE, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16614000: Waihee Rv abv Waihee Dtch intk nr Waihee, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16617000: LEFT BR MAKAMAKAOLE STREAM NR WAIHEE, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16618000: Kahakuloa Stream near Honokohau, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16620000: Honokohau Stream near Honokohau, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16621000: HONOKOHAU DITCH AT INTAKE NR HONOKOHAU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16623000: Honolua Stream nr Honokohau, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16629000: HONOKOWAI DITCH NEAR LAHAINA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16630000: HONOKOWAI STREAM NR LAHAINA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16633000: Kahoma Development Tunnel nr Lahaina, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16634000: KAHOMA STREAM NR LAHAINA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16636000: Kanaha Stream ab PL Intake nr Lahaina, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16638000: KANAHA STREAM NR LAHAINA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:16638500: Kahoma Stream at Lahaina, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16641000: Kauaula Stream abv ditch diversion, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16643000: Kauaula Ditch nr Lahaina, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16644000: LAUNIUPOKO STREAM NR LAHAINA, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16645000: OLOWALU DITCH NEAR OLOWALU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16646200: Olowalu Stream at Olowalu, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16647000: Ukumehame Gulch nr Olowalu, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16648000: SOUTH SIDE WAIKAPU DITCH NR WAIKAPU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16649000: PALOLO DITCH NR WAIKAPU, MAUI, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16660000: Kulanihakoi Gulch near Kihei, Maui, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:16681000: Hakioawa Gulch at alt. 75 ft, Kahoolawe, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:16682000: Kaulana Gulch at alt. 100 ft, Kahoolawe, HI
    00060: <0 * Minutes>  Discharge, cubic feet per second
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:203721156151601: 255.0 Kepuni Gulch Rain Gage, Maui, HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:204017156031701: 280.1 Oheo Gulch RG at dam near Kipahulu, Maui, HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:204606156270301: 311.3 Kulanihakoi Raingage nr Kihei, Maui, HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:204902156063101: Hanawi 23, Nahiku, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:204909156281401: 6-4928-02 Puunene Airport Shaft (S34), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:204916156083701: 348.5 West Wailuaiki Rain Gage nr Keanae, Maui, HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:204917156364901: Olowalu C.Res. Loi 9 Outflow (OCR6-CO), Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:204918156364902: Olowalu C.Res. Loi 3 Inflow (OCR5-CI), Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:204923156371501: 297.0 Olowalu Raingage, Maui, HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:204934156061901: Hanawi Stream, lower habitat site, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:204952156073501: Kopiliula Stream, middle habitat site, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:204959156072201: Kopiliula Stream, lower hab. site, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
```

(continues on next page)

(continued from previous page)

```
USGS:205008156082901: Wailuanui Str, middle habitat site, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205022156080901: Unnamed ditch, complex inlet, Waikani, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205025156080901: Unnamed ditch, complex outlet, Waikani, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205036156080501: Wailuanui Stream, lower hab. site, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205041156082301: Hamau Str div, complex inlet, Lakini, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205044156080901: Waiokamilo Str div, complex inlet, Wailua, Maui,HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205044156082101: Hamau Str div, NW complex outlet, Lakini, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205102156282501: 6-5128-02 Waikapu Shaft (S16), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205120156312801: Waikapu Stream near alt 820 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205121156321501: Waikapu Stream near alt 1,160 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205123156320501: Waikapu Stream near alt 1,080 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205125156304801: Waikapu Str, US of left bank taro intake, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205126156315501: Waikapu Stream near alt 1,020 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205132156084901: Palauhulu Str div, complex inlet, Keanae, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205140156304501: 6-5130-01 Waikapu 1 (W14), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205154156303801: 6-5130-02 Waikapu 2, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205159156384201: Kauaula Waimana Taro Ditch Outlet(Ka2-CO),Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205201156261001: 6-5226-02 Puunene, Pump 6 (S18), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205204156383301: 6-5238-01 Launiupoko 3, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205206156383901: Kauaula Waimana Taro Ditch Intake(Ka1-CI),Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205224156113101: WAIKAMOI 14, MAUI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205233156111901: Waikamoi Str, low-mid hab site blw spring, Maui,HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205250156324401: Iao Stream near alt 920 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205254156265801: 6-5226-01 Kaluahonu (S19), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205255156322401: Wailuku River near alt 810 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205256156321601: Wailuku River near alt 740 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
```

(continues on next page)

(continued from previous page)

```
USGS:205304156314401: Wailuku River near alt 580 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205305156304401: 6-5330-05 Shaft 33, well 1, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205327156270301: 6-5327-07 Central PP, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205327156351102: 380.0 Puu Kukui Rain Gage at alt 5,771 ft, Maui,HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:205329156305502: 6-5330-09 Mokuhau Pump 2 (W15A), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205405156305401: 6-5430-05 Waiehu Deep Monitor Well, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205419156304401: 6-5430-03 TH-E, Waiehu, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205427156313101: South Waiehu Stream near alt 635 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205433156311501: South Waiehu Stream near alt 555 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205437156310501: 6-5431-01 TH-B, Waiehu, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205439156303401: S Waiehu Str US of Spreckels Dt intake, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205455156302301: Waiehu Stream near alt 190 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205548156143901: Diversion Ditch at outlet, nr Honopou Str, Maui,HI
    00011: <0 * Minutes>  Temperature, water, degrees Fahrenheit
USGS:205549156143601: Diversion 1, loi outlet, Honopou Stream, Maui, HI
    00011: <0 * Minutes>  Temperature, water, degrees Fahrenheit
USGS:205549156143602: Diversion 2, loi outlet, Honopou Stream, Maui, HI
    00011: <0 * Minutes>  Temperature, water, degrees Fahrenheit
USGS:205611156324601: Waihee River near alt 590 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205617156311101: 6-5631-01 TH-A1, Waihee, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205625156323401: Waihee River near alt 500 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205632156322001: Waihee River near alt 400 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205635156313901: Waihee River near alt 260 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205639156310001: Unnamed Dt return flw to Waihee Str,Waihee,Maui,HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205639156310401: Waihee Str div, lw complex SW inlet,Waihee,Maui,HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205642156310801: Waihee Str div, complex inlet, Waihee, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205644156304401: Waihee River near alt 45 ft, Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:205650156311601: 6-5731-04 Kanoa 2, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205651156313201: 6-5631-02 North Waihee 1, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
```

(continues on next page)

(continued from previous page)

```
USGS:205656156311601: 6-5731-02 Kanoa 1, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205705156312401: 6-5731-05 Kanoa TH, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205711156312901: 6-5731-03 Kupaa 1, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205856156400101: 6-5840-01 Alaeloa (W318), Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:205902156380901: 6-5938-04 Kapalua 3, Maui, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210047156362401: Honokohau Taro Ditch Intake (Ho3-CI), Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:210052156362201: Honokohau Taro Ditch Outlet (Ho4-CO), Maui, HI
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:210258156525001: 4-0353-04 Kamalo Makai, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210351156525901: 4-0352-05 Kamalo Mauka (Aukaina), Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210414156562601: 4-0456-18 Kawela (DW2A), Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210414156565601: 4-0456-04 Breadfruit Shaft, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210423156574801: 4-0457-02 Kanoa Shaft (Kawela), Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210425156483001: 4-0448-02 Mapulehu Shaft 2, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210544157014801: 4-0501-09 Forestry 1, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210605157012001: 4-0601-01 Kaunakakai, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210623157035501: 4-0604-05 MECO Dug Well, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210825157004301: 4-0800-01 Kualapuu Deep Monitor Well, Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:210857156010701: 4-0801-02 Kauluwai (DHHL 2), Molokai, HI
    72150: <0 * Minutes>  Groundwater level relative to Mean Sea Level (MSL), feet
USGS:211039157123101: 551.5 Kakaako Rain Gage near Mauna Loa, Molokai,HI
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:410607095430601: Raingage at Keg Cr at Epperson Ave nr Glenwood, IA
    00045: <0 * Minutes>  Precipitation, total, inches
Start: 1913-08-31 00:00:00+00:00
End:   2019-07-22 00:00:00+00:00
```

5.4 Request data using a bounding box

The coordinates for the bounding box should be in decimal degrees, with negative values for Western and Southern hemispheres. Give the coordinates counter clockwise: West, South, East, North

[12]:
```
# Request multiple sites using a bounding box
test = hf.NWIS(bBox=[-105.430, 39.655, -104, 39.863])
test
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&bBox=-105.
↪43%2C39.655%2C-104%2C39.863
```

[12]:
```
USGS:06711565: SOUTH PLATTE RIVER AT ENGLEWOOD, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:06711570: HARVARD GULCH AT COLORADO BLVD. AT DENVER, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06711575: HARVARD GULCH AT HARVARD PARK AT DENVER, CO
    00045: <0 * Minutes>  Precipitation, total, inches
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06711580: HARVARD GULCH TRIBUTARY AT ENGLEWOOD, CO.
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:06711590: SOUTH PLATTE RIVER AT FLORIDA AVE AT DENVER, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06711618: WEIR GULCH UPSTREAM FROM 1ST AVE. AT DENVER, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06711770: DRY GULCH AT DENVER, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06711780: LAKEWOOD GULCH AT DENVER, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06713300: CHERRY CREEK AT GLENDALE, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06713500: CHERRY CREEK AT DENVER, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
USGS:06714000: SOUTH PLATTE RIVER AT DENVER, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06714100: THIRTY-SIXTH STREET STORM SEWER AT DENVER, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06714130: SOUTH PLATTE RIVER AT 50TH AVENUE AT DENVER, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06714210: SOUTH PLATTE RIVER TRIBUTARY AT DENVER, CO.
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:06714215: SOUTH PLATTE R AT 64TH AVE. COMMERCE CITY, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06714360: SAND CRK ABV BURLINGTON DITCH NR COMMERCE CITY, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:06718500: NORTH CLEAR CREEK NEAR BLACK HAWK, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06718550: NORTH CLEAR CREEK ABOVE MOUTH NR BLACK HAWK, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06719000: CLEAR CREEK AT FORKS CREEK, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06719500: CLEAR CREEK NEAR GOLDEN, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06719505: CLEAR CREEK AT GOLDEN, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
    80154: <0 * Minutes>  Suspended sediment concentration, milligrams per liter
    80155: <0 * Minutes>  Suspended sediment discharge, short tons per day
USGS:06719526: CLEAR CREEK AT TABOR STREET, AT WHEATRIDGE, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06719560: LENA GULCH AT LAKEWOOD, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06719725: RALSTON CREEK NEAR PLAINVIEW, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:06719730: SCHWARTZWALDER MINE EFFLUENT NEAR PLAINVIEW, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:06719735: RALSTON C BL SHWARTZWLDER MINE NR PLAINVIEW, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:06719740: RALSTON CREEK AB RALSTON RES, NR PLAINVIEW, CO.
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
    00060: <0 * Minutes>  Discharge, cubic feet per second
    00095: <0 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <0 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <0 * Minutes>  pH, water, unfiltered, field, standard units
USGS:06719770: CLEAR CREEK TRIBUTARY AT ARVADA, CO.
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:06719840: LITTLE DRY CREEK AT WESTMINSTER, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06720000: CLEAR CREEK AT MOUTH NEAR DERBY, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06720255: UVALDA INTERCEPT BL 56TH AV AT ROCKY MTN ARS, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
USGS:06720280: PEORIA INTERCEPT BL 56TH AV, AT ROCKY MTN ARS, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06720285: HAVANA INTERCEPT BL 56TH AV, AT ROCKY MTN ARS, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06720460: FIRST CR BEL BUCKLEY RD, AT ROCKY MTN ARSENAL, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:06758300: KIOWA CREEK AT BENNETT, CO.
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:393938104572101:  HARVARD GUL PRECIP STA AT SLAVENS SC AT DENVER,CO
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:393947104555101: HARVARD GUL PRECIP STA AT BRADLEY SCH AT DENVER,CO
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:394028104560201: HARVARD GUL PRECIP STA AT DENVER ACAD AT DENVER,CO
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:394028104565501:  HARVARD GUL PRECIP STA AT UNIV PK SC AT DENVER CO
    00045: <0 * Minutes>  Precipitation, total, inches
USGS:394329104490101: TOLL GATE CREEK ABOVE 6TH AVE AT AURORA, CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
USGS:394409105020501: LAKEWOOD GULCH ABV KNOX ST AT DENVER, CO
    00010: <0 * Minutes>  Temperature, water, degrees Celsius
USGS:394839104570300: SAND CREEK AT MOUTH NR COMMERCE CITY,CO
    00060: <0 * Minutes>  Discharge, cubic feet per second
Start: 1912-10-31 00:00:00+00:00
End:   2019-07-23 00:00:00+00:00
```

CHAPTER

SIX

VIEWING DATAFRAMES

After you successfully request a dataset from the USGS, Hydrofunctions will process the data into a huge table and make it available to you in several formats.

If you are working with Python and timeseries data, then you should already know about Pandas and Numpy, the numerical systems Hydrofunctions is built upon. These two data analysis packages are involved in almost all scientific data analysis, and are the starting point for hundreds of projects.

Use the following dataset for the examples below:

[1]:
```
import hydrofunctions as hf
%matplotlib inline
data = hf.NWIS(['01650800', '01589330'], 'iv', start_date='2019-05-01', end_date='2019-
↪06-01', file='view-example.parquet')
```

```
Reading data from view-example.parquet
```

This dataset has the following properties:

[2]:
```
data
```

[2]:
```
USGS:01589330: DEAD RUN AT FRANKLINTOWN, MD
    00060: <5 * Minutes>  Discharge, cubic feet per second
    00065: <5 * Minutes>  Gage height, feet
USGS:01650800: SLIGO CREEK NEAR TAKOMA PARK, MD
    00010: <5 * Minutes>  Temperature, water, degrees Celsius
    00060: <5 * Minutes>  Discharge, cubic feet per second
    00065: <5 * Minutes>  Gage height, feet
    00095: <5 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <5 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <5 * Minutes>  pH, water, unfiltered, field, standard units
    63680: <5 * Minutes>  Turbidity, water, unfiltered, monochrome near infra-red LED␣
↪light, 780-900 nm, detection angle 90 +-2.5 degrees, formazin nephelometric units (FNU)
Start: 2019-05-01 04:00:00+00:00
End:   2019-06-02 03:55:00+00:00
```

It includes two sites, with seven different types of data being collected at one site, and two at the other.

6.1 View the entire table

Let's start by viewing all of the columns in the first five rows of our table. To view all of our data as a dataframe, we use the `.df()` method of `NWIS`. The `.head()` method limits display of our table to just the first five rows:

[3]:
```
data.df().head()
```

[3]:
```
                          USGS:01589330:00060:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01589330:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       1.99
2019-05-01 04:05:00+00:00                       1.99
2019-05-01 04:10:00+00:00                       1.99
2019-05-01 04:15:00+00:00                       1.99
2019-05-01 04:20:00+00:00                       1.85

                          USGS:01589330:00065:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01589330:00065:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       0.51
2019-05-01 04:05:00+00:00                       0.51
2019-05-01 04:10:00+00:00                       0.51
2019-05-01 04:15:00+00:00                       0.51
2019-05-01 04:20:00+00:00                       0.50

                          USGS:01650800:00010:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01650800:00010:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       17.9
2019-05-01 04:05:00+00:00                       17.8
2019-05-01 04:10:00+00:00                       17.8
2019-05-01 04:15:00+00:00                       17.8
```

(continues on next page)

(continued from previous page)

```
2019-05-01 04:20:00+00:00                          17.7

                          USGS:01650800:00060:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01650800:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                          8.98
2019-05-01 04:05:00+00:00                          8.65
2019-05-01 04:10:00+00:00                          8.65
2019-05-01 04:15:00+00:00                          8.34
2019-05-01 04:20:00+00:00                          8.03

                          USGS:01650800:00065:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01650800:00065:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                          1.06
2019-05-01 04:05:00+00:00                          1.05
2019-05-01 04:10:00+00:00                          1.05
2019-05-01 04:15:00+00:00                          1.04
2019-05-01 04:20:00+00:00                          1.03

                          USGS:01650800:00095:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                    P
2019-05-01 04:05:00+00:00                                    P
2019-05-01 04:10:00+00:00                                    P
2019-05-01 04:15:00+00:00                                    P
2019-05-01 04:20:00+00:00                                    P

                           USGS:01650800:00095:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                         574.0
2019-05-01 04:05:00+00:00                         580.0
2019-05-01 04:10:00+00:00                         587.0
2019-05-01 04:15:00+00:00                         592.0
2019-05-01 04:20:00+00:00                         597.0

                          USGS:01650800:00300:00000_qualifiers  \
datetimeUTC
```

(continues on next page)

(continued from previous page)

```
2019-05-01 04:00:00+00:00                                        P
2019-05-01 04:05:00+00:00                                        P
2019-05-01 04:10:00+00:00                                        P
2019-05-01 04:15:00+00:00                                        P
2019-05-01 04:20:00+00:00                                        P

                          USGS:01650800:00300:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                        7.6
2019-05-01 04:05:00+00:00                        7.6
2019-05-01 04:10:00+00:00                        7.6
2019-05-01 04:15:00+00:00                        7.6
2019-05-01 04:20:00+00:00                        7.6

                         USGS:01650800:00400:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                        P
2019-05-01 04:05:00+00:00                                        P
2019-05-01 04:10:00+00:00                                        P
2019-05-01 04:15:00+00:00                                        P
2019-05-01 04:20:00+00:00                                        P

                          USGS:01650800:00400:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                        7.3
2019-05-01 04:05:00+00:00                        7.3
2019-05-01 04:10:00+00:00                        7.3
2019-05-01 04:15:00+00:00                        7.3
2019-05-01 04:20:00+00:00                        7.3

                         USGS:01650800:63680:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                        P
2019-05-01 04:05:00+00:00                                        P
2019-05-01 04:10:00+00:00                                        P
2019-05-01 04:15:00+00:00                                        P
2019-05-01 04:20:00+00:00                                        P

                          USGS:01650800:63680:00000
datetimeUTC
2019-05-01 04:00:00+00:00                       10.3
2019-05-01 04:05:00+00:00                        9.7
2019-05-01 04:10:00+00:00                        9.1
2019-05-01 04:15:00+00:00                        8.2
2019-05-01 04:20:00+00:00                        7.9
```

This is equivalent to `data.df('all').head()`

We now have nine different columns containing data. Each column has a twin 'qualifiers' column, which contains metadata flags.

You can list the columns separately by viewing the columns attribute of the dataframe:

[4]:
```
data.df().columns
```

[4]:
```
Index(['USGS:01589330:00060:00000_qualifiers', 'USGS:01589330:00060:00000',
       'USGS:01589330:00065:00000_qualifiers', 'USGS:01589330:00065:00000',
       'USGS:01650800:00010:00000_qualifiers', 'USGS:01650800:00010:00000',
       'USGS:01650800:00060:00000_qualifiers', 'USGS:01650800:00060:00000',
       'USGS:01650800:00065:00000_qualifiers', 'USGS:01650800:00065:00000',
       'USGS:01650800:00095:00000_qualifiers', 'USGS:01650800:00095:00000',
       'USGS:01650800:00300:00000_qualifiers', 'USGS:01650800:00300:00000',
       'USGS:01650800:00400:00000_qualifiers', 'USGS:01650800:00400:00000',
       'USGS:01650800:63680:00000_qualifiers', 'USGS:01650800:63680:00000'],
      dtype='object')
```

6.2 Viewing only the data columns

To omit the 'qualifier' columns from our dataframe, only ask for the nine 'data' columns:

[5]:
```
data.df('data').head()
```

[5]:
```
                           USGS:01589330:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       1.99
2019-05-01 04:05:00+00:00                       1.99
2019-05-01 04:10:00+00:00                       1.99
2019-05-01 04:15:00+00:00                       1.99
2019-05-01 04:20:00+00:00                       1.85

                           USGS:01589330:00065:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       0.51
2019-05-01 04:05:00+00:00                       0.51
2019-05-01 04:10:00+00:00                       0.51
2019-05-01 04:15:00+00:00                       0.51
2019-05-01 04:20:00+00:00                       0.50

                           USGS:01650800:00010:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       17.9
2019-05-01 04:05:00+00:00                       17.8
2019-05-01 04:10:00+00:00                       17.8
2019-05-01 04:15:00+00:00                       17.8
2019-05-01 04:20:00+00:00                       17.7

                           USGS:01650800:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                       8.98
2019-05-01 04:05:00+00:00                       8.65
2019-05-01 04:10:00+00:00                       8.65
2019-05-01 04:15:00+00:00                       8.34
2019-05-01 04:20:00+00:00                       8.03

                           USGS:01650800:00065:00000  \
```

(continues on next page)

(continued from previous page)

```
datetimeUTC
2019-05-01 04:00:00+00:00                              1.06
2019-05-01 04:05:00+00:00                              1.05
2019-05-01 04:10:00+00:00                              1.05
2019-05-01 04:15:00+00:00                              1.04
2019-05-01 04:20:00+00:00                              1.03

                           USGS:01650800:00095:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                             574.0
2019-05-01 04:05:00+00:00                             580.0
2019-05-01 04:10:00+00:00                             587.0
2019-05-01 04:15:00+00:00                             592.0
2019-05-01 04:20:00+00:00                             597.0

                           USGS:01650800:00300:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                               7.6
2019-05-01 04:05:00+00:00                               7.6
2019-05-01 04:10:00+00:00                               7.6
2019-05-01 04:15:00+00:00                               7.6
2019-05-01 04:20:00+00:00                               7.6

                           USGS:01650800:00400:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                               7.3
2019-05-01 04:05:00+00:00                               7.3
2019-05-01 04:10:00+00:00                               7.3
2019-05-01 04:15:00+00:00                               7.3
2019-05-01 04:20:00+00:00                               7.3

                           USGS:01650800:63680:00000
datetimeUTC
2019-05-01 04:00:00+00:00                              10.3
2019-05-01 04:05:00+00:00                               9.7
2019-05-01 04:10:00+00:00                               9.1
2019-05-01 04:15:00+00:00                               8.2
2019-05-01 04:20:00+00:00                               7.9
```

6.3 Viewing only the data columns for one site

You can also limit your table so that it only contains data for one site by specifying your site number '01589330'. If you don't specify 'flags', it is assumed you only want the data columns:

[6]:
```
data.df('01589330').head()
```

[6]:
```
                           USGS:01589330:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                              1.99
2019-05-01 04:05:00+00:00                              1.99
```

(continues on next page)

(continued from previous page)

```
2019-05-01 04:10:00+00:00                             1.99
2019-05-01 04:15:00+00:00                             1.99
2019-05-01 04:20:00+00:00                             1.85

                           USGS:01589330:00065:00000
datetimeUTC
2019-05-01 04:00:00+00:00                             0.51
2019-05-01 04:05:00+00:00                             0.51
2019-05-01 04:10:00+00:00                             0.51
2019-05-01 04:15:00+00:00                             0.51
2019-05-01 04:20:00+00:00                             0.50
```

6.4 Viewing one parameter

It is possible to limit your view to only one parameter by entering the five digit parameter number, such as '00065' for stage. Some common parameters have an alias, such as 'q' and 'discharge' for '00060'. Since discharge is collected at both sites, this request will return two columns:

[7]:
```
data.df('q').head()
```

[7]:
```
                           USGS:01589330:00060:00000  \
datetimeUTC
2019-05-01 04:00:00+00:00                             1.99
2019-05-01 04:05:00+00:00                             1.99
2019-05-01 04:10:00+00:00                             1.99
2019-05-01 04:15:00+00:00                             1.99
2019-05-01 04:20:00+00:00                             1.85

                           USGS:01650800:00060:00000
datetimeUTC
2019-05-01 04:00:00+00:00                             8.98
2019-05-01 04:05:00+00:00                             8.65
2019-05-01 04:10:00+00:00                             8.65
2019-05-01 04:15:00+00:00                             8.34
2019-05-01 04:20:00+00:00                             8.03
```

The previous example selected discharge data at both sites in the dataset, but you can combine your requests in any order to get just the columns you want. For example, the stage data at a single site would be: `.df('01589330', 'stage')`

6.5 Viewing and interpreting the qualifier flags

Every data column also comes matched with a 'qualifier' column that contains a set of metadata flags for each observation. These flags are usually not provided unless you request the full table or specifically request 'flags'.

This request will provide the flags for the two discharge columns we viewed above:

[8]:
```
data.df('q', 'flags').head()
```

```
[8]:                            USGS:01589330:00060:00000_qualifiers  \
datetimeUTC
2019-05-01 04:00:00+00:00                                      P
2019-05-01 04:05:00+00:00                                      P
2019-05-01 04:10:00+00:00                                      P
2019-05-01 04:15:00+00:00                                      P
2019-05-01 04:20:00+00:00                                      P

                            USGS:01650800:00060:00000_qualifiers
datetimeUTC
2019-05-01 04:00:00+00:00                                      P
2019-05-01 04:05:00+00:00                                      P
2019-05-01 04:10:00+00:00                                      P
2019-05-01 04:15:00+00:00                                      P
2019-05-01 04:20:00+00:00                                      P
```

In this case, 'P' flags indicate "Provisional" data, meaning the data is less than a year old, and the USGS has not reviewed it yet and released it as the "Approved" ('A') official data.

Other flags include:

- **'Ice'** for readings that were missed due to sensor icing
- **'hf.upsampled'** for readings that were interpolated between two valid readings by Hydrofunctions due to a mismatch in observation frequency
- **'hf.missing'** for readings that appear to be missing from the record

A more complete listing of qualifier flags can be found here: https://waterdata.usgs.gov/usa/nwis/uv?codes_help#dv_cd1

6.6 Working with dataframes

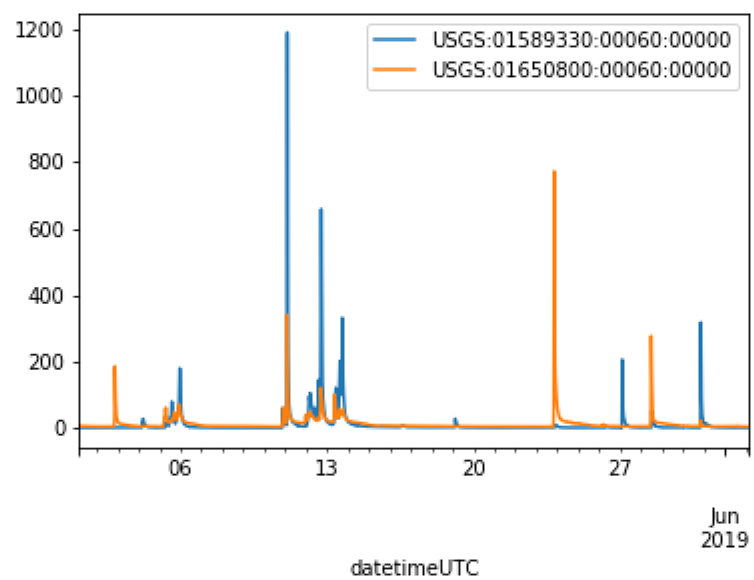
Once you have your dataframe, you have access to all of the power of Pandas. Some of the things you can do are:

plotting data

```
[9]: data.df('discharge').plot()
```

```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

```
[9]: <matplotlib.axes._subplots.AxesSubplot at 0x287dae95208>
```



View descriptive statistics

[10]:
```
data.df('discharge').describe()
```

[10]:

	USGS:01589330:00060:00000	USGS:01650800:00060:00000
count	9216.000000	9216.000000
mean	11.630525	13.615661
std	48.421805	29.256210
min	1.390000	4.850000
25%	1.940000	5.530000
50%	2.410000	6.610000
75%	4.630000	12.800000
max	1190.000000	772.000000

6.7 Error messages

Hydrofunctions will let you know if you request something you don't have.

[11]:
```
data.df('00000000')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-11-ba193784fd2d> in <module>
----> 1 data.df('00000000')

c:\users\marty\google drive\pydev\src\hydrofunctions\hydrofunctions\station.py in␣
↪df(self, *args)
    228                         sites = self._dataframe.columns.str.contains(station_arg)
    229                         if not sites.any():
--> 230                             raise ValueError("The site '{site}' is not in this␣
↪dataset.".format(site=item))
    231                     else:
```

(continues on next page)

(continued from previous page)

```
    232                         raise ValueError("The argument '{item}' is not recognized.".
↪format(item=item))

ValueError: The site '00000000' is not in this dataset.
```

CHAPTER

SEVEN

GRAPHING

Hydrofunctions provides several ways to plot your data in a Jupyter notebook. Many of these methods use the graphing methods built-in to the Pandas dataframe. Some of these methods are specific to Hydrofunctions. All of these techniques use matplotlib under the hood to create the charts, so Hydrofunctions includes it during installation.

7.1 Getting ready

7.1.1 First step: Automatic display of charts in Jupyter

The first step for creating a graph is to import Hydrofunctions, and then provide Jupyter with a directive to automatically display all charts from matplotlib:

```
>>> import hydrofunctions as hf
>>> %matplotlib inline
```

7.1.2 Second step: Preparing our data for plotting

The next step is to request some data from the NWIS for us to plot:

```
>>> request = hf.NWIS('01585200', 'dv', period='P999D')
```

Next, we create a dataframe called 'data' from our request:

```
>>> data = request.df('q')
```

The rest of the examples will assume that we have a dataframe called data.

7.2 Flow duration chart

Flow duration charts are cumulative frequency charts that are used in hydrology to compare stream flow at different stream gauges. They are constructed with daily mean discharge data, and are typically used to analyze the typical baseflow and low flow conditions. They are similar to a flood frequency analysis, in that they both try to illustrate the relationship between the magnitude of the flows and the frequency that these flows occur. Large discharges occur infrequently, while it is common to see a baseflow discharge or higher.

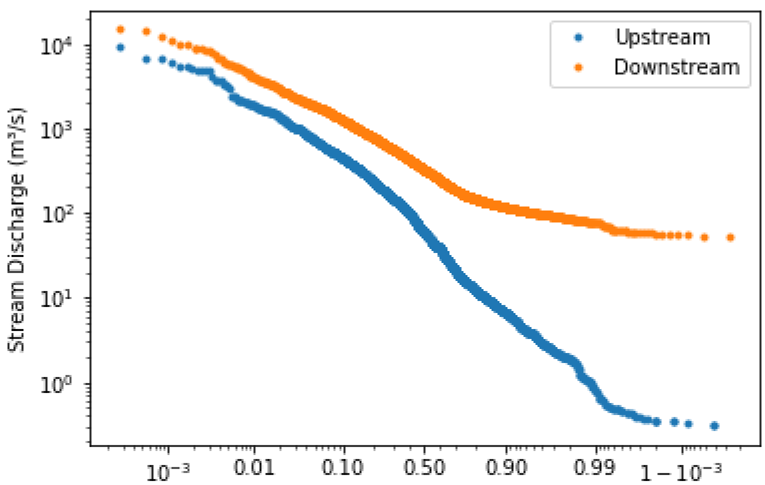
Hydrofunctions includes a Flow Duration Chart, which you access as a function:

```
>>> hf.flow_duration(data)
```

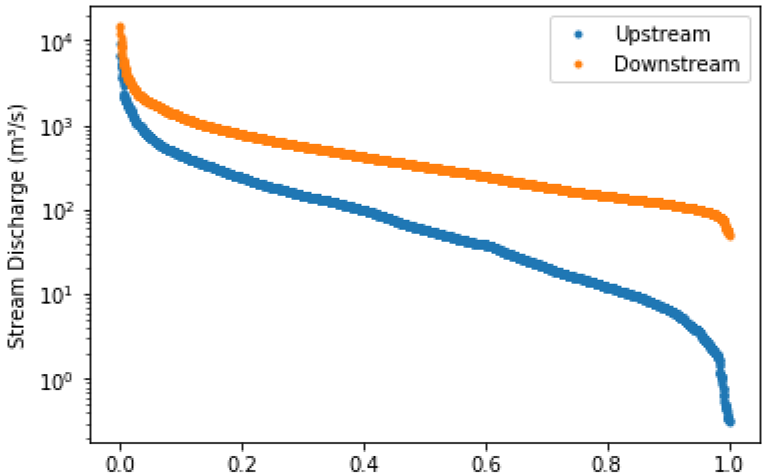
Options include the ability to change the following:

- xscale: default is 'logit'; may also be 'linear'
- yscale: default is 'log'; may also be 'linear'
- ylabel: default is 'Discharge'; may be any string.
- **symbol: default is '.' for small points. May also be:**
  - pixel point: ','
  - up triangle: '^'
  - circle: 'o'
  - plus: '+'

Example of a flow duration chart using all of the defaults ('log' y axis and 'logit' x axis).



This chart compares the discharge at two locations: a smaller upstream site on the Shenandoah River, and a larger downstream site with higher discharges. The y axis uses the default logarithmic scale, because river discharges frequently range over several orders of magnitude. The x axis plots the probability that a discharge of this size or larger will occur on any given day. Note that the graph slopes from the upper left to the lower right in an inverse relationship. This means that the river has a low probability of exceeding the high flows, and a high probability of exceeding the low flows. The line for the larger stream is always above the line for the smaller stream because the larger stream has higher discharges.



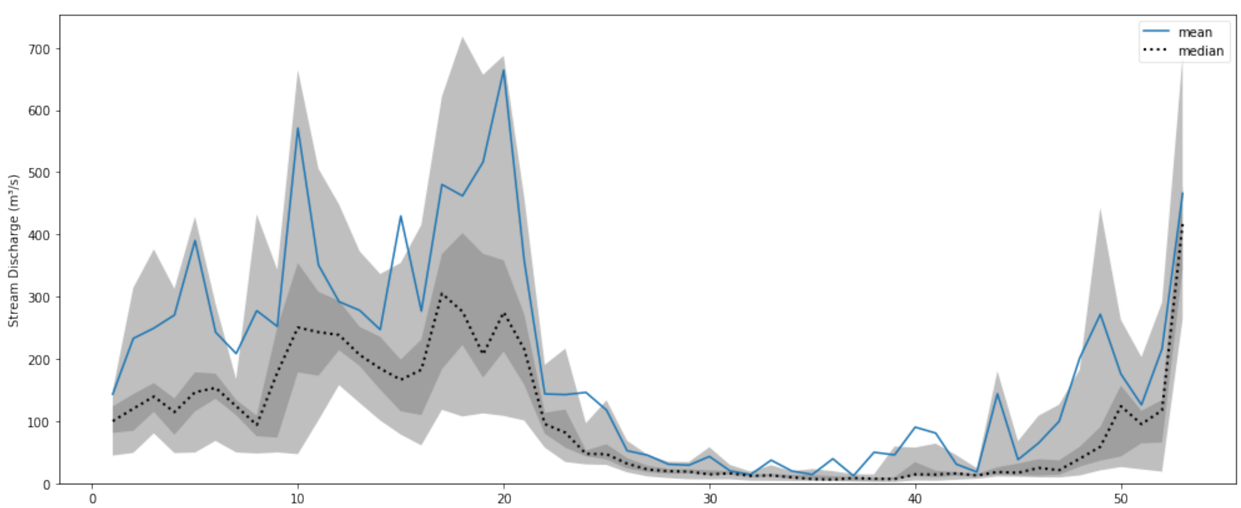
The first graph used a 'logit' scale on the x axis, while this second graph uses a 'linear' x axis to plot the same data. The logit scale will render a normal "bell curve" distribution as almost a straight line in a cumulative graph. In the second chart, the linear x axis uses the same distance between 80 and 90% as between 40 and 50%. In both graphs, the median value is plotted in the center of the x axis, at the 50% mark, since it has a 50% chance of happening on any given day.

7.3 Cycle plot

The cycle plot was inspired by an example created by Jake VanderPlas in his 2016 book, Python data science handbook: essential tools for working with data.

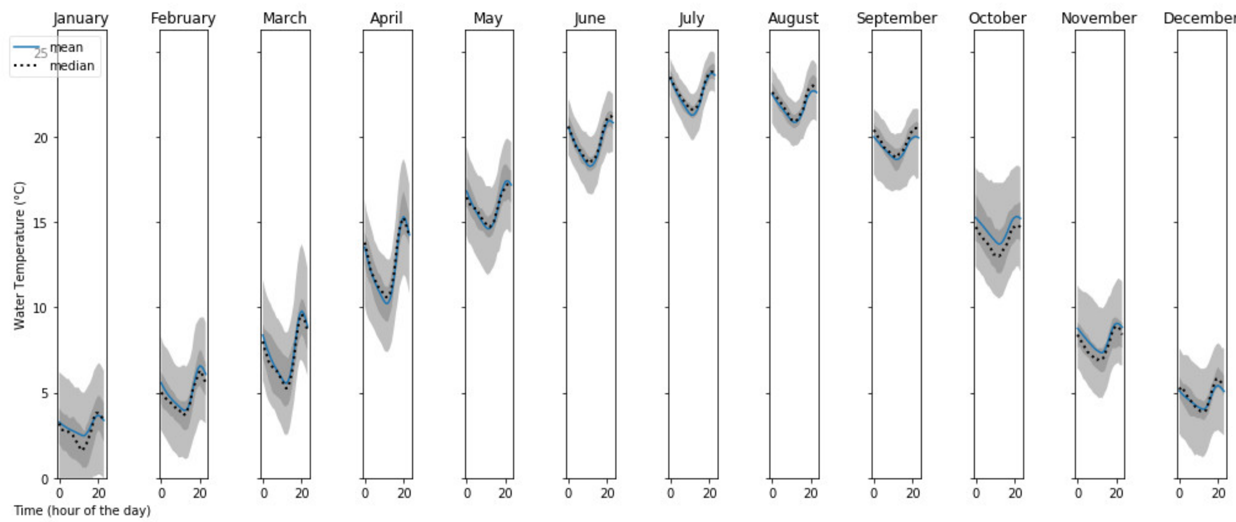
This graph is designed for discovering cyclical components within a series. For example, there is a 'diurnal' cycle in the amount of sunlight that you get over the course of a day. Because the amount of sunlight also changes with the seasons, you can see an 'annual' cycle in sunlight and in temperatures over the course of a year.

```
In  [7]: hf.cycleplot(data, 'annual-week')
Out [7]:
```



In the graph above, several years of data for a site along the Shenandoah River were grouped into 52 different bins, each corresponding to a different week of the year. Then the mean and median values in each bin were connected with lines. A light gray band was drawn around the 0.2 to 0.8 quantile range and a dark gray band was drawn around the 0.4 to 0.6 quantile range.

```
In  [7]: hf.cycleplot(temp_data, 'diurnal', compare='month', y_label="Water Temperature␣
↪(°C)")
Out [7]:
```



In this graph, we are showing how the water temperature varies over the course of a day during each of the twelve months of the year. Temperatures are warmest in July and coldest in January. However, over the course of a day, the lowest temperature occurs at hour 12 and the warmest temperature around hour 20. Since these temperatures are in UTC, and this site is in the Eastern Time Zone (UTC-5), these times correspond to 7 am and 3 pm (20 -5 = 15:00 hours, or 3pm).

7.4 Pandas plotting

The Pandas dataframe has several different graphing methods built-in. Access these methods using dot notation, like this:

Plotting a hydrograph:

```
In  [1]: data.plot()
```

Plotting a histogram:

```
In  [1]: data.hist()
In  [1]: data.plot.hist()
```

Box plot:

```
In  [1]: data.plot.box()
```

Kernel density plot:

```
In  [1]: data.plot.kde()
```

CHAPTER

EIGHT

REQUESTING SITE INFORMATION

The USGS provides a site file for every stream gauge in its network. This file contains information about the location where the data are collected and about the watershed. You can request this information using the hydrofunctions `site_file()` function. This will return a `hydroRDB` object, which contains a table (dataframe) with a row for every site that you request, and a header that describes every column in the dataset.

Some of the most useful information provided by the site file includes:

- latitude, longitude, altitude, & datum
- location data including state, county, and Hydrologic Unit Codes (HUC)
- accuracy codes
- additional codes describing the topographic setting and aquifer
- watershed area & landcover
- well data (if a well is present): depth & aquifer code

For more information about a site and the data collected at the site, try these sources:

- To access information about the data collected at a site, use the `data_catalog()` function.
- To access the rating curve at a site (for translating water stage into discharge), use the `rating_curve()` function.
- To access field data collected by USGS personnel during site visits, use the `field_meas()` function.
- To access the annual peak discharges at a site, use the `peaks()` function.
- To access daily, monthly, or annual statistics for data at a site, use the `stats()` function.

[1]:
```
import hydrofunctions as hf
```

[2]:
```
output = hf.site_file('01585200')
```

```
Retrieved the site file for site #01585200 from https://waterservices.usgs.gov/nwis/site/
↪?format=rdb&sites=01585200&siteOutput=expanded&siteStatus=all
```

Our new 'output' is a hydroRDB object. It has several useful properties, including:

- **.table**, which returns a dataframe of the data. Each row corresponds to a different site.
- **.header**, which is the original descriptive header provided by the USGS. It lists and describes the variables in the dataset.
- **.columns**, which is a list of the column names
- **.dtypes**, which is a list of the data types and column widths for each variable in the USGS RDB format.

If you print or evaluate the hydroRDB object, it will return a tuple of the header and dataframe table, like this:

[3]:
```
output
```

[3]:
```
hydroRDB(header=#
#
# US Geological Survey
# retrieved: 2021-08-09 22:28:38 -04:00    (vaas01)
#
# The Site File stores location and general information about groundwater,
# surface water, and meteorological sites
# for sites in USA.
#
# File-format description:  http://help.waterdata.usgs.gov/faq/about-tab-delimited-output
# Automated-retrieval info: http://waterservices.usgs.gov/rest/Site-Service.html
#
# Contact:   gs-w_support_nwisweb@usgs.gov
#
# The following selected fields are included in this output:
#
#  agency_cd       -- Agency
#  site_no         -- Site identification number
#  station_nm      -- Site name
#  site_tp_cd      -- Site type
#  lat_va          -- DMS latitude
#  long_va         -- DMS longitude
#  dec_lat_va      -- Decimal latitude
#  dec_long_va     -- Decimal longitude
#  coord_meth_cd   -- Latitude-longitude method
#  coord_acy_cd    -- Latitude-longitude accuracy
#  coord_datum_cd  -- Latitude-longitude datum
#  dec_coord_datum_cd -- Decimal Latitude-longitude datum
#  district_cd     -- District code
#  state_cd        -- State code
#  county_cd       -- County code
#  country_cd      -- Country code
#  land_net_ds     -- Land net location description
#  map_nm          -- Name of location map
#  map_scale_fc    -- Scale of location map
#  alt_va          -- Altitude of Gage/land surface
#  alt_meth_cd     -- Method altitude determined
#  alt_acy_va      -- Altitude accuracy
#  alt_datum_cd    -- Altitude datum
#  huc_cd          -- Hydrologic unit code
#  basin_cd        -- Drainage basin code
#  topo_cd         -- Topographic setting code
#  instruments_cd  -- Flags for instruments at site
#  construction_dt -- Date of first construction
#  inventory_dt    -- Date site established or inventoried
#  drain_area_va   -- Drainage area
#  contrib_drain_area_va -- Contributing drainage area
#  tz_cd           -- Time Zone abbreviation
#  local_time_fg   -- Site honors Daylight Savings Time
#  reliability_cd  -- Data reliability code
#  gw_file_cd      -- Data-other GW files
```

(continues on next page)

(continued from previous page)

```
#  nat_aqfr_cd     -- National aquifer code
#  aqfr_cd         -- Local aquifer code
#  aqfr_type_cd    -- Local aquifer type code
#  well_depth_va   -- Well depth
#  hole_depth_va   -- Hole depth
#  depth_src_cd    -- Source of depth data
#  project_no      -- Project number
#,
table=  agency_cd   site_no                                station_nm site_tp_cd  \
0      USGS  01585200  WEST BRANCH HERRING RUN AT IDLEWYLDE, MD         ST

     lat_va   long_va  dec_lat_va  dec_long_va coord_meth_cd coord_acy_cd  \
0  392225.1  763503.6   39.373639   -76.584333             M            S

   ... local_time_fg reliability_cd  gw_file_cd  nat_aqfr_cd  aqfr_cd  \
0  ...             Y            NaN    NNNNNNNN          NaN      NaN

  aqfr_type_cd  well_depth_va hole_depth_va  depth_src_cd  project_no
0          NaN            NaN           NaN           NaN         NaN

[1 rows x 42 columns]
```

[ ]:

CHAPTER

NINE

REQUESTING A DATA CATALOG

Almost every site or 'station' in the NWIS network collects more than one type of data. A simple way to find out what gets collected at a station would be to request everything collected over the past day, like this:

```
[1]: import hydrofunctions as hf
karthaus = hf.NWIS('01542500', 'iv', period='P1D')
```

```
Requested data from https://waterservices.usgs.gov/nwis/iv/?format=json%2C1.1&
↪sites=01542500&period=P1D
```

You can list what is contained in the request:

```
[2]: karthaus
```

```
[2]: USGS:01542500: WB Susquehanna River at Karthaus, PA
    00010: <15 * Minutes>  Temperature, water, degrees Celsius
    00060: <15 * Minutes>  Discharge, cubic feet per second
    00065: <15 * Minutes>  Gage height, feet
    00095: <15 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00300: <15 * Minutes>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <15 * Minutes>  pH, water, unfiltered, field, standard units
Start: 2021-08-09 04:00:00+00:00
End:   2021-08-10 03:30:00+00:00
```

The basic NWIS object will provide a list of every parameter collected at the site, the frequency of observations for that parameter, the name of the parameter, and the units of the observations. It also tells you the date and time of the first and last observation in the request.

This is great, but it doesn't tell you when a parameter was first collected, or if a parameter was discontinued. If you leave out the 'period' part of the request, the USGS will give you the most recent value for every parameter, no matter how old, but this still doesn't tell you when observations were first collected.

For more detailed information about the parameters collected at a site, request a 'data catalog' using the `data_catalog()` function. This will return a `hydroRDB` object containing a table (dataframe) with a row for every parameter that you request, and a header that describes every column in the dataset.

Some of the most useful information in the data catalog are the:

- data type code: describes the frequency of observations
  - dv: daily values
  - uv, rt, or iv: 'real time' data collected more frequently than daily
  - sv: site visits conducted irregularly

    - ad: values listed in the USGS Annual Water Reports
    - more information: https://waterservices.usgs.gov/rest/Site-Service.html#outputDataTypeCd

- parameter code: describes the type of data collected
- statistic code: describes the statistic used to report the parameter
- begin date, end date: the first and last observation made for this parameter
- count_nu: the number of observations made between the start and end dates.

More information about the values in the Data Catalog are located in the header, and also from https://waterservices.usgs.gov/rest/Site-Service.html

For more information about a site and the data collected at the site, try these sources:

- To access information about the site itself, use the `site_file()` function.
- To access the rating curve at a site (for translating water stage into discharge), use the `rating_curve()` function.
- To access field data collected by USGS personnel during site visits, use the `field_meas()` function.
- To access the annual peak discharges at a site, use the `peaks()` function.
- To access daily, monthly, or annual statistics for data at a site, use the `stats()` function.

9.1 Example Usage

[3]:
```
output = hf.data_catalog('01585200')
```

```
Retrieved the data catalog for site #01585200 from https://waterservices.usgs.gov/nwis/
↪site/?format=rdb&sites=01585200&seriesCatalogOutput=true&siteStatus=all
```

Our new 'output' is a hydroRDB object. It has several useful properties, including:

- **.table**, which returns a dataframe of the data. Each row corresponds to a different parameter.
- **.header**, which is the original descriptive header provided by the USGS. It lists and describes the variables in the dataset.
- **.columns**, which is a list of the column names
- **.dtypes**, which is a list of the data types and column widths for each variable in the USGS RDB format.

If you print or evaluate the hydroRDB object, it will return a tuple of the header and dataframe table.

[4]:
```
print(output.header)
```

```
#
#
# US Geological Survey
# retrieved: 2021-08-09 23:51:11 -04:00        (sdas01)
#
# The Site File stores location and general information about groundwater,
# surface water, and meteorological sites
# for sites in USA.
#
# File-format description:  http://help.waterdata.usgs.gov/faq/about-tab-delimited-output
# Automated-retrieval info: http://waterservices.usgs.gov/rest/Site-Service.html
#
```

(continues on next page)

(continued from previous page)

```
# Contact:   gs-w_support_nwisweb@usgs.gov
#
# The following selected fields are included in this output:
#
#  agency_cd       -- Agency
#  site_no         -- Site identification number
#  station_nm      -- Site name
#  site_tp_cd      -- Site type
#  dec_lat_va      -- Decimal latitude
#  dec_long_va     -- Decimal longitude
#  coord_acy_cd    -- Latitude-longitude accuracy
#  dec_coord_datum_cd -- Decimal Latitude-longitude datum
#  alt_va          -- Altitude of Gage/land surface
#  alt_acy_va      -- Altitude accuracy
#  alt_datum_cd    -- Altitude datum
#  huc_cd          -- Hydrologic unit code
#  data_type_cd    -- Data type
#  parm_cd         -- Parameter code
#  stat_cd         -- Statistical code
#  ts_id           -- Internal timeseries ID
#  loc_web_ds      -- Additional measurement description
#  medium_grp_cd   -- Medium group code
#  parm_grp_cd     -- Parameter group code
#  srs_id          -- SRS ID
#  access_cd       -- Access code
#  begin_date      -- Begin date
#  end_date        -- End date
#  count_nu        -- Record count
#
```

[5]:

```
# Transposing the table to show all of the columns as rows:
output.table.T
```

[5]:

```
                                   2060003
data_type_cd                            dv
parm_cd                               60.0
stat_cd                                3.0
ts_id                                68214
loc_web_ds                             NaN
medium_grp_cd                          wat
parm_grp_cd                            NaN
srs_id                             1645423
access_cd                                0
begin_date                      1957-07-01
end_date                        2021-08-08
count_nu                             19913

                                         2   \
agency_cd                             USGS
site_no                           01585200
station_nm      WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                              ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                             39.373639
dec_long_va                                           -76.584333
coord_acy_cd                                                   S
dec_coord_datum_cd                                         NAD83
alt_va                                                    278.13
alt_acy_va                                                  0.01
alt_datum_cd                                              NAVD88
huc_cd                                                   2060003
data_type_cd                                                  pk
parm_cd                                                      NaN
stat_cd                                                      NaN
ts_id                                                          0
loc_web_ds                                                   NaN
medium_grp_cd                                                wat
parm_grp_cd                                                  NaN
srs_id                                                         0
access_cd                                                      0
begin_date                                            1958-07-06
end_date                                              2020-07-22
count_nu                                                      54

                                                               3  \
agency_cd                                                   USGS
site_no                                                 01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                    ST
dec_lat_va                                             39.373639
dec_long_va                                           -76.584333
coord_acy_cd                                                   S
dec_coord_datum_cd                                         NAD83
alt_va                                                    278.13
alt_acy_va                                                  0.01
alt_datum_cd                                              NAVD88
huc_cd                                                   2060003
data_type_cd                                                  qw
parm_cd                                                      NaN
stat_cd                                                      NaN
ts_id                                                          0
loc_web_ds                                                   NaN
medium_grp_cd                                                wat
parm_grp_cd                                                  ALL
srs_id                                                         0
access_cd                                                      0
begin_date                                            1965-01-06
end_date                                              1987-08-20
count_nu                                                     119

                                                               4  \
agency_cd                                                   USGS
site_no                                                 01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                    ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                              39.373639
dec_long_va                                            -76.584333
coord_acy_cd                                                    S
dec_coord_datum_cd                                          NAD83
alt_va                                                     278.13
alt_acy_va                                                   0.01
alt_datum_cd                                               NAVD88
huc_cd                                                    2060003
data_type_cd                                                   qw
parm_cd                                                      10.0
stat_cd                                                       NaN
ts_id                                                           0
loc_web_ds                                                    NaN
medium_grp_cd                                                 wat
parm_grp_cd                                                   PHY
srs_id                                                    1645597
access_cd                                                       0
begin_date                                             1965-01-06
end_date                                               1987-08-20
count_nu                                                      119

                                                                5   \
agency_cd                                                    USGS
site_no                                                  01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                     ST
dec_lat_va                                              39.373639
dec_long_va                                            -76.584333
coord_acy_cd                                                    S
dec_coord_datum_cd                                          NAD83
alt_va                                                     278.13
alt_acy_va                                                   0.01
alt_datum_cd                                               NAVD88
huc_cd                                                    2060003
data_type_cd                                                   qw
parm_cd                                                      20.0
stat_cd                                                       NaN
ts_id                                                           0
loc_web_ds                                                    NaN
medium_grp_cd                                                 wat
parm_grp_cd                                                   PHY
srs_id                                                    1645720
access_cd                                                       0
begin_date                                             1965-01-06
end_date                                               1987-08-20
count_nu                                                      119

                                                                6   \
agency_cd                                                    USGS
site_no                                                  01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                     ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                                39.373639
dec_long_va                                              -76.584333
coord_acy_cd                                                      S
dec_coord_datum_cd                                            NAD83
alt_va                                                       278.13
alt_acy_va                                                     0.01
alt_datum_cd                                                 NAVD88
huc_cd                                                      2060003
data_type_cd                                                     qw
parm_cd                                                        28.0
stat_cd                                                         NaN
ts_id                                                             0
loc_web_ds                                                      NaN
medium_grp_cd                                                   wat
parm_grp_cd                                                     INF
srs_id                                                            0
access_cd                                                         0
begin_date                                               1981-10-06
end_date                                                 1984-01-30
count_nu                                                          7

                                                                 7   \
agency_cd                                                      USGS
site_no                                                    01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                       ST
dec_lat_va                                                39.373639
dec_long_va                                              -76.584333
coord_acy_cd                                                      S
dec_coord_datum_cd                                            NAD83
alt_va                                                       278.13
alt_acy_va                                                     0.01
alt_datum_cd                                                 NAVD88
huc_cd                                                      2060003
data_type_cd                                                     qw
parm_cd                                                        61.0
stat_cd                                                         NaN
ts_id                                                             0
loc_web_ds                                                      NaN
medium_grp_cd                                                   wat
parm_grp_cd                                                     PHY
srs_id                                                      1645415
access_cd                                                         0
begin_date                                               1965-01-06
end_date                                                 1987-08-20
count_nu                                                        119

                                                                 8   \
agency_cd                                                      USGS
site_no                                                    01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                       ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                             39.373639
dec_long_va                                           -76.584333
coord_acy_cd                                                   S
dec_coord_datum_cd                                         NAD83
alt_va                                                    278.13
alt_acy_va                                                  0.01
alt_datum_cd                                              NAVD88
huc_cd                                                   2060003
data_type_cd                                                  qw
parm_cd                                                     65.0
stat_cd                                                      NaN
ts_id                                                          0
loc_web_ds                                                   NaN
medium_grp_cd                                                wat
parm_grp_cd                                                  PHY
srs_id                                                  17164583
access_cd                                                      0
begin_date                                            1965-01-06
end_date                                              1987-08-20
count_nu                                                     118

                                                               9   \
agency_cd                                                   USGS
site_no                                                 01585200
station_nm            WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                    ST
dec_lat_va                                             39.373639
dec_long_va                                           -76.584333
coord_acy_cd                                                   S
dec_coord_datum_cd                                         NAD83
alt_va                                                    278.13
alt_acy_va                                                  0.01
alt_datum_cd                                              NAVD88
huc_cd                                                   2060003
data_type_cd                                                  qw
parm_cd                                                  30207.0
stat_cd                                                      NaN
ts_id                                                          0
loc_web_ds                                                   NaN
medium_grp_cd                                                wat
parm_grp_cd                                                  PHY
srs_id                                                  17164583
access_cd                                                      0
begin_date                                            1965-01-06
end_date                                              1987-08-20
count_nu                                                     118

                                                              10   \
agency_cd                                                   USGS
site_no                                                 01585200
station_nm            WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                    ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                              39.373639
dec_long_va                                            -76.584333
coord_acy_cd                                                    S
dec_coord_datum_cd                                          NAD83
alt_va                                                     278.13
alt_acy_va                                                   0.01
alt_datum_cd                                               NAVD88
huc_cd                                                    2060003
data_type_cd                                                   qw
parm_cd                                                   30209.0
stat_cd                                                       NaN
ts_id                                                           0
loc_web_ds                                                    NaN
medium_grp_cd                                                 wat
parm_grp_cd                                                   PHY
srs_id                                                    1645415
access_cd                                                       0
begin_date                                             1965-01-06
end_date                                               1987-08-20
count_nu                                                      119

                                                               11  \
agency_cd                                                    USGS
site_no                                                  01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                     ST
dec_lat_va                                              39.373639
dec_long_va                                            -76.584333
coord_acy_cd                                                    S
dec_coord_datum_cd                                          NAD83
alt_va                                                     278.13
alt_acy_va                                                   0.01
alt_datum_cd                                               NAVD88
huc_cd                                                    2060003
data_type_cd                                                   sv
parm_cd                                                       NaN
stat_cd                                                       NaN
ts_id                                                           0
loc_web_ds                                                    NaN
medium_grp_cd                                                 wat
parm_grp_cd                                                   NaN
srs_id                                                          0
access_cd                                                       0
begin_date                                             1957-07-10
end_date                                               2021-06-17
count_nu                                                      511

                                                               12  \
agency_cd                                                    USGS
site_no                                                  01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                     ST
```

(continues on next page)

(continued from previous page)

```
dec_lat_va                                            39.373639
dec_long_va                                          -76.584333
coord_acy_cd                                                  S
dec_coord_datum_cd                                        NAD83
alt_va                                                   278.13
alt_acy_va                                                 0.01
alt_datum_cd                                             NAVD88
huc_cd                                                  2060003
data_type_cd                                                 uv
parm_cd                                                    60.0
stat_cd                                                     NaN
ts_id                                                     69659
loc_web_ds                                                  NaN
medium_grp_cd                                               wat
parm_grp_cd                                                 NaN
srs_id                                                  1645423
access_cd                                                     0
begin_date                                           1996-10-01
end_date                                             2021-08-09
count_nu                                                   9078

                                                             13
agency_cd                                                  USGS
site_no                                                01585200
station_nm          WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
site_tp_cd                                                   ST
dec_lat_va                                            39.373639
dec_long_va                                          -76.584333
coord_acy_cd                                                  S
dec_coord_datum_cd                                        NAD83
alt_va                                                   278.13
alt_acy_va                                                 0.01
alt_datum_cd                                             NAVD88
huc_cd                                                  2060003
data_type_cd                                                 uv
parm_cd                                                    65.0
stat_cd                                                     NaN
ts_id                                                     69660
loc_web_ds                                                  NaN
medium_grp_cd                                               wat
parm_grp_cd                                                 NaN
srs_id                                                 17164583
access_cd                                                     0
begin_date                                           2007-10-01
end_date                                             2021-08-09
count_nu                                                   5061
```

CHAPTER TEN

REQUESTING PEAK DISCHARGES

It is possible to request a list of each year's largest flood using the USGS's Peak Discharge service. This list of annual peak discharges is perfect for doing a flood frequency analysis: you can calculate recurrence intervals and plot a flood frequency graph using these data.

The data provided by the USGS peak service are based on a 'water year', which starts on October 1st and ends in the last moments of September. This unusual definition for the start of the year is standard practice in US hydrology: September/October often marks the lowest flows of the year for much of the country, so it is less likely that there will be a large flood that spans the end of one year and continues on into the new year. As an example of the problems that we are trying to avoid by using a 'water year', imagine the largest flood in a decade peaking in the last days of December. As the water slowly receeds, it may still be quite high on January 1st. The tail end of last year's flood may well be the highest discharge of the whole next year! This would interfere with our statistical assumptions about each flood event. By setting the end of the year/start of the year to the low-flow season, we mostly avoid double-counting floods without any extra processing.

[1]:
```
import hydrofunctions as hf
```

[2]:
```
output = hf.peaks('01541200')
```

```
Retrieving annual peak discharges for site # 01541200  from  https://nwis.waterdata.usgs.
↪gov/nwis/peak?site_no=01541200&agency_cd=USGS&format=rdb
```

Our new 'output' is a hydroRDB object. It has several useful properties, including:

- **.table**, which returns a dataframe of the data
- **.header**, which is the original descriptive header provided by the USGS. It lists and describes the variables in the dataset.
- **.columns**, which is a list of the column names
- **.dtypes**, which is a list of the data types and column widths for each variable in the USGS RDB format.

If you print or evaluate the hydroRDB object, it will return a tuple of the header and dataframe table, like this:

[3]:
```
output
```

[3]:
```
#
# U.S. Geological Survey
# National Water Information System
# Retrieved: 2020-03-04 21:25:01 EST
#
# ---------------------------------- WARNING ----------------------------------------
# Some of the data that you have obtained from this U.S. Geological Survey database
# may not have received Director's approval. Any such data values are qualified
```

(continues on next page)

(continued from previous page)

```
# as provisional and are subject to revision. Provisional data are released on the
# condition that neither the USGS nor the United States Government may be held liable
# for any damages resulting from its use.
#
# More data may be available offline.
# For more information on these data,  contact  USGS Water Data Inquiries.
# This file contains the annual peak streamflow data.
#
# This information includes the following fields:
#
#  agency_cd     Agency Code
#  site_no       USGS station number
#  peak_dt       Date of peak streamflow (format YYYY-MM-DD)
#  peak_tm       Time of peak streamflow (24 hour format, 00:00 - 23:59)
#  peak_va       Annual peak streamflow value in cfs
#  peak_cd       Peak Discharge-Qualification codes (see explanation below)
#  gage_ht       Gage height for the associated peak streamflow in feet
#  gage_ht_cd    Gage height qualification codes
#  year_last_pk  Peak streamflow reported is the highest since this year
#  ag_dt         Date of maximum gage-height for water year (if not concurrent with peak)
#  ag_tm         Time of maximum gage-height for water year (if not concurrent with peak
#  ag_gage_ht    maximum Gage height for water year in feet (if not concurrent with peak
#  ag_gage_ht_cd maximum Gage height code
#
# Sites in this file include:
#  USGS 01541200 WB Susquehanna River near Curwensville, PA
#
# Peak Streamflow-Qualification Codes(peak_cd):
#   1 ... Discharge is a Maximum Daily Average
#   2 ... Discharge is an Estimate
#   3 ... Discharge affected by Dam Failure
#   4 ... Discharge less than indicated value,
#           which is Minimum Recordable Discharge at this site
#   5 ... Discharge affected to unknown degree by
#           Regulation or Diversion
#   6 ... Discharge affected by Regulation or Diversion
#   7 ... Discharge is an Historic Peak
#   8 ... Discharge actually greater than indicated value
#   9 ... Discharge due to Snowmelt, Hurricane,
#           Ice-Jam or Debris Dam breakup
#   A ... Year of occurrence is unknown or not exact
#   Bd ... Day of occurrence is unknown or not exact
#   Bm ... Month of occurrence is unknown or not exact
#   C ... All or part of the record affected by Urbanization,
#            Mining, Agricultural changes, Channelization, or other
#   F ... Peak supplied by another agency
#   O ... Opportunistic value not from systematic data collection
#   R ... Revised
#
# Gage height qualification codes(gage_ht_cd,ag_gage_ht_cd):
#   1 ... Gage height affected by backwater
#   2 ... Gage height not the maximum for the year
```

(continues on next page)

(continued from previous page)

```
#   3 ... Gage height at different site and(or) datum
#   4 ... Gage height below minimum recordable elevation
#   5 ... Gage height is an estimate
#   6 ... Gage datum changed during this year
#   7 ... Debris, mud, or hyper-concentrated flow
#   8 ... Gage height tidally affected
#   Bd ... Day of occurrence is unknown or not exact
#   Bm ... Month of occurrence is unknown or not exact
#   F ... Peak supplied by another agency
#   R ... Revised
#
#
           agency_cd   site_no peak_tm  peak_va  peak_cd  gage_ht gage_ht_cd  \
peak_dt
1956-07-03      USGS  01541200     NaN    10200      NaN    10.89        NaN
1957-04-06      USGS  01541200     NaN     9240      NaN    10.44        NaN
1958-05-08      USGS  01541200     NaN     4800      NaN     7.66        NaN
1959-01-22      USGS  01541200     NaN    12500      NaN    12.12        NaN
1960-03-31      USGS  01541200     NaN    10400      NaN    11.03        NaN
1961-02-26      USGS  01541200     NaN    12100      NaN    11.62        NaN
1962-02-24      USGS  01541200     NaN     9000      NaN    10.10        NaN
1963-03-18      USGS  01541200     NaN     9000      NaN    10.00        2,3
1964-03-10      USGS  01541200     NaN    15700      NaN    14.19        NaN
1965-01-03      USGS  01541200     NaN     5840      NaN     8.34        2,3
1966-02-20      USGS  01541200     NaN     4400      6.0     7.31        NaN
1967-03-07      USGS  01541200     NaN     4010      6.0     7.28        NaN
1968-04-10      USGS  01541200     NaN     7120      6.0     9.28        NaN
1969-04-24      USGS  01541200     NaN     4780      6.0     7.34        NaN
1970-04-08      USGS  01541200     NaN     5640      6.0     7.77        NaN
1971-02-23      USGS  01541200     NaN     5280      6.0     7.59        NaN
1972-06-25      USGS  01541200     NaN     8590      6.0    11.40        NaN
1973-02-03      USGS  01541200     NaN     4080      6.0     7.29        NaN
1974-01-22      USGS  01541200     NaN     3960      6.0     6.84        NaN
1975-03-03      USGS  01541200     NaN     5940      6.0     8.74        NaN
1976-02-23      USGS  01541200     NaN     4720      6.0     7.62        NaN
1977-07-23      USGS  01541200     NaN     5660      6.0     8.47        NaN
1978-05-18      USGS  01541200     NaN     4920      6.0     7.80        NaN
1979-03-07      USGS  01541200     NaN     5610      6.0     8.42        NaN
1980-04-01      USGS  01541200     NaN     3650      6.0     6.78        NaN
1981-02-23      USGS  01541200     NaN     5170      6.0     8.02        NaN
1982-03-15      USGS  01541200     NaN     4330      6.0     7.29        NaN
1983-06-21      USGS  01541200     NaN     4560      6.0     7.47        NaN
1984-06-20      USGS  01541200     NaN     4980      6.0     7.82        NaN
1985-04-02      USGS  01541200     NaN     4320      6.0     7.28        NaN
...              ...       ...     ...      ...      ...      ...        ...
1988-02-03      USGS  01541200     NaN     4210      6.0     7.26        NaN
1989-06-23      USGS  01541200     NaN     4660      6.0     7.61        NaN
1990-07-14      USGS  01541200     NaN     4290      6.0     7.32        NaN
1990-12-19      USGS  01541200     NaN     4300      6.0     7.33        NaN
1992-04-01      USGS  01541200     NaN     3370      6.0     6.61        NaN
1993-04-07      USGS  01541200     NaN     5250      6.0     8.31        NaN
1994-03-30      USGS  01541200     NaN     5390      6.0     8.23        NaN
```

(continues on next page)

(continued from previous page)

```
1994-11-28      USGS  01541200     NaN     3650      6.0     6.82        NaN
1996-01-23      USGS  01541200     NaN     5890      6.0     8.68        NaN
1997-05-26      USGS  01541200     NaN     5390      6.0     8.23        NaN
1997-11-10      USGS  01541200     NaN     5690      6.0     8.49        NaN
1999-01-26      USGS  01541200     NaN     5590      6.0     8.40        NaN
1999-11-26      USGS  01541200     NaN     4220      6.0     7.20        NaN
2000-10-18      USGS  01541200     NaN     3720      6.0     6.80        NaN
2002-03-27      USGS  01541200     NaN     5610      6.0     8.42        NaN
2003-01-01      USGS  01541200     NaN     4800      6.0     7.69        NaN
2004-01-08      USGS  01541200     NaN     5700      6.0     8.50        NaN
2005-01-13      USGS  01541200     NaN     5710      6.0     8.51        NaN
2005-11-30      USGS  01541200     NaN     4310      6.0     7.27        NaN
2006-11-17      USGS  01541200     NaN     5190      6.0     8.04        NaN
2008-03-21      USGS  01541200     NaN     4650      6.0     7.55        NaN
2008-12-20      USGS  01541200     NaN     5170      6.0     8.03        NaN
2010-03-15      USGS  01541200     NaN     4860      6.0     7.74        NaN
2011-09-11      USGS  01541200     NaN     5350      6.0     8.19        NaN
2011-11-23      USGS  01541200     NaN     4460      6.0     7.39        NaN
2013-06-27      USGS  01541200     NaN     4430      6.0     7.35        NaN
2013-12-23      USGS  01541200   08:45     4120      6.0     7.06        NaN
2015-03-16      USGS  01541200   10:15     5240      6.0     8.08        NaN
2016-02-03      USGS  01541200   23:00     2860      6.0     5.94        NaN
2016-12-19      USGS  01541200     NaN     5130      6.0     7.98        NaN

            year_last_pk       ag_dt  ag_tm  ag_gage_ht  ag_gage_ht_cd
peak_dt
1956-07-03           NaN         NaN    NaN         NaN            NaN
1957-04-06           NaN         NaN    NaN         NaN            NaN
1958-05-08           NaN         NaN    NaN         NaN            NaN
1959-01-22           NaN         NaN    NaN         NaN            NaN
1960-03-31           NaN         NaN    NaN         NaN            NaN
1961-02-26           NaN         NaN    NaN         NaN            NaN
1962-02-24           NaN         NaN    NaN         NaN            NaN
1963-03-18           NaN  1963-03-06    NaN       11.50            1.0
1964-03-10           NaN         NaN    NaN         NaN            NaN
1965-01-03           NaN  1965-02-08    NaN       10.79            1.0
1966-02-20           NaN         NaN    NaN         NaN            NaN
1967-03-07           NaN         NaN    NaN         NaN            NaN
1968-04-10           NaN         NaN    NaN         NaN            NaN
1969-04-24           NaN         NaN    NaN         NaN            NaN
1970-04-08           NaN         NaN    NaN         NaN            NaN
1971-02-23           NaN         NaN    NaN         NaN            NaN
1972-06-25           NaN         NaN    NaN         NaN            NaN
1973-02-03           NaN         NaN    NaN         NaN            NaN
1974-01-22           NaN         NaN    NaN         NaN            NaN
1975-03-03           NaN         NaN    NaN         NaN            NaN
1976-02-23           NaN         NaN    NaN         NaN            NaN
1977-07-23           NaN         NaN    NaN         NaN            NaN
1978-05-18           NaN         NaN    NaN         NaN            NaN
1979-03-07           NaN         NaN    NaN         NaN            NaN
1980-04-01           NaN         NaN    NaN         NaN            NaN
1981-02-23           NaN         NaN    NaN         NaN            NaN
```

(continues on next page)

(continued from previous page)

```
1982-03-15          NaN         NaN    NaN          NaN            NaN
1983-06-21          NaN         NaN    NaN          NaN            NaN
1984-06-20          NaN         NaN    NaN          NaN            NaN
1985-04-02          NaN         NaN    NaN          NaN            NaN
...                 ...         ...    ...          ...            ...
1988-02-03          NaN         NaN    NaN          NaN            NaN
1989-06-23          NaN         NaN    NaN          NaN            NaN
1990-07-14          NaN         NaN    NaN          NaN            NaN
1990-12-19          NaN         NaN    NaN          NaN            NaN
1992-04-01          NaN         NaN    NaN          NaN            NaN
1993-04-07          NaN         NaN    NaN          NaN            NaN
1994-03-30          NaN         NaN    NaN          NaN            NaN
1994-11-28          NaN         NaN    NaN          NaN            NaN
1996-01-23          NaN         NaN    NaN          NaN            NaN
1997-05-26          NaN         NaN    NaN          NaN            NaN
1997-11-10          NaN         NaN    NaN          NaN            NaN
1999-01-26          NaN         NaN    NaN          NaN            NaN
1999-11-26          NaN         NaN    NaN          NaN            NaN
2000-10-18          NaN         NaN    NaN          NaN            NaN
2002-03-27          NaN         NaN    NaN          NaN            NaN
2003-01-01          NaN         NaN    NaN          NaN            NaN
2004-01-08          NaN         NaN    NaN          NaN            NaN
2005-01-13          NaN         NaN    NaN          NaN            NaN
2005-11-30          NaN         NaN    NaN          NaN            NaN
2006-11-17          NaN         NaN    NaN          NaN            NaN
2008-03-21          NaN         NaN    NaN          NaN            NaN
2008-12-20          NaN         NaN    NaN          NaN            NaN
2010-03-15          NaN         NaN    NaN          NaN            NaN
2011-09-11          NaN         NaN    NaN          NaN            NaN
2011-11-23          NaN         NaN    NaN          NaN            NaN
2013-06-27          NaN         NaN    NaN          NaN            NaN
2013-12-23          NaN         NaN    NaN          NaN            NaN
2015-03-16          NaN         NaN    NaN          NaN            NaN
2016-02-03          NaN         NaN    NaN          NaN            NaN
2016-12-19          NaN         NaN    NaN          NaN            NaN

[62 rows x 12 columns]
```

CHAPTER

ELEVEN

REQUESTING STATISTICS FROM THE USGS STATISTICS SERVICE

The USGS calculates various types of statistics for its data and provides these values through a web service. You can access this service through the `stats` function. Learn more about the USGS Statistics Service.

There are three types of report that you can request using the `StatReportType` parameter.

- **'annual'**: This summarizes all of the official daily data for each year using max, min, mean, and the 5, 10, 20, 25, 50, 75, 80, 90, and 95th percentiles.
- **'monthly'**: This calculates the mean of the 28 to 31 daily values that occur for each of the months in each of the years of record.
- **'daily'**: This summarizes all of the data for this month and day, using max, min, mean, and the 5, 10, 20, 25, 50, 75, 80, 90, and 95th percentiles.

11.1 Request multiple sites

You can request multiple sites by separating them with commas, like this: `'01541200,01542500'`

11.2 Providing additional arguments

The USGS Statistics Service allows you to specify a wide array of additional parameters in your request. You can provide these parameters as keyword arguments, like in this example:

```
hf.stats('01452500', parameterCD='00060')
```

This will only request statistics for discharge, which is specified with the '00060' parameter code.

11.3 Limiting requests to only certain parameters

The default behavior for the USGS Statistics Service is to provide statistics for every parameter that is collected at a site. This can make for a long table that you will have to filter by the parameter that you want, like this:

```
my_stat_dataframe.loc(my_stat_dataframe['parameter_cd']='00060')
```

Alternatively, you can just request the parameter that you are interested in, rather than all of the parameters. To limit your request, provide the `parameterCD` keyword argument, like this:

```
hf.stats('01452500', parameterCD='00060')
```

You can request more than one parameter by listing every parameter code that you are interested in, separated by a comma: `parameterCD='00060,00065'`

11.4 Calculating annual statistics using water years

The default behavior for the USGS Statistics Service is to calculate annual statistics using calendar years. Unfortunately, for many places in the US, this will split the wet season in half. Since discharge data tends to be autocorrelated, you are more likely to get a large flood in January 2020 if you had a large flood in December 2019. To fix this, hydrologists often use 'Water Years', which split the year during the more or less dry season, on October 1st. To calculate annual statistics using the water year, provide the `statYearType='water'` argument, like this:

`hf.stats('01452500','annual', statYearType='water')`

11.5 Missing data

The default behavior for the USGS Statistics Service is to not calculate statistics for months or years if there are -ANY- missing values. In other words, in an annual report, every year reported will be based on 365 or 366 (leap year) values. You can override this behavior by providing the `missingData='on'` parameter. This will calculate the statistics as long as there are at least one measurement. You can decide whether or not to use the statistic by looking at the `count_nu` column to see how many values were used to generate the statistic.

11.6 Viewing the metadata header or the data

The USGS accompanies every dataset with a header that explains the data. Hydrofunctions will automatically display this header along with the data. To access just one item, use either the .header or .df attribute.

```
test = stats('01542500')

test        # Print the header & dataframe
test.header # print just the header
test.df     # print just the dataframe.
```

11.7 Examples

The first step as always is to import hydrofunctions.

[1]:
```
import hydrofunctions as hf
print(hf.__version__)
```

```
0.2.0
```

To get started, let's request some data from Karthus, PA to see what typically gets collected there.

[2]:
```
may_2019 = hf.NWIS('01542500', 'dv', '2019-05-01', '2019-06-01')
may_2019
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01542500&startDT=2019-05-01&endDT=2019-06-01
```

[2]:
```
USGS:01542500: WB Susquehanna River at Karthaus, PA
    00010: <Day>  Temperature, water, degrees Celsius
    00060: <Day>  Discharge, cubic feet per second
    00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
```

(continues on next page)

(continued from previous page)

```
↪at 25 degrees Celsius
    00300: <Day>  Dissolved oxygen, water, unfiltered, milligrams per liter
    00400: <Day>  pH, water, unfiltered, field, standard units
Start: 2019-05-01 00:00:00+00:00
End:   2019-06-01 00:00:00+00:00
```

11.7.1 Requesting annual statistics

This site has collected discharge data since 1960, but other parameters, such as water temperature ('00010'), have only been collected since 2010. Unfortunately, in 2010, only 41 days of water temperature measurements were collected. By setting the missingData argument to on, we can ask the USGS to report averages for incomplete years. Now it is up to you to decide if 41 values is an adequate number!

```
[3]: annual_stats = hf.stats('01542500', 'annual', missingData='on')
# Use annual_stats.header to access just the header, or .df for just the dataframe.
# If you don't specify, both will be provided.
annual_stats
```

```
Retrieving annual statistics for site #01542500 from https://waterservices.usgs.gov/nwis/
↪stat/?statReportType=annual&statType=all&sites=01542500&format=rdb&missingData=on
```

```
[3]: #
#
# US Geological Survey, Water Resources Data
# retrieved: 2020-03-04 21:24:34 -05:00    (natwebcaas01)
#
# This file contains USGS Annual Statistics
#
# Note:The statistics generated are based on approved daily-mean data and may not match␣
↪those published by the USGS in official publications.
# The user is responsible for assessment and use of statistics from this site.
# For more details on why the statistics may not match, visit http://help.waterdata.usgs.
↪gov/faq/about-statistics.
#
# Data heading explanations.
# agency_cd       -- agency code
# site_no         -- Site identification number
# parameter_cd    -- Parameter code
# station_nm      -- Site name
# loc_web_ds      -- Additional measurement description
#
# Data for the following 1 site(s) are contained in this file
# agency_cd   site_no      parameter_cd   station_nm (loc_web_ds)
# USGS        01542500     00010          WB Susquehanna River at Karthaus, PA
# USGS        01542500     00060          WB Susquehanna River at Karthaus, PA
# USGS        01542500     00095          WB Susquehanna River at Karthaus, PA
# USGS        01542500     00300          WB Susquehanna River at Karthaus, PA
#
# Explanation of Parameter Codes
# parameter_cd    Parameter Name
# 00010         Temperature, water, degrees Celsius
# 00060         Discharge, cubic feet per second
```

(continues on next page)

(continued from previous page)

```
# 00095         Specific conductance, water, unfiltered, microsiemens per centimeter at␣
↪25 degrees Celsius
# 00300         Dissolved oxygen, water, unfiltered, milligrams per liter
#
# Data heading explanations.
# year_nu     ... Calendar year for value.
# mean_va     ... Mean of daily mean values for this month.
# count_nu    ... Number of values used in the calculation.
#
   agency_cd   site_no parameter_cd   ts_id  loc_web_ds  year_nu  mean_va  \
0       USGS  01542500        00010  118870         NaN     2010     4.70
1       USGS  01542500        00010  118870         NaN     2011    12.92
2       USGS  01542500        00010  118870         NaN     2012    13.98
3       USGS  01542500        00010  118870         NaN     2013    12.76
4       USGS  01542500        00010  118870         NaN     2014    12.43
5       USGS  01542500        00010  118870         NaN     2015    12.46
6       USGS  01542500        00010  118870         NaN     2016    12.64
7       USGS  01542500        00010  118870         NaN     2017    12.37
8       USGS  01542500        00010  118870         NaN     2018    11.64
9       USGS  01542500        00010  118870         NaN     2019    11.80
10      USGS  01542500        00010  118870         NaN     2020     2.46
11      USGS  01542500        00060  118867         NaN     1960   325.50
12      USGS  01542500        00060  118867         NaN     1961  2418.00
13      USGS  01542500        00060  118867         NaN     1962  2048.00
14      USGS  01542500        00060  118867         NaN     1963  1682.00
15      USGS  01542500        00060  118867         NaN     1964  2247.00
16      USGS  01542500        00060  118867         NaN     1965  1820.00
17      USGS  01542500        00060  118867         NaN     1966  2019.00
18      USGS  01542500        00060  118867         NaN     1967  2693.00
19      USGS  01542500        00060  118867         NaN     1968  2045.00
20      USGS  01542500        00060  118867         NaN     1969  1624.00
21      USGS  01542500        00060  118867         NaN     1970  3063.00
22      USGS  01542500        00060  118867         NaN     1971  2522.00
23      USGS  01542500        00060  118867         NaN     1972  3923.00
24      USGS  01542500        00060  118867         NaN     1973  2796.00
25      USGS  01542500        00060  118867         NaN     1974  2713.00
26      USGS  01542500        00060  118867         NaN     1975  3232.00
27      USGS  01542500        00060  118867         NaN     1976  2319.00
28      USGS  01542500        00060  118867         NaN     1977  3114.00
29      USGS  01542500        00060  118867         NaN     1978  2670.00
..       ...       ...          ...     ...         ...      ...      ...
50      USGS  01542500        00060  118867         NaN     2010  2222.00
51      USGS  01542500        00060  118867         NaN     2011  3398.00
52      USGS  01542500        00060  118867         NaN     2012  1793.00
53      USGS  01542500        00060  118867         NaN     2013  1845.00
54      USGS  01542500        00060  118867         NaN     2014  2017.00
55      USGS  01542500        00060  118867         NaN     2015  2462.00
56      USGS  01542500        00060  118867         NaN     2016  1860.00
57      USGS  01542500        00060  118867         NaN     2017  2494.00
58      USGS  01542500        00060  118867         NaN     2018  4482.00
59      USGS  01542500        00060  118867         NaN     2019  2781.00
60      USGS  01542500        00095  118873         NaN     2010   363.40
```

(continues on next page)

(continued from previous page)

```
61      USGS  01542500        00095  118873         NaN     2011   391.70
62      USGS  01542500        00095  118873         NaN     2012   413.90
63      USGS  01542500        00095  118873         NaN     2013   401.70
64      USGS  01542500        00095  118873         NaN     2014   376.40
65      USGS  01542500        00095  118873         NaN     2015   387.10
66      USGS  01542500        00095  118873         NaN     2016   404.50
67      USGS  01542500        00095  118873         NaN     2017   385.20
68      USGS  01542500        00095  118873         NaN     2018   296.70
69      USGS  01542500        00095  118873         NaN     2019   346.40
70      USGS  01542500        00095  118873         NaN     2020   260.60
71      USGS  01542500        00300  118879         NaN     2011     9.56
72      USGS  01542500        00300  118879         NaN     2012     9.42
73      USGS  01542500        00300  118879         NaN     2013     9.38
74      USGS  01542500        00300  118879         NaN     2014     9.45
75      USGS  01542500        00300  118879         NaN     2015     9.91
76      USGS  01542500        00300  118879         NaN     2016     9.30
77      USGS  01542500        00300  118879         NaN     2017     9.07
78      USGS  01542500        00300  118879         NaN     2018     9.71
79      USGS  01542500        00300  118879         NaN     2019     9.33

    count_nu
0         41
1        354
2        360
3        365
4        362
5        365
6        358
7        358
8        353
9        362
10        14
11        92
12       365
13       365
14       365
15       366
16       365
17       365
18       365
19       366
20       365
21       365
22       365
23       366
24       365
25       365
26       365
27       366
28       365
29       365
..       ...
```

(continues on next page)

(continued from previous page)

```
50        365
51        365
52        366
53        365
54        365
55        365
56        366
57        365
58        365
59        346
60         41
61        354
62        358
63        365
64        362
65        365
66        358
67        358
68        353
69        362
70         14
71        227
72        243
73        246
74        224
75        258
76        208
77        205
78        227
79        164

[80 rows x 8 columns]
```

11.7.2 Requesting monthly statistics

The monthly report provides the mean value for each parameter for every month since 1960, when data collection began at this site.

Since this site collects lots of parameters, we can limit our display of the dataframe by filtering everything out except the discharge parameter ('00060').

[4]:
```
monthly_stats = hf.stats('01542500', 'monthly')
monthly_stats.df.loc[monthly_stats.df['parameter_cd']=='00060']
```

```
Retrieving monthly statistics for site #01542500 from https://waterservices.usgs.gov/
↪nwis/stat/?statReportType=monthly&statType=all&sites=01542500&format=rdb
```

[4]:
```
    agency_cd   site_no parameter_cd   ts_id  loc_web_ds  year_nu  month_nu  \
93       USGS  01542500        00060  118867         NaN     1960        10
94       USGS  01542500        00060  118867         NaN     1960        11
95       USGS  01542500        00060  118867         NaN     1960        12
96       USGS  01542500        00060  118867         NaN     1961         1
```

(continues on next page)

(continued from previous page)

```
97       USGS  01542500        00060  118867         NaN     1961         2
98       USGS  01542500        00060  118867         NaN     1961         3
99       USGS  01542500        00060  118867         NaN     1961         4
100      USGS  01542500        00060  118867         NaN     1961         5
101      USGS  01542500        00060  118867         NaN     1961         6
102      USGS  01542500        00060  118867         NaN     1961         7
103      USGS  01542500        00060  118867         NaN     1961         8
104      USGS  01542500        00060  118867         NaN     1961         9
105      USGS  01542500        00060  118867         NaN     1961        10
106      USGS  01542500        00060  118867         NaN     1961        11
107      USGS  01542500        00060  118867         NaN     1961        12
108      USGS  01542500        00060  118867         NaN     1962         1
109      USGS  01542500        00060  118867         NaN     1962         2
110      USGS  01542500        00060  118867         NaN     1962         3
111      USGS  01542500        00060  118867         NaN     1962         4
112      USGS  01542500        00060  118867         NaN     1962         5
113      USGS  01542500        00060  118867         NaN     1962         6
114      USGS  01542500        00060  118867         NaN     1962         7
115      USGS  01542500        00060  118867         NaN     1962         8
116      USGS  01542500        00060  118867         NaN     1962         9
117      USGS  01542500        00060  118867         NaN     1962        10
118      USGS  01542500        00060  118867         NaN     1962        11
119      USGS  01542500        00060  118867         NaN     1962        12
120      USGS  01542500        00060  118867         NaN     1963         1
121      USGS  01542500        00060  118867         NaN     1963         2
122      USGS  01542500        00060  118867         NaN     1963         3
..        ...       ...          ...     ...         ...      ...       ...
616      USGS  01542500        00060  118867         NaN     2017         6
617      USGS  01542500        00060  118867         NaN     2017         7
618      USGS  01542500        00060  118867         NaN     2017         8
619      USGS  01542500        00060  118867         NaN     2017         9
620      USGS  01542500        00060  118867         NaN     2017        10
621      USGS  01542500        00060  118867         NaN     2017        11
622      USGS  01542500        00060  118867         NaN     2017        12
623      USGS  01542500        00060  118867         NaN     2018         1
624      USGS  01542500        00060  118867         NaN     2018         2
625      USGS  01542500        00060  118867         NaN     2018         3
626      USGS  01542500        00060  118867         NaN     2018         4
627      USGS  01542500        00060  118867         NaN     2018         5
628      USGS  01542500        00060  118867         NaN     2018         6
629      USGS  01542500        00060  118867         NaN     2018         7
630      USGS  01542500        00060  118867         NaN     2018         8
631      USGS  01542500        00060  118867         NaN     2018         9
632      USGS  01542500        00060  118867         NaN     2018        10
633      USGS  01542500        00060  118867         NaN     2018        11
634      USGS  01542500        00060  118867         NaN     2018        12
635      USGS  01542500        00060  118867         NaN     2019         1
636      USGS  01542500        00060  118867         NaN     2019         2
637      USGS  01542500        00060  118867         NaN     2019         3
638      USGS  01542500        00060  118867         NaN     2019         4
639      USGS  01542500        00060  118867         NaN     2019         5
640      USGS  01542500        00060  118867         NaN     2019         6
```

(continues on next page)

(continued from previous page)

```
641      USGS  01542500        00060  118867         NaN     2019         7
642      USGS  01542500        00060  118867         NaN     2019         8
643      USGS  01542500        00060  118867         NaN     2019         9
644      USGS  01542500        00060  118867         NaN     2019        10
645      USGS  01542500        00060  118867         NaN     2019        11

     mean_va  count_nu
93     258.8        31
94     441.1        30
95     280.5        31
96     474.2        31
97    5155.0        28
98    6108.0        31
99    6145.0        30
100   3320.0        31
101   2109.0        30
102   1004.0        31
103   1107.0        31
104    564.7        30
105    286.6        31
106   1576.0        30
107   1457.0        31
108   2391.0        31
109   2387.0        28
110   6124.0        31
111   7340.0        30
112   1699.0        31
113    707.5        30
114    296.1        31
115    271.0        31
116    586.3        30
117    891.4        31
118   1267.0        30
119    704.3        31
120   1666.0        31
121    706.1        28
122   7965.0        31
..       ...       ...
616   3073.0        30
617   1582.0        31
618    810.5        31
619    450.4        30
620    724.0        31
621   2341.0        30
622   1265.0        31
623   2708.0        31
624   7874.0        28
625   3952.0        31
626   4711.0        30
627   4117.0        31
628   3448.0        30
629   2900.0        31
```

(continues on next page)

(continued from previous page)

```
630   2458.0        31
631   7562.0        30
632   4185.0        31
633   5450.0        30
634   4855.0        31
635   3982.0        31
636   5452.0        28
637   3927.0        31
638   4020.0        30
639   4316.0        31
640   3618.0        30
641   1594.0        31
642    855.8        31
643    810.0        30
644    795.1        31
645   1357.0        30

[553 rows x 9 columns]
```

11.7.3 Requesting daily reports

The daily statistics report is different from the monthly and annual reports in that it aggregates multiple years together from across the entire period of record. So in the following example, in line 0, the report provides statistics for January 1st by calculating the mean of every January 1st from 1961 ('begin_yr') to 2019 ('end_yr').

Note that there are 366 rows, or 365 days each year plus Febrary 29th on leap years.

[5]:
```
daily_stats = hf.stats('01542500', 'daily', parameterCd='00060')
daily_stats.df
```

```
Retrieving daily statistics for site #01542500 from https://waterservices.usgs.gov/nwis/
↪stat/?statReportType=daily&statType=all&sites=01542500&format=rdb&parameterCd=00060
```

[5]:
```
    agency_cd   site_no parameter_cd   ts_id  loc_web_ds  month_nu  day_nu  \
0        USGS  01542500        00060  118867         NaN         1       1
1        USGS  01542500        00060  118867         NaN         1       2
2        USGS  01542500        00060  118867         NaN         1       3
3        USGS  01542500        00060  118867         NaN         1       4
4        USGS  01542500        00060  118867         NaN         1       5
5        USGS  01542500        00060  118867         NaN         1       6
6        USGS  01542500        00060  118867         NaN         1       7
7        USGS  01542500        00060  118867         NaN         1       8
8        USGS  01542500        00060  118867         NaN         1       9
9        USGS  01542500        00060  118867         NaN         1      10
10       USGS  01542500        00060  118867         NaN         1      11
11       USGS  01542500        00060  118867         NaN         1      12
12       USGS  01542500        00060  118867         NaN         1      13
13       USGS  01542500        00060  118867         NaN         1      14
14       USGS  01542500        00060  118867         NaN         1      15
15       USGS  01542500        00060  118867         NaN         1      16
16       USGS  01542500        00060  118867         NaN         1      17
17       USGS  01542500        00060  118867         NaN         1      18
```

(continues on next page)

(continued from previous page)

```
18       USGS  01542500        00060  118867         NaN         1      19
19       USGS  01542500        00060  118867         NaN         1      20
20       USGS  01542500        00060  118867         NaN         1      21
21       USGS  01542500        00060  118867         NaN         1      22
22       USGS  01542500        00060  118867         NaN         1      23
23       USGS  01542500        00060  118867         NaN         1      24
24       USGS  01542500        00060  118867         NaN         1      25
25       USGS  01542500        00060  118867         NaN         1      26
26       USGS  01542500        00060  118867         NaN         1      27
27       USGS  01542500        00060  118867         NaN         1      28
28       USGS  01542500        00060  118867         NaN         1      29
29       USGS  01542500        00060  118867         NaN         1      30
..        ...       ...          ...     ...         ...       ...     ...
336      USGS  01542500        00060  118867         NaN        12       2
337      USGS  01542500        00060  118867         NaN        12       3
338      USGS  01542500        00060  118867         NaN        12       4
339      USGS  01542500        00060  118867         NaN        12       5
340      USGS  01542500        00060  118867         NaN        12       6
341      USGS  01542500        00060  118867         NaN        12       7
342      USGS  01542500        00060  118867         NaN        12       8
343      USGS  01542500        00060  118867         NaN        12       9
344      USGS  01542500        00060  118867         NaN        12      10
345      USGS  01542500        00060  118867         NaN        12      11
346      USGS  01542500        00060  118867         NaN        12      12
347      USGS  01542500        00060  118867         NaN        12      13
348      USGS  01542500        00060  118867         NaN        12      14
349      USGS  01542500        00060  118867         NaN        12      15
350      USGS  01542500        00060  118867         NaN        12      16
351      USGS  01542500        00060  118867         NaN        12      17
352      USGS  01542500        00060  118867         NaN        12      18
353      USGS  01542500        00060  118867         NaN        12      19
354      USGS  01542500        00060  118867         NaN        12      20
355      USGS  01542500        00060  118867         NaN        12      21
356      USGS  01542500        00060  118867         NaN        12      22
357      USGS  01542500        00060  118867         NaN        12      23
358      USGS  01542500        00060  118867         NaN        12      24
359      USGS  01542500        00060  118867         NaN        12      25
360      USGS  01542500        00060  118867         NaN        12      26
361      USGS  01542500        00060  118867         NaN        12      27
362      USGS  01542500        00060  118867         NaN        12      28
363      USGS  01542500        00060  118867         NaN        12      29
364      USGS  01542500        00060  118867         NaN        12      30
365      USGS  01542500        00060  118867         NaN        12      31

     begin_yr  end_yr  count_nu  ...  mean_va  p05_va  p10_va  p20_va  p25_va  \
0        1961    2019        46  ...     2850   520.0     807     970    1080
1        1961    2019        46  ...     2950   487.0     800    1040    1100
2        1961    2019        46  ...     2880   523.0     793    1110    1290
3        1961    2019        46  ...     2720   541.0     753    1120    1280
4        1961    2019        46  ...     2710   551.0     716    1080    1180
5        1961    2019        46  ...     2850   556.0     802    1040    1120
6        1961    2019        46  ...     2780   576.0     819    1020    1100
```

(continues on next page)

(continued from previous page)

```
7         1961   2019       46  ...   2630  603.0     792     976    1100
8         1961   2019       46  ...   2710  621.0     788     929    1130
9         1961   2019       46  ...   2560  696.0     847     961    1070
10        1961   2019       46  ...   2430  738.0     829     979    1100
11        1961   2019       46  ...   2450  649.0     773    1070    1240
12        1961   2019       46  ...   2750  621.0     771    1000    1170
13        1961   2019       46  ...   2760  607.0     778    1000    1190
14        1961   2019       46  ...   2640  614.0     740     974    1080
15        1961   2019       46  ...   2480  601.0     724     938    1040
16        1961   2019       46  ...   2420  587.0     716     860     948
17        1961   2019       46  ...   2380  554.0     715     893     988
18        1961   2019       46  ...   2390  564.0     687     964    1120
19        1961   2019       46  ...   2340  570.0     678     962    1090
20        1961   2019       46  ...   2340  544.0     655     894    1070
21        1961   2019       46  ...   2330  581.0     662     952    1100
22        1961   2019       46  ...   2250  572.0     662     991    1070
23        1961   2019       46  ...   2340  544.0     667    1000    1000
24        1961   2019       46  ...   2780  521.0     716     989    1040
25        1961   2019       46  ...   2950  492.0     772     993    1100
26        1961   2019       46  ...   3040  496.0     757    1070    1090
27        1961   2019       46  ...   2960  510.0     766     999    1050
28        1961   2019       46  ...   2750  497.0     791     928     956
29        1961   2019       46  ...   2880  487.0     800     888     943
..         ...    ...      ...  ...    ...    ...     ...     ...     ...
336       1961   2020       47  ...   3010  375.0     784    1130    1180
337       1961   2020       47  ...   2810  415.0     782    1090    1170
338       1961   2020       47  ...   2720  401.0     745     994    1070
339       1961   2020       47  ...   2670  517.0     757     967    1070
340       1961   2020       47  ...   2650  708.0     871     994    1310
341       1961   2020       47  ...   2810  713.0     812    1070    1270
342       1961   2020       47  ...   2790  535.0     730    1050    1310
343       1961   2020       47  ...   2800  483.0     690    1130    1370
344       1961   2020       47  ...   2850  407.0     718    1120    1320
345       1961   2020       47  ...   2980  444.0     673    1320    1650
346       1961   2020       47  ...   2890  624.0     935    1330    1400
347       1961   2019       46  ...   2880  637.0     906    1190    1380
348       1961   2019       46  ...   2750  695.0     893    1200    1340
349       1961   2019       46  ...   2670  684.0     770    1100    1270
350       1961   2019       46  ...   2610  531.0     766    1030    1270
351       1961   2019       46  ...   2600  530.0     707    1020    1190
352       1961   2019       46  ...   2510  580.0     669    1030    1180
353       1961   2019       46  ...   2670  611.0     660    1020    1210
354       1961   2019       46  ...   2680  576.0     662    1050    1200
355       1961   2019       46  ...   2690  535.0     683    1000    1200
356       1961   2019       46  ...   3020  587.0     682     959    1230
357       1961   2019       46  ...   2990  607.0     718    1110    1450
358       1961   2019       46  ...   3060  607.0     660    1250    1310
359       1961   2019       46  ...   3160  601.0     718    1140    1340
360       1961   2019       46  ...   3040  544.0     770    1130    1380
361       1961   2019       46  ...   3030  499.0     720    1040    1280
362       1961   2019       46  ...   3060  525.0     658    1050    1200
363       1961   2019       46  ...   2980  528.0     727    1030    1100
```

(continues on next page)

(continued from previous page)

```
364      1961    2019        46  ...     2830   514.0     680     961    1080
365      1961    2019        46  ...     2890   533.0     706     924    1110

     p50_va  p75_va  p80_va  p90_va   p95_va
0      2180    3790    4340    6800   7600.0
1      2080    3730    3960    6400   9460.0
2      2110    3400    4590    6450   8390.0
3      1890    3280    4190    6070   7510.0
4      1850    3880    4480    5760   8090.0
5      2140    3820    4420    6160   7460.0
6      1900    3650    4060    5500   6930.0
7      1770    3700    4050    4870   6090.0
8      1920    3620    3720    5280   8450.0
9      1720    3320    3700    5370   7960.0
10     1660    3340    3780    5170   6370.0
11     1750    3150    3480    4620   6550.0
12     2020    3690    4130    6160   7620.0
13     1900    3780    3940    6600   7040.0
14     1900    3520    4050    5850   7560.0
15     1790    3250    3340    4630   7750.0
16     2040    2910    3440    5160   6920.0
17     1780    2950    3630    6000   7630.0
18     1610    2760    3540    5490   7260.0
19     1740    3160    3470    4700   6780.0
20     1690    2960    3270    5060   6480.0
21     1470    2830    3080    4750   7770.0
22     1640    2870    3260    3980   6860.0
23     1510    3150    3550    4870   5960.0
24     1620    3860    4550    6270   8750.0
25     1740    3830    4130    6700   8240.0
26     1860    3900    5470    7430   7980.0
27     2050    4180    4730    6530   9460.0
28     2210    3880    4370    5970   8140.0
29     1930    3780    4430    6130   7100.0
..      ...     ...     ...     ...      ...
336    1880    3710    4400    6840   9630.0
337    2030    3750    4040    6500   7330.0
338    1930    3720    4370    5820   6970.0
339    1830    3850    4020    5980   6920.0
340    1800    3570    4180    5590   7450.0
341    2140    3770    4210    5330   7230.0
342    2220    3900    4150    4650   9390.0
343    2140    3480    3890    5640   9520.0
344    2170    3720    4320    5630   8720.0
345    2440    3680    4130    6360   8870.0
346    2430    4000    4360    6440   7060.0
347    2440    4350    4580    5490   6460.0
348    2430    3750    4240    4880   7310.0
349    2120    3760    4100    5020   7450.0
350    2110    3710    3950    4830   6140.0
351    2230    3510    4100    4740   6160.0
352    2250    3630    3870    4540   5150.0
```

(continues on next page)

(continued from previous page)

```
353    2120    3260    3440    4390  10400.0
354    2000    3160    3510    5740  10000.0
355    1900    3820    4480    5980   8600.0
356    1890    5000    5810    7070   8830.0
357    1920    4160    4540    7530   8910.0
358    1950    3830    4820    7250   8070.0
359    2530    4370    4800    6590   8800.0
360    2660    4280    4730    5840   8060.0
361    2600    4100    4610    5810   8770.0
362    2530    4500    4570    5800   8950.0
363    2560    4130    4480    6180   7660.0
364    2440    3820    4390    6130   7520.0
365    2270    4130    4550    6050   7770.0

[366 rows x 24 columns]
```

CHAPTER

TWELVE

GUIDE TO CONTRIBUTING CODE

Contributions are always welcome and greatly appreciated!

You can contribute in many ways:

- Reporting bugs, suggesting features, providing feedback: Use the issues page.
- Adding documentation: add a new notebook, fix a typo, improve the docstrings, mention us in a blog post…
- Adding features, fixing bugs, writing tests: I'll respond to your pull request quickly! Details below!

12.1 Submitting a pull request

- Start by forking mroberge/hydrofunctions into your own GitHub account.
- Working from your own account, move to the Develop branch to see the current state of the project.
- When you are ready to make changes, create a new branch off of Develop with a short, boring, yet descriptive name, like "bug-nwis-parsing" or "feature-metric-units".
- Keep your commits small and focused: deal with just one issue at a time.
- Use helpful comments on each commit. Refer to an issue number if possible.
- Try to sync your local commits with GitHub at least once a day. People are curious!
- **Submit your Pull Request early, while you are just starting to get started! This will:**
  - open up a forum in GitHub where people can add comments
  - set up a checklist of things to get done
  - enables automatic unit testing.
  - all your subsequent commits get added to the pull request automatically.
- Don't apologize for mistakes or not being done yet!

12.2 Standards for the ideal pull request

"We are all in the gutter, but some of us are looking at the stars." -Oscar Wilde

- If you make a change, add your name to AUTHORS.rst
- Note your change in HISTORY.rst and initial it.
- Use docstrings, illustrate features in a notebook, add a section to docs/usage.rst
- I use 'Black' for code formatting. Follow PEP8 standards!
- Use Google-style docstrings , described here.
- Add tests! Lots of tests! Make sure you test your code!
- Your code should work for Python 3.6, 3.7, 3.8, 3.9, & 3.10. This gets tested by the CI

12.3 A detailed guide to contributing new code

Ready to contribute? Here's how to set up *hydrofunctions* for local development.

1. Fork the hydrofunctions repo on GitHub by clicking the 'Fork' button near the upper right corner.
2. Open the GitHub page for your fork, and clone it to your local computer where you will be doing your coding. To clone your fork, I like to use GitHub's Desktop tool, which does a great job of putting most of Git's best features into an easy-to-use GUI.

   Download GitHub Desktop.

   There are two ways to clone your files to your local computer:

   - Starting from the GitHub page for your fork: Select the 'Code' button and select 'Open with GitHub Desktop'
   - or, if you have GitHub Desktop open: File > clone repository…

   These options will set up a local folder on your computer where you can use git while editing your version of hydrofunctions.
3. Install Anaconda 3 if you don't already have it on your system. This includes the latest version of Python, and a package manager, **conda**, which is an alternative to pip that also manages virtual environments, replacing venv, and virtualenv.:

   Download Anaconda: https://www.continuum.io/downloads
4. Much of the rest involves the command line. In the following examples I'll be using Windows, but this will only affect simple commands like making a new directory, or changing directory. The important commands are the same on different platforms. Also, if you are using Windows, use **cmd.exe** as your command line instead of PowerShell, which seems to interfere with one of our tools, conda.
5. Create a new conda environment named *my39env*. Include everything we need: python 3.9, git, and the anaconda set of scientific packages:

   ```
   > conda create -n my36env python=3.9 anaconda git
   ```
6. List all of your available environments and activate my36env. The active environment will have a star next to it:

   ```
   > conda info -e
   > activate my39env
   ```

7. For kicks, check that you've got the right version of python running, and list all of the packages that you have available to you in this environment:

```
> python --version
> conda list
```

8. Now we are going to install hydrofunctions using the file from **your** forked version, instead of the standard version from PyPI. To do this, first move into the directory where you had GitHub Desktop put your clone:

```
> cd GitHub/hydrofunctions
```

9. We'll use pip to install your files in 'develop' mode using the '-e' flag. Now you can edit the source files and have the edits freshly interpreted again when you *import hydrofunctions* during a python session. The [dev] part tells pip to install all of the extra requirements that you'll need as a developer:

```
> pip install -e .[dev]
```

10. Run the automatic tests to make sure everything is hunky-dory:

```
> pytest
```

11. Before you start improving hydrofunctions with all your fantastic changes, create a new branch. Give it a simple name that explains what your change adds:

```
> git checkout -b name-of-your-bugfix-or-feature
```

Alternatively, use GitHub's Desktop tool:

- Branch > New branch…

12. Go ahead and make changes to the files now. I like to use VS Code or Spyder, which you installed already with anaconda:

```
> spyder
```

13. After you've made a small change, make sure you didn't break anything by running the tests again. I find it easiest to run the tests from the command line inside the hydrofunctions directory:

```
> pytest
```

14. Before you make too many changes, 'commit' what you've done. Ideally, each group of changes that you put into a commit will be logically related to each other, and the group of changes will be really small. Make sure that you explain your changes in the commit message. Use GitHub Desktop. If you use the command line, then type:

```
> git add .
> git commit -m "Your detailed description of your changes."
```

15. When you are done commiting changes, push your branch and all of the commits in it to GitHub. This can be done with the 'Sync' button in the upper right corner of Desktop, or use the command line:

```
> git push origin name-of-your-bugfix-or-feature
```

16. Finally, submit a pull request to me through the GitHub website. Your branch doesn't need to be done to submit- this just warns people that you exist and prevents duplication.

12.4 The non-conda version

Caveat emptor: I haven't tested the following steps recently!

1. Fork the *hydrofunctions* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/hydrofunctions.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv hydrofunctions
$ cd hydrofunctions/
$ pip install -e .[dev]
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 hydrofunctions tests
$ pytest
```

   or `$ python -m unittest -v`

   then:

```
$ tox
```

   To get flake8 and tox, just pip install them into your virtualenv.:

```
$ pip install flake8
$ pip install tox
```

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

12.5 Tips

- The Spyder IDE will highlight bad code formatting if you turn this feature on: Tools > Preferences > Code Introspection/Analysis > Real-time code style analysis
- Test out your .rst files using the Online reStructuredText editor
- To run a subset of tests, like the file *test_hydrofunctions.py*:

```
$ python -m unittest tests.test_hydrofunctions
```

CHAPTER

THIRTEEN

AUTHORS

13.1 Development Lead

- Martin Roberge <mroberge@towson.edu>

13.2 Contributors

- Joseph Hughes <jdhughes@usgs.gov>
- Molly Pickel <mpicke8@students.towson.edu>
- Nathan Korinek <Nathan.Korinek@colorado.edu>
- Ibrahim El Merehbi <https://github.com/elmerehbi>
- James McCreight <https://github.com/jmccreight>

CHAPTER

FOURTEEN

HISTORY

14.1 0.1.0 (2016-08-11)

- Project set up.

14.2 0.1.1 (2016-08-11)

- Customized project boilerplate.

14.3 0.1.2 (2016-08-26)

- Set up template for custom exceptions
- Added get_nwis function

14.4 0.1.3 (2016-09-09)

- Check user inputs & raise explanatory exceptions
- Extract data from response into a dataframe.
- Stations object for managing data.

14.5 0.1.4 (2016-09-18)

- Added tests & documentation.

14.6 0.1.5 (2018-02-22)

- Updated to support Python 3.6
- Updated docs, added notebooks (mcr jdh)
- Added parameterCd to allow requests for different datasets (thanks @jdhughes-usgs!)
- Added ability to query sites by state or county (jdh)
- Added ability to request lists of sites or counties (jdh)
- Improved column names: now includes site id & variable description (for example, '07228000 - Mean Discharge, cubic feet per second')(jdh)
- Added descriptive warnings to explain why queries fail (mcr)

14.7 0.1.6 (2018-03-07)

- Added draw_map function for help selecting sites (mcr)
- Added ability to query sites by a bounding box (jdh)
- Revised pandas dataframe column names to be the name composed of the provider, site id, parameter, and statistic (for example. USGS:01638500:00060:0001). (jdh)
- Added qualifiers for each station as a column in the dataframe. (jdh)
- Added ability extract NWIS property data from the response object (for example, siteName). (jdh)

14.8 0.1.7 (2018-11-11)

- Added a flow duration chart (mcr)
- Added the cycleplot chart (mcr)
- Added code coverage (mcr)
- Improved the build scripts for TravisCI (mcr)
- Updated to support Python 3.7

14.9 0.2.0 (2020-06-19 Juneteenth)

- **NWIS has a simpler interface and improved functionality:**
  - No need to use .get_data(); data is fetched automatically.
  - **NWIS.df() creates dataframes using only the parts you want**
    - .df('discharge') returns a dataframe with only discharge data
    - .df('01585200') returns all of the data for just this site
    - .df('flags') returns a dataframe with the qualifier flags.
  - New and improved REPR: lists stations, parameters, and frequency for a dataset.
- **Saving data to a file:**

  - the 'file' parameter for NWIS allows you to save your data locally
  - If the file doesn't exist, NWIS requests the data and creates the file
  - Uses the parquet format for faster load times and smaller file sizes
- **Improved parsing of data from NWIS:**
  - missing observations are noted & can be replaced with interpolated values
  - duplicates found & removed
  - unsorted data found & cleaned.
  - different frequencies raise a warning when resampled
- parameterCd now accepts multiple parameters in request.
- If parameterCd is not specified, then all available parameters will be requested (default).
- hf.rating_curve(site) retrieves the current rating curve for a USGS site.
- hf.peaks(site) retrieves the annual peak discharges for a USGS site.
- hf.field_meas(site) retrieves the field data and notes used by the USGS to create a rating curve.
- hf.stats(site, statReportType) retrieves Annual, Monthly, or Daily reports from the USGS.
- hf.site_file(site) retrieves expanded site file.
- hf.data_catalog(site) retrieves history of data collected at site.
- Dropped Python 3.4 & 3.5 support, added 3.8.

14.10 0.2.1 (2021-05-28)

- Moved from TravisCI to Github Actions.
- Added Python 3.9 support.
- Created a verbose mode that is on by default.
- Added the flood stage service from the USGS waterwatch (iem)
- Save & read NWIS data in json.gzip files
- Added ability to read multiple instrument methods at the same time
- Bugfix: divide by zero error when you request data from many sites
- Bugfix: save_parquet

14.11 0.2.2 (2021-11-06)

- Added documentation for the *site_file* and *data_catalog* services
- Added "What is Hysteresis?" example to User's Guide
- Minor feature: URL message will print before & after requests are made
- Added verbose mode to USGS RDB functions
- Bugfix: *hf.df('q')* returned non-discharge data; fixed & added tests & fixture (dgk)
- Bugfix: renamed *typing.py* module to *validate.py* to prevent interference with typing

- Bugfix: HF will raise HydroNoDataError when non-200 response comes back from USGS
- Bugfix: logging off by default.
- Added hf._start_logging() to create a log & start logging.
- HydroExceptions will now create a log message when raised.

14.12 0.2.3 (2022-04-18)

- Moved package requirements & dev requirements to setup.cfg
- Create new 'dev' extra_requirements. Install from inside hydrofunctions: *pip install -e .[dev]*
- Changed default behavior of NWIS & extract_nwis_df to NOT fill missing values by interpolation.

14.13 0.2.4 (2023-06-14)

- Remove support for Python 3.6 & 3.7; add Python 3.10 & 3.11
- Update to Pandas 2+ (jm)
- Update documentation to reflect gauge changes

CHAPTER

FIFTEEN

HYDROFUNCTIONS 0.2.4 PACKAGE

15.1 Package contents

15.1.1 Hydrofunctions

Hydrofunctions is a suite of convenience functions to help you explore hydrology data interactively.

Basic Usage:

```
>>> import hydrofunctions as hf

>>> site = '01589440'
>>> jones = hf.NWIS(site, 'iv', period='P10D')
Requested data from https://waterservices.usgs.gov/nwis/iv/?format=json%2C1.1&
↪sites=01589440&period=P10D
```

Examine the dataset:

```
>>> jones
USGS:01589440: JONES FALLS AT SORRENTO, MD
    00060: <15 * Minutes> Discharge, cubic feet per second
    00065: <15 * Minutes> Gage height, feet
Start: 2022-10-27 17:30:00+00:00
End:   2022-11-06 17:15:00+00:00
```

The listing reports each of the parameters collected at the site that was requested, how frequently the data are collected, and the name of the parameter written out with units. The start and end of the dataset are given in Universal Time (UTC).

View the first five rows of a dataframe that only contains the discharge data:

```
>>> jones.df('discharge').head()
                           USGS:01589440:00060:00000
datetimeUTC
2022-10-27 17:30:00+00:00                       14.6
2022-10-27 17:45:00+00:00                       15.2
2022-10-27 18:00:00+00:00                       15.2
2022-10-27 18:15:00+00:00                       15.8
2022-10-27 18:30:00+00:00                       16.4
```

Because the .df() method returns a dataframe, you have access to all of the methods associated with Pandas, including .plot(), .describe(), and .info() !

Learn more about hydrofunctions and the NWIS object with help():

```
>>> help(hf)
>>> help(hf.NWIS)
```

Read more about Hydrofunctions here: https://hydrofunctions.readthedocs.io/

15.2 hydrofunctions.charts module

15.2.1 hydrofunctions.charts

This module contains charting functions for Hydrofunctions.

`hydrofunctions.charts.`**`cycleplot`**(*Qseries*, *cycle='diurnal'*, *compare=None*, *y_label='Discharge (ft³/s)'*, *legend=True*, *legend_loc='best'*, *title=''*)

Creates a chart to illustrate annual and diurnal cycles.

This chart will use the pandas groupby method to plot the mean and median values for a time-indexed dataframe. It helps you identify diurnal patterns by plotting the mean and median values over 24 hours for a diurnal pattern, and over a year for annual patterns.

This function will also use the 'compare' argument to create a series of charts to compare how well these cycles appear in different groups. For example, is the diurnal cycle more visible in December versus June? In this case, you would use:

```
hf.cycleplot(myDataFrame, cycle='diurnal', compare = 'month')
```

This will produce twelve charts, each covering 24 hours. A line will represent the mean values over 24 hours, another line represents the median, and two grey stripes represent the 0.4 to 0.6 quantile, and the 0.2 to 0.8 quantile range.

Parameters

- **`Qseries`** (*`series`*) – a Pandas series of discharge values.
  - Values should be arranged in columns
  - Should use a dateTimeIndex
- **`cycle`** (*`str`*) – The period of the cycle to be illustrated, along with the method for binning. The options are:
  - diurnal (default): plots the values for a 24 hour cycle.
  - diurnal-smallest: uses the smallest increment of time available to bin the time units for a 24 hour cycle.
  - diurnal-hour: uses hours to bin measurements for a 24-hour cycle.
  - annual: plots values into a 365 day cycle.
  - annual-day: the annual cycle using 365 day-long bins.
  - annual-week: the annual cycle using 52 week-long bins.
  - annual-month: the annual cycle using 12 month-long bins.

- weekly: a 7-day cycle using seven 24-hour long bins. Note that unlike the others, this is not a natural cycle, and is likely has anthropogenic origins.

- **compare** (*str*) – The system for splitting the data into groups for a set of comparison charts.
  - None (default): No comparison will be made; only one chart.
  - month: twelve plots will be produced, one for each month.
  - weekday: seven plots will be produced, one for each day of the week.
  - weekend: two plots will be produced, one for the five weekdays, one for Saturday and Sunday.
  - night: two plots will be produced, one for night (6pm to 6am), one for day (6am to 6pm).
- **y_label** (*str*) – The label for the y axis.
- **legend** (*bool*) – default True. Whether the legend should be plotted.
- **legend_loc** (*str*) – default is 'best'. The location of the legend.
  - 'best': Automatically choose the option below with the least overlap.
  - 'upper left', 'upper right', 'lower left', 'lower right': place the legend at the corresponding corner of the axes/figure.
  - 'upper center', 'lower center', 'center left', 'center right': place the legend at the center of the corresponding edge of the axes/figure.
  - 'center': place the legend at the center of the axes/figure.
  - The location can also be a 2-tuple giving the coordinates of the lower-left corner of the legend in axes coordinates.
- **title** (*str*) – default is ''. Text to use as a figure title. If no text is provided, no title will be created (default).

Returns

Returns a tuple that includes a matplotlib 'figure' and 'axes'. The figure is a container with all of the drawing inside of it; the axes are an array of matplotlib charts. Together, they will plot immediately in a Jupyter notebook if the command *%matplotlib inline* was previously issued. The figure and axes may be altered after they are returned.

Return type

fig, ax (matplotlib.figure.Figure, matplotlib.axes.Axes)

Note: inspired by https://jakevdp.github.io/PythonDataScienceHandbook/03.11-working-with-time-series.html Jake VanderPlas. 2016. Python Data Science Handbook. O'Reilly Media, Inc.

hydrofunctions.charts.**flow_duration**(*Qdf*, *xscale='logit'*, *yscale='log'*, *ylabel='Stream Discharge (m³/s)'*, *symbol='.'*, *legend=True*, *legend_loc='best'*, *title=''*)

Creates a flow duration chart from a dataframe of discharges.

Parameters

- **Qdf** (*dataframe*) – a dataframe of discharge values.
  - Values should be arranged in columns
  - No sorting necessary
  - Rows do not need an index

  - If more than one column, each column will be added as a separate color to the chart.
  - Only include columns with discharge values; no metadata
- **xscale** (*str, 'logit' | 'linear'*) – Type of x scale for plotting probabilities default is 'logit', so that each standard deviation is nearly the same distance on the x scale. 'linear' is the other option.
- **yscale** (*str, 'log' | 'linear'*) – The type of y scale for plotting discharge. Default is 'log'.
- **ylabel** (*str*, default *'Stream Discharge (ft^3/s)'*) – The label for the Y axis.
- **xlabel** (*not implemented*) –
- **symbol** (*str, '.' | ','*) – formatting symbol for points.
  - point: '.' (default)
  - pixel point: ','
  - circle: 'o'
  - triangle up: '^'

  See https://matplotlib.org/api/markers_api.html for full list of point formatters.
- **legend** (*bool*, default *True*) – Whether the legend should be plotted.
- **legend_loc** (*str*, default *best*) – the location of the legend.
  - 'best': Automatically choose the option below with the least overlap.
  - 'upper left', 'upper right', 'lower left', 'lower right': place the legend at the corresponding corner of the axes/figure.
  - 'upper center', 'lower center', 'center left', 'center right': place the legend at the center of the corresponding edge of the axes/figure.
  - 'center': place the legend at the center of the axes/figure.
  - The location can also be a 2-tuple giving the coordinates of the lower-left corner of the legend in axes coordinates.
- **title** (*str*, default '') – Text to use as a figure title. If no text is provided, no title will be created (default).

Returns

Returns a tuple that includes a matplotlib 'figure' and 'axes'. The figure is a container with all of the drawing inside of it; the axes are an array of matplotlib charts. Together, they will plot immediately in a Jupyter notebook if the command *%matplotlib inline* was previously issued. The figure and axes may be altered after they are returned.

Return type

fig, ax (*matplotlib.figure.Figure*, *matplotlib.axes.Axes*)

15.3 hydrofunctions.exceptions module

15.3.1 hydrofunctions.exceptions

This module contains all of the custom exceptions defined in this package. The base class is HydroException and all custom exceptions are subclasses of HydroException.

Use the errors like this:

```
try:
    #some code here that might return no data
    #more code that might get encoded improperly
except HydroNoDataError('This site has no data'):
    # handle error here.
except HydroEncodeError():
    # handle this error here.
else:
    # code to complete if there is no exception raised.
finally:
    # code that you want to run whether an exception is raised or not.
    # If an exception wasn't caught, then this code gets run, and the
    # exception gets re-raised after this finally clause gets run.
```

Keep the try clause short: if you put too many things in there, it can be difficult to figure out what broke. On the other hand, like in my example above, it is more readable if you group a series of statements and then handle their exceptions together.

Example:

```
>>> raise HydroNoDataError("Oh no, NWIS doesn't have this data for you!")
```

https://axialcorps.com/2013/08/29/5-simple-rules-for-building-great-python-packages/

---

exception hydrofunctions.exceptions.**HydroEncodeError**(*msg=''*)

Bases: *HydroException*

Raised when an error occurs while encoding or decoding an argument.

Example:

```
try:
    # bunch of code from your package
except HydroException:
    # blanked condition to handle all errors from your package
```

exception hydrofunctions.exceptions.**HydroException**(*msg=''*)

Bases: Exception

This is the base class for all exceptions created for the HydroFunctions package. This class is not meant to be raised.

exception hydrofunctions.exceptions.**HydroNoDataError**(*msg=''*)

Bases: *HydroException*

Raised when a service returns an empty dataset or indicates that it has no data for the request.

Usage:

```
raise HydroNoDataError("The NWIS service had no data for this request.")
```

Do not catch this error for interactive sessions: The user should get a useful message from the error when they try to request something that doesn't exist.

Catch this error in automated systems so that the system can reconsider the request and either fix the request or move on to the next request.

Example:

```
try:
    hf.NWIS('666666666')
except HydroNoDataError as err:
    print("This is just to illustrate how to capture this error.")
    print(err)
```

exception hydrofunctions.exceptions.**HydroUserWarning**(*msg=''*)

Bases: UserWarning

Warn user of a hazardous condition or when an action has been triggered that may be unexpected.

This is the base class for all warnings created for the HydroFunctions package. This class can be used if there is no more specific warning available.

Usage:

```
import hydrofunctions as hf
import warnings
... code
warnings.warn('This is my warning message.', hf.HydroUserWarning)
```

Note: Warnings can be hidden or turned off depending on how the user is accessing Python and the settings for their interface.

Use HydroException if a process must be shut down, or is doomed to fail anyway. This will at least give the user a helpful error message.

15.4 hydrofunctions.helpers module

15.4.1 hydrofunctions.helpers

This module holds functions designed to help out the user in an IPython session.

---

hydrofunctions.helpers.**count_number_of_truthy**(*my_list*)

hydrofunctions.helpers.**draw_map**(*width=700*, *height=400*, *url='http://hydrocloud.org'*)

Draws a map of stream gages in a Jupyter Notebook.

This function will draw an interactive map of stream gages from hydrocloud.org into an iframe and display it in a Jupyter Notebook. Each dot represents a stream gage. Click on the dot to learn its name, which you can use to request data.

Parameters

- **width** (*int*) – The width of the map iframe.
- **height** (*int*) – The height of the map iframe.
- **url** (*str*) – The URL to put inside of the IFrame. Defaults to https://hydrocloud.org

Returns
HTML display object.

Example:

```
>>> import hydrofunctions as hf
>>> hf.draw_map()
```

A map appears.

```
>>> hf.draw_map(width=900, height=600)
```

Draws a larger map.

15.5 hydrofunctions.hydrofunctions module

15.5.1 hydrofunctions.hydrofunctions

This module contains the main functions used in an interactive session.

---

hydrofunctions.hydrofunctions.**calc_freq**(*index*)

hydrofunctions.hydrofunctions.**extract_nwis_df**(*nwis_dict*, *interpolate=False*)

Returns a Pandas dataframe and a metadata dict from the NWIS response object or the json dict of the response.

Parameters

- **nwis_dict** (*obj*) – the json from a response object as returned by get_nwis().json(). Alternatively, you may supply the response object itself.
- **interpolate** (*bool*) – fill missing data values with interpolated values. Default False.

Returns
a pandas dataframe.

Raises

- *HydroNoDataError* – when the request is valid, but NWIS has no data for the parameters provided in the request.
- *HydroUserWarning* – when one dataset is sampled at a lower frequency than another dataset in the same request.

hydrofunctions.hydrofunctions.**get_nwis**(*site*, *service='dv'*, *start_date=None*, *end_date=None*, *stateCd=None*, *countyCd=None*, *bBox=None*, *parameterCd='all'*, *period=None*, *verbose=True*)

Request stream gauge data from the USGS NWIS.

Parameters

- **site** (*str or list of strings*) – a valid site is '01585200' or ['01585200', '01646502']. site should be *None* if stateCd or countyCd are not *None*.
- **service** (*str*) –

  can either be 'iv' or 'dv' for instantaneous or daily data.

  – 'dv'(default): daily values. Mean value for an entire day.

  – 'iv': instantaneous value measured at this time. Also known as 'Real-time data'. Can be measured as often as every five minutes by the USGS. 15 minutes is more typical.
- **start_date** (*str*) – should take on the form yyyy-mm-dd
- **end_date** (*str*) – should take on the form yyyy-mm-dd
- **stateCd** (*str*) – a valid two-letter state postal abbreviation. Default is *None*.
- **countyCd** (*str or list of strings*) – a valid county abbreviation. Default is *None*.
- **bBox** (*str, list, or tuple*) –

  a set of coordinates that defines a bounding box.

  – Coordinates are in decimal degrees

  – Longitude values are negative (west of the prime meridian).

  – Latitude values are positive (north of the equator).

  – comma-delimited, no spaces, if provided as a string.

  – The order of the boundaries should be: "West,South,East,North"

  – Example: "-83.000000,36.500000,-81.000000,38.500000"
- **parameterCd** (*str or list of strings*) –

  NWIS parameter code. Usually a five digit code. Default is 'all'. A valid code can also be given as a list: `parameterCd=['00060','00065']`

  – if value of 'all' is submitted, then NWIS will return every parameter collected at this site. (default option)

  – stage: '00065'

  – discharge: '00060'

  – not all sites collect all parameters!

  – See https://nwis.waterdata.usgs.gov/usa/nwis/pmcodes for full list
- **period** (*str*) –

  NWIS period code. Default is *None*.

  – Format is "PxxD", where xx is the number of days before today.

  – Either use start_date or period, but not both.
- **verbose** (*bool*) – If True (default); will print confirmation messages with the url before and after the request is made.

Returns

a response object. This function will always return the response,
even if the NWIS returns a status_code that indicates a problem.

- response.url: the url we used to request data

- response.json: the content translated as json
- **response.status_code: the internet status code**
  - '200': is a good request
  - non-200 codes will be reported as a warning.
  - '400': is a 'Bad Request'– the parameters did not make sense
  - see <https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html> for more codes and meaning.
- response.ok: *True* when we get a '200' status_code

Raises

- **ConnectionError** – due to connection problems like refused connection or DNS Error.
- **SyntaxWarning** – when NWIS returns a response code that is not 200.

Example:

```
>>> import hydrofunctions as hf
>>> response = hf.get_nwis('01585200', 'dv', '2012-06-01', '2012-07-01')
```

```
>>> response
<response [200]>
```

```
>>> response.json()
*JSON ensues*
```

```
>>> hf.extract_nwis_df(response)
*a Pandas dataframe appears*
```

Other Valid Ways to Make a Request:

```
>>> sites = ['07180500', '03380475', '06926000'] # Request a list of sites.
>>> service = 'iv'  # Request real-time data
>>> days = 'P10D'  # Request the last 10 days.
>>> stage = '00065' # Sites that collect discharge usually collect water depth too.
>>> response2 = hf.get_nwis(sites, service, period=days, parameterCd=stage)
```

Request Data By Location:

```
>>> # Request the most recent daily data for every site in Maine
>>> response3 = hf.get_nwis(None, 'dv', stateCd='ME')
>>> response3
<Response [200]>
```

The specification for the USGS NWIS IV service is located here: http://waterservices.usgs.gov/rest/IV-Service.html

hydrofunctions.hydrofunctions.**get_nwis_property**(*nwis_dict*, *key=None*, *remove_duplicates=False*)

Returns a list containing property data from an NWIS response object.

Parameters

- **nwis_dict** (*dict*) – the json returned in a response object as produced by `get_nwis().json()`.

- **key** (*str*) –

  a valid NWIS response property key. Default is *None*. The index is returned if key is *None*. Valid keys are:

  - None
  - name - constructed name "provider:site:parameterCd:statistic"
  - siteName
  - siteCode
  - timeZoneInfo
  - geoLocation
  - siteType
  - siteProperty
  - variableCode
  - variableName
  - variableDescription
  - valueType
  - unit
  - options
  - noDataValue
- **remove_duplicates** (*bool*) – a flag used to remove duplicate values in the returned list.

Returns
a list with the data for the passed key string.

Raises

- *HydroNoDataError* – when the request is valid, but NWIS has no data for the parameters provided in the request.
- **ValueError when the key is not available.** –

hydrofunctions.hydrofunctions.**nwis_custom_status_codes**(*response*)

Raise custom warning messages from the NWIS when it returns a status_code that is not 200.

Parameters
response – a response object as returned by get_nwis().

Returns
None if response.status_code == 200

Raises
HydroNoDataError – when a non-200 status code is returned. https://en.wikipedia.org/wiki/List_of_HTTP_status_codes

---

Note:

NWIS status_code messages come from:
https://waterservices.usgs.gov/docs/portable_code.html

Additional status code documentation:
https://waterservices.usgs.gov/rest/IV-Service.html#Error

`hydrofunctions.hydrofunctions.`**`read_json_gzip`**`(`*filename*`)`

Read a gzipped JSON file into a Python dictionary.

Reads JSON files that have been zipped and returns a Python dictionary. Usually the files should have an extension .json.gz Hydrofunctions uses this function to store the original JSON format WaterML response from the USGS NWIS.

Parameters
`filename` (*`str`*) – A string with the filename and extension.

Returns
a dictionary of the file contents.

`hydrofunctions.hydrofunctions.`**`read_parquet`**`(`*filename*`)`

Read a hydrofunctions parquet file.

This function will read a parquet file that was saved by hydrofunctions.save_parquet() and return a dataframe and a metadata dictionary.

Parameters
`filename` (*`str`*) – A string with the filename and extension.

Returns
a pandas dataframe. meta (dict): a dictionary with the metadata for the NWIS data request, if it exists.

Return type
dataframe (pd.DataFrame)

`hydrofunctions.hydrofunctions.`**`save_json_gzip`**`(`*filename*, *json_dict*`)`

Save a Python dictionary as a gzipped JSON file.

This save function is especially designed to compress and save the original JSON response from the USGS NWIS. If no file extension is specified, then a .json.gz extension will be provided.

Parameters

- **`filename`** (*`str`*) – A string with the filename and extension.
- **`json_dict`** (*`dict`*) – A dictionary representing the json content.

`hydrofunctions.hydrofunctions.`**`save_parquet`**`(`*filename*, *dataframe*, *hf_meta*`)`

Save a hydrofunctions parquet file.

This function will save a dataframe and a dictionary into the parquet format. Parquet files are a compact, easy to process format that work well with Pandas and large datasets. This function will accompany the dataframe with a dictionary of NWIS metadata that is produced by the hydrofunctions.extract_nwis_df() function. This file can then be read by the hydrofunctions.read_parquet() function.

Parameters

- **`filename`** (*`str`*) – A string with the filename and extension.
- **`dataframe`** (*`pd.DataFrame`*) – a pandas dataframe.
- **`hf_meta`** (*`dict`*) – a dictionary with the metadata for the NWIS data request, if it exists.

`hydrofunctions.hydrofunctions.`**`select_data`**`(`*nwis_df*`)`

Create a boolean array of columns that contain data.

Parameters
`nwis_df` – A pandas dataframe created by `extract_nwis_df`.

Returns
an array of Boolean values corresponding to the columns in the original dataframe.

Example

```
>>> my_dataframe[:, select_data(my_dataframe)]
```

returns a dataframe with only the data columns; the qualifier columns do not show.

15.6 hydrofunctions.logging module

15.6.1 hydrofunctions.logging

This module contains the tools used for internal diagnostic logging.

Logging is disabled by default. Users can start logging by using the *hf._start_logging()* function. This will create a file "hydrofunctions_testing.log" in the main directory. This function also allows users to set the level of severity that will be logged. The default is to capture all messages, including the lowest level 'DEBUG' messages.

To create log messages within a module, follow these steps:

1. **Create a custom logger for the module.**
    - Place the statement *logger = logging.getLogger(__name__)* at the top of the module.
    - This will create a custom logger that is named after the module.
    - call the logger like this: logger.info("Hello!")
2. **Log a message within your code.**
    - Create a message: *msg = "This is the text of the message."*
    - Include the value of important variables
    - There is no need to include the time or name of the module or function. These are included in the standard message format.
    - **Decide on a 'level' for the message:**
        - DEBUG: this is the lowest level; for tracking ordinary internal values
        - INFO: internal or user events that are working as expected
        - WARNING: situations where back-up procedures are needed, unexpected situations and possibly ordinary exceptions that have been caught
        - ERROR: internal problems that prevent the software from completing an action
        - CRITICAL: serious errors that cause the shutdown of the software
    - Add the message to the log file: *logger.info(msg)*
    - Each level of severity has its own method: .debug(), .info(), .critical(), etc.
    - HydroExceptions such as HydroNoDataError generate their own error logs.
    - It might be useful to log other errors when they are raised.
3. **Start the logging system.**
    - Logging is off by default.

- To start logging, call hydrofunctions._start_logging()
- You can specify the level that will be captured in the log with the *loglevel* parameter.
- Set level like this: *hf._start_logging('info')* (Case does not matter)
- The default is to capture from the lowest level (DEBUG) up
- Starting the logging system will create a new file "hydrofunctions_testing.log" if it doesn't already exist; if it does, it will add new messages at the bottom under a start up message to the 'root' module.

4. **Read the log.**
   - The file, "hydrofunctions_testing.log" will appear in the root directory
   - **All messages from hydrofunctions will have the following:**
     - timestamp
     - name of logger: 'root' for the start message, all others should be named for the module that creates the message
     - levelname: the level of the message ('DEBUG', 'INFO', etc)
     - funcName: the name of the function that sent the message to the log
     - message: the message generated by the logger function: *logger.info("Hello!")*
   - The first message created by the *hf._start_logging()* will be from 'root'
   - Messages from dependencies will be captured too.

15.7 hydrofunctions.station module

15.7.1 hydrofunctions.station

This module contains the Station and NWIS classes, which are used for organizing and managing data for data collection sites.

`class hydrofunctions.station.NWIS(`*site=None, service='dv', start_date=None, end_date=None, stateCd=None, countyCd=None, bBox=None, parameterCd='all', period=None, interpolate=False, file=None, verbose=True*`)`

Bases: `Station`

A class for working with data from the USGS NWIS service.

Parameters

- **site** (*str or list of strings*) – a valid site is '01585200' or ['01585200', '01646502']. Default is *None*. If site is not specified, you will need to select sites using stateCd or countyCd.
- **service** (*str*) –

  can either be 'iv' or 'dv' for instantaneous or daily data.

  – 'dv'(default): daily values. Mean value for an entire day.

    - 'iv': instantaneous value measured at this time. Also known as 'Real-time data'. Can be measured as often as every five minutes by the USGS. 15 minutes is more typical.

- **start_date** (*str*) – should take on the form 'yyyy-mm-dd'
- **end_date** (*str*) – should take on the form 'yyyy-mm-dd'
- **stateCd** (*str*) – a valid two-letter state postal abbreviation, such as 'MD'. Default is None. Selects all stations in this state. Because this type of site selection returns a large number of sites, you should limit the amount of data requested for each site.
- **countyCd** (*str or list of strings*) – a valid county FIPS code. Default is None. Requests all stations within the county or list of counties. See https://en.wikipedia.org/wiki/FIPS_county_code for an explanation of FIPS codes.
- **bBox** (*str, list, or tuple*) –

  a set of coordinates that defines a bounding box.

    - Coordinates are in decimal degrees.
    - Longitude values are negative (west of the prime meridian).
    - Latitude values are positive (north of the equator).
    - comma-delimited, no spaces, if provided as a string.
    - The order of the boundaries should be: "West,South,East,North"
    - Example: "-83.000000,36.500000,-81.000000,38.500000"

- **parameterCd** (*str or list of strings*) – NWIS parameter code. Usually a five digit code. Default is 'all'. A valid code can also be given as a list: parameterCd=['00060','00065'] This will request data for this parameter.
  - if value is 'all', or no value is submitted, then NWIS will return every parameter collected at this site. (default option)
  - stage: '00065'
  - discharge: '00060'
  - not all sites collect all parameters!
  - See https://nwis.waterdata.usgs.gov/usa/nwis/pmcodes for full list
- **period** (*str*) –

  NWIS period code. Default is None.

    - Format is "PxxD", where xx is the number of days before today, with a maximum of 999 days accepted.
    - Either use start_date or period, but not both.

- **interpolate** (*bool*) – Fill missing values through interpolation. Default False.
- **file** (*str*) – A filename for acting as a cache for the data request. Accepts file extensions of '*.json.gz' (default) and* '.parquet'. If this parameter is included, the NWIS object will first attempt to read its data from the file. If the file does not exist, it will use the other parameters to obtain the data and will then save to the provided filename.

  Zipped JSON files will save the original WaterML JSON provided by the NWIS. Parquet files will save the dataframe and the metadata for the NWIS object.
- **verbose** (*bool*) – Print output for actions such as making data requests. Default is True.

df(**args*)

Return a subset of columns from the dataframe.

Parameters

- **''** – If no args are provided, the entire dataframe will be returned.
- **'all'** (*str*) – the entire dataframe will be returned.
- **'data'** (*str*) – all of the parameters will be returned, with no flags.
- **'flags'** (*str*) – Only the _qualifier flags will be returned. Unless the flags arg is provided, only data columns will be returned. Visit https://waterdata.usgs.gov/usa/nwis/uv?codes_help#dv_cd1 to see a more complete listing of possible codes.
- **'q'** (*str 'discharge' or*) – discharge columns ('00060') will be returned.
- **'stage'** (*str*) – Gauge height columns ('00065') will be returned.
- **number** (*str any eight to twelve digit*) – any matching parameter columns will be returned. '00065' returns stage, for example.
- **number** – any matching stations will be returned.

get_data()

Deprecated since version version: 0.2.0 No longer needed. NWIS object will request data upon creation.

read(*file*)

Read from a zipped WaterML file '.json.gz' or from a parquet file.

Parameters

file (*str*) – the filename to read from.

save(*file*)

Save the dataframe and metadata to a parquet file.

Parameters

file (*str*) – the filename to save to.

class hydrofunctions.station.**Station**(*site=None*)

Bases: `object`

A class for organizing stream gauge data for a single request.

station_dict = {}

15.8 hydrofunctions.validate module

15.8.1 hydrofunctions.validate

This module contains functions for testing that user input is valid.

Why 'pre-check' user imputs, instead of using standard python duck typing? These functions are meant to enhance an interactive session for the user, and will check a user's parameters before requesting data from an online resource. Otherwise, the server will return a 404 code and the user will have no idea why. Hydrofunctions tries to raise an exception (usually a TypeError) before a request is made, so that the user can fix their request. It also tries to provide a helpful error message to an interactive session user.

Suggested format for these functions:

- first check that the input is a string,

- then do a regular expression to check that the input is more or less valid.
- raise exceptions when user input breaks format.

---

hydrofunctions.validate.**check_NWIS_bBox**(*input*)

Checks that the USGS bBox is valid.

hydrofunctions.validate.**check_NWIS_service**(*input*)

Checks that the service is valid: either 'iv' or 'dv'

hydrofunctions.validate.**check_datestr**(*input*)

Checks that the start_date or end_date parameter is in yyyy-mm-dd format.

hydrofunctions.validate.**check_parameter_string**(*candidate*, *param*)

Checks that a parameter is a string or a list of strings.

hydrofunctions.validate.**check_period**(*input*)

Checks that the period parameter in is the P##D format, where ## is the number of days before now.

15.9 hydrofunctions.usgs_rdb module

15.9.1 hydrofunctions.usgs_rdb

This module is for working with the various USGS dataservices that use the rdb text format. These include the statistics service, the field measurements service, the rating curve service, and the peak discharge service.

hydrofunctions.usgs_rdb.**data_catalog**(*site*, *verbose=True*)

Load a history of the data collected at a site into a Pandas dataframe.

Parameters

- **site** (*str*) – The gauge ID number for the site.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a pandas dataframe. The dataframe will have one row for every type of data collected at each site requested; for each data parameter it will provide information including: parameter code, date of first observation, date of last observation, and total number of observations. A full description of the data catalog is given in the header; more information is available at: http://waterservices.usgs.gov/rest/Site-Service.html

For information about the site itself, including watershed area and HUC code, use the 'site_file' function.

Example:

```
>>> test = data_catalog('01542500')
>>> test
hydroRDB(header=<a mulit-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

`hydrofunctions.usgs_rdb.`**`field_meas`**(*site*, *verbose=True*)

Load USGS field measurements of stream discharge into a Pandas dataframe.

Parameters

- **site** (*str*) – The gauge ID number for the site.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a pandas dataframe. Each row of the table represents an observation on a given date of river conditions at the gauge by USGS personnel. Values are stored in columns, and include the measured stream discharge, channel width, channel area, depth, and velocity.

Example:

```
>>> test = field_meas('01542500')
>>> test
hydroRDB(header=<a mulit-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

Discussion:

The USGS operates over 8,000 stream gages around the United States and territories. Each of these sensors records the depth, or 'stage' of the water. In order to translate this stage data into stream discharge, the USGS staff creates an empirical relationship called a 'rating curve' between the river stage and stream discharge. To construct this curve, the USGS personnel visit all of the gage every one to eight weeks, and measure the stage and the discharge of the river manually.

The `field_meas()` function returns all of the field-collected data for this site. The USGS uses these data to create the rating curve. You can use these data to see how the site has changed over time, or to read the notes about local conditions.

The `rating_curve()` function returns the most recent 'expanded shift- adjusted' rating curve constructed for this site. This is the current official rating curve.

To plot a rating curve from the field measurements, use:

```
>>> header, data = hf.field_meas('01581830')

>>> data.plot(x='gage_height_va', y='discharge_va', kind='scatter')
```

Rating curves are typically plotted with the indepedent variable, gage_height, plotted on the Y axis.

`hydrofunctions.usgs_rdb.get_usgs_RDB_service`(*url*, *headers=None*, *params=None*)

Request data from a USGS dataservice and handle errors.

Parameters

- **url** (*str*) – a string used by Requests as the base URL.
- **header** (*dict*) – a dict of parameters used to request the data.
- **params** (*dict*) – a dict of parameters used to modify the url of a REST service.

Returns
A Requests response object.

Raises
This function will raise an exception for any non-200 status code, and in cases where the USGS service returns anything that is not obviously an RDB file. If an exception is raised, then an attempt will be made to display the error page which the USGS sometimes sends back to the user. –

class `hydrofunctions.usgs_rdb.hydroRDB`(*header*, *table*, *columns*, *dtypes*, *rdb_str*)

Bases: `object`

A class for holding the information from USGS rdb files.

Parameters

- **header** (*str*) – A multi-line string from the header of the rdb file. The header often contain important metadata and user warnings.
- **table** (*pandas dataframe*) – This is a dataframe made from the rdb file.
- **columns** (*str*) – A string from the rdb file that lists the column names.
- **dtypes** (*str*) – A string from the rdb file that gives the data type and length of each column.
- **rdb** (*str*) – The complete original text of the rdb file.

Properties:

header (str):
A multi-line string from the header of the rdb file. The header often contain important metadata and user warnings.

table (pandas dataframe):
This is a dataframe made from the rdb file.

columns (str):
A string from the rdb file that lists the column names.

dtypes (str):
A string from the rdb file that gives the data type and length of each column.

rdb (str):
The original, unparsed rdb file as returned by the USGS.

You can also access the header and the dataframe as a named tuple:

```
hydroRDB(header=<a multi-line string>, table=<pandas dataframe>)
```

Note:

- The args to create this object are supplied by hf.read_rdb().
- The hydroRDB object is returned from several functions that request RDB files from a USGS data service, including: peaks(), field_meas(), rating_curve(), stats(), site_file(), and data_catalog().
- You can read more about the RDB format here: https://pubs.usgs.gov/of/2003/ofr03123/6.4rdb_format.pdf

`hydrofunctions.usgs_rdb.peaks(site, verbose=True)`

Return a series of annual peak discharges.

Parameters

- **site** (*str*) – The gauge ID number for the site.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a table. The header is a multi-line string of metadata supplied by the USGS with the data series. The table is a dataframe containing the annual peak discharge series. You can use these data to conduct a flood frequency analysis.

Example:

```
>>> test = hf.peaks('01542500')
>>> test
hydroRDB(header=<a mulit-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

`hydrofunctions.usgs_rdb.rating_curve(site, verbose=True)`

Return the most recent USGS expanded-shift-adjusted rating curve for a given stream gage into a dataframe.

Parameters

- **site** (*str*) – The gage ID number for the site.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a table. The header is a multi-line string of metadata supplied by the USGS with the data series. The table is a dataframe containing the latest official rating curve for the site.

Example:

```
>>> test = rating_curve('01542500')
>>> test
```

(continues on next page)

(continued from previous page)

```
hydroRDB(header=<a mulit-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

Discussion:

The USGS operates over 8,000 stream gauges around the United States and territories. Each of these sensors records the depth, or 'stage' of the water. In order to translate this stage data into stream discharge, the USGS staff creates an empirical relationship called a 'rating curve' between the river stage and stream discharge.

See `hf.field_meas()` to access the field data used to construct the rating curve.

Note: Rating curves change over time.

hydrofunctions.usgs_rdb.**read_rdb**(*text*)

Read strings that are in rdb format.

Parameters

text (*str*) – A long string containing the contents of a rdb file. A common way to obtain these would be from the .text property of a requests response, as in the example usage below.

Returns

Every commented line at the top of the rdb file is marked with a
'#' symbol. Each of these lines is stored in this output.

outputDF (pandas.DataFrame):
A dataframe containing the information in the rdb file. *site_no* and *parameter_cd* are interpreted as a string, but every other number is interpreted as a float or int; missing values as an np.nan; strings for everything else.

columns (list of strings):
The column names, taken from the rdb header row.

dtypes (list of strings):
The second header row from the rdb file. These mostly tell the column width, and typically record everything as string data ('s') type. The exception to this are dates, which are listed with a 'd'.

Return type

header (multi-line string)

hydrofunctions.usgs_rdb.**site_file**(*site*, *verbose=True*)

Load USGS site file into a Pandas dataframe.

Parameters

- **site** (*str*) – The gauge ID number for the site.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a pandas dataframe. The dataframe will have one row for every site requested; for each site it will provide detailed site characteristics such as watershed area, drainage basin HUC code, site latitude, longitude, altitude, and datum; the date the site was established, hole depth for wells, and other information. All of the columns are listed in the header; for more information, visit: http://waterservices.usgs.gov/rest/Site-Service.html

For information on the data collected at this site (including the start and stop dates for data collection), use the 'data_catalog' function.

Example:

```
>>> test = site_file('01542500')
```

```
>>> test
hydroRDB(header=<a multi-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

`hydrofunctions.usgs_rdb.stats`(*site*, *statReportType='daily'*, *verbose=True*, ***kwargs*)

Return statistics from the USGS Stats Service as a dataframe.

Parameters

- **site** (*str*) – The gage ID number for the site, or a series of gage IDs separated by commas, like this: '01546500,01548000'.
- **statReportType** (*'daily'|'monthly'|'annual'*) – There are three different types of report that you can request.
  - 'daily' (default): calculate statistics for each of 365 days.
  - 'monthly': calculate statistics for each of the twelve months.
  - 'annual': calculate annual statistics for each year since the start of the record.
- **verbose** (*bool*) – If True (default), will print confirmation messages with the url before and after the request.

Returns

a hydroRDB object or tuple consisting of the header and a table. The header is a multi-line string of metadata supplied by the USGS with the data series. The table is a dataframe containing the latest official statistics for the site.

Raises

HTTPError – when a non-200 http status code is returned.

Example:

```
>>> test = stats('01542500', 'monthly')
>>> test
hydroRDB(header=<a mulit-line string of the header>,
         table=<a Pandas dataframe>)
```

You can also access the header, dataframe, column names, and data types through the associated properties *header*, *table*, *columns*, *dtypes*:

```
>>> test.table
<a Pandas dataframe>
```

Note: This function is based on the USGS statistics service, described here: https://waterservices.usgs.gov/rest/Statistics-Service.html

The USGS Statistics Service allows you to specify a wide array of additional parameters in your request. You can provide these parameters as keyword arguments, like in this example:

```
>>> hf.stats('01452500', parameterCD='00060')
```

This will only request statistics for discharge, which is specified with the '00060' parameter code.

Additional useful parameters include:

- *parameterCD='00060,00065'* Limit the request for statistics to only one parameter or to a list of parameters. The default behavior is to provide statistics for every parameter that has been measured at this site. In this example, statistics for discharge ('00060') and stage ('00065') are requested.
- *statYearType='water'* Calculate annual statistics based on the water year, which runs from October 1st to September 31st. This parameter is only for use with annual reports. If not specified, the default behavior will use calendar years for reporting.
- *missingData='on'* Calculate statistics even when there are some missing values. If not specified, the default behavior is to drop years that have fewer than 365 values from annual reports, and to drop months that have fewer than 30 values in monthly reports. The number of values used to calculate a statistic is reported in the 'count_nu' column.
- You can read about other useful parameters here: https://waterservices.usgs.gov/rest/Statistics-Service.html#statistical_Controls

---

15.10 hydrofunctions.waterwatch module

15.10.1 hydrofunctions.waterwatch

This module is for working with the five USGS WaterWatch Data Services. Description of data services https://waterwatch.usgs.gov/webservices/

Main page: https://waterwatch.usgs.gov

NOTICE (taken from waterwatch.usgs.gov): In January 2020, USGS WaterWatch began operating in maintenance-only mode. Existing tools, features, and web data services are being fully maintained as before, but new tools and enhancements will no longer be developed. Please click here for more information or contact USGS WaterWatch if you have any questions.

The WaterWatch program provides five data services with REST APIs:

- Current Conditions Real-Time Streamflow Service
- Flood and High Flow Service
- Average Streamflow for 7, 14, and 28 Days Service
- Hourly Flow Change Service

- Flood Stage Service

Hydrofunctions allows you to access each of these services as either a dictionary or a dataframe with the station ID as the key/index. ——

`hydrofunctions.waterwatch.`**`filter_flood_stages`**(*all_flood_stages*, *sites_numbers=None*)

Filters flood states of specific station numbers

`hydrofunctions.waterwatch.`**`get_flood_stage`**(*site=None*, *output_format='dict'*)

Retrieves flood stages for a list of station numbers.

This function retrieves a dictionary of flood stages for each site. The 'stage' of a river is the height of the river surface at a stream gauge, expressed as a height above an arbitrary datum. It is similar to water depth, except that datums are usually set so that the zero (0) to be well below the lowest elevation of the stream bed. This is done so that even if there is erosion over time, the stream bed and the river stage will never reach an elevation that is less than zero. Stage is usually expressed in feet in this dataset. You can retrieve the stage of the river using the parameter '00065', whereas the discharge of the river is '00060'.

There are several kinds of flood stage reported in these data:

- **action stage: If the water gets above this level, it triggers an action by** the National Weather Service.
- **flood stage: Water at this level begins to be a hazard to lives, property, or** commerce. Not necessarily the same as bankfull stage.
- moderate flood stage: structures and roads begin to be inundated.
- **major flood stage: requires significant evacuations of people or transfer of** property to higher elevations.

See https://waterwatch.usgs.gov/webservices/ for more information.

Parameters

- **site** (*str or list of str*) – The USGS site ID number or a list of numbers.
- **output_format** – Optional output format. Returns dict if 'dict' else returns pd.DataFrame

Returns: Dictionary or DataFrame of station numbers and their flood stages. If there is no flood stage for a station, *None* is returned.

Example

```
>>> stations = ["07144100", "07144101"]
>>> res = get_flood_stage(stations, output_format="dict")  # dictionary output
>>> print(res)
{'07144100': {'action_stage': '20', 'flood_stage': '22', 'moderate_flood_stage': '25
↪', 'major_flood_stage': '26'},
 '07144101': None}
>>> print(get_flood_stage(stations))
```

CHAPTER

SIXTEEN

EXAMPLE APPLICATIONS

16.1 Comparing Urban and Rural Streams

In this notebook we'll compare the hydrology of two streams in the Baltimore area: * Grave Run, a rural stream with only 0.3% impervious surfaces, and * Dead Run, an urban stream that is 39% impervious surfaces.

These two watersheds are similar in size, topography, and geology. They also have very emo names.

[1]:
```
# Start with the usual.
import hydrofunctions as hf
%matplotlib inline
hf.__version__
```

[1]:
```
'0.2.0'
```

[ ]:
```
# request data for our two sites for a three-year period.
sites = ['01589330', '01581830']
request = hf.NWIS(sites, start_date='2002-01-01', end_date='2005-01-01', file='Urban_
↪Rural.parquet')
request # Describe the dataset that we've received.
```

[4]:
```
# We'll store our discharge data in a dataframe named 'Q'
Q = request.df('discharge')
Q.head()  # Print the first five rows to verify.
```

[4]:
```
                           USGS:01581830:00060:00003  \
datetimeUTC
2002-01-01 00:00:00+00:00                       3.20
2002-01-02 00:00:00+00:00                       3.20
2002-01-03 00:00:00+00:00                       3.30
2002-01-04 00:00:00+00:00                       3.20
2002-01-05 00:00:00+00:00                       4.18

                           USGS:01589330:00060:00003
datetimeUTC
2002-01-01 00:00:00+00:00                       0.60
2002-01-02 00:00:00+00:00                       0.61
2002-01-03 00:00:00+00:00                       0.60
2002-01-04 00:00:00+00:00                       0.64
2002-01-05 00:00:00+00:00                       0.60
```

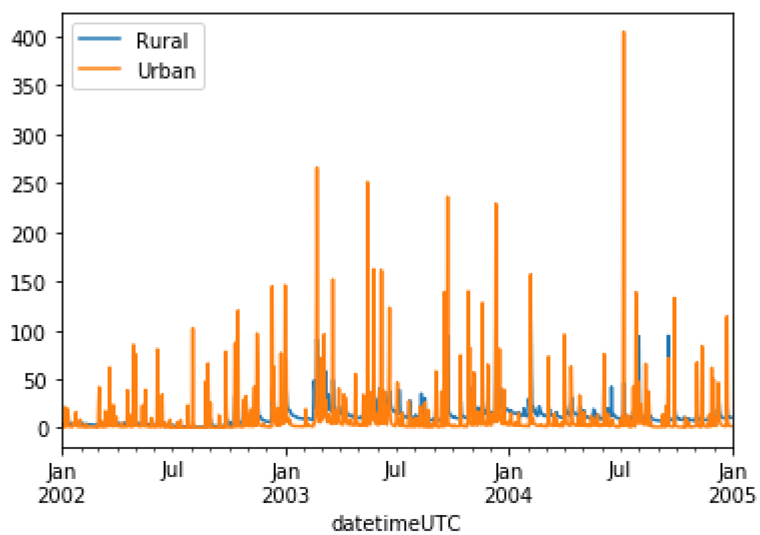
[5]:
```
# Rename the columns to 'Urban' and 'Rural' so we remember which is which!
Q = Q.rename(columns={"USGS:01589330:00060:00003": "Urban", "USGS:01581830:00060:00003":
↪"Rural"})
Q.head()  # List the first five rows.
```

[5]:
```
                           Rural  Urban
datetimeUTC
2002-01-01 00:00:00+00:00   3.20   0.60
2002-01-02 00:00:00+00:00   3.20   0.61
2002-01-03 00:00:00+00:00   3.30   0.60
2002-01-04 00:00:00+00:00   3.20   0.64
2002-01-05 00:00:00+00:00   4.18   0.60
```

[6]:
```
# Let's plot our data to create a hydrograph. plot() is a method that is built-in to␣
↪dataframes.
Q.plot()
```

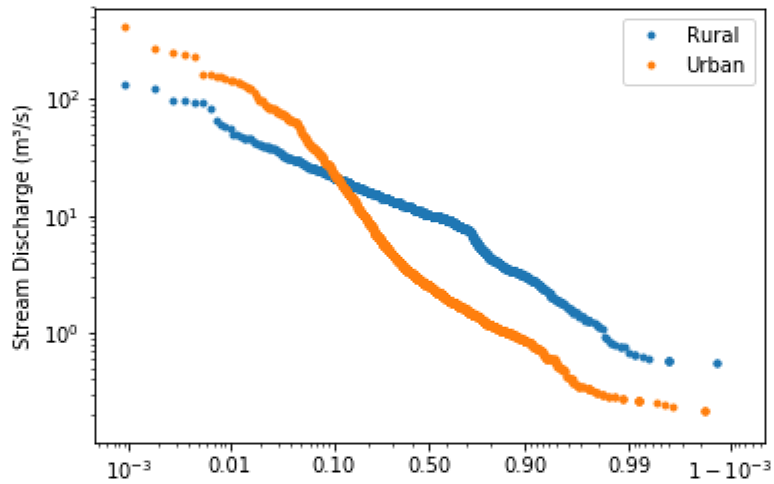
```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

[6]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a99c4fd278>
```



[7]:
```
# Let's create a flow duration chart for our two sites!
# .flow_duration is a function included in Hydrofunctions. It accepts dataframes as␣
↪input.
hf.flow_duration(Q)
```

[7]:
```
(<Figure size 432x288 with 1 Axes>,
 <matplotlib.axes._subplots.AxesSubplot at 0x1a99c68d828>)
```



16.1.1 Description of the two sites

If you look carefully at the **hydrograph** above, you can see that the orange urban site tends to have lower baseflow, but it also tends to have higher peaks during storms. Unfortunately, this obscures the hydrograph for the rural site a little!

The second diagram is a **flow duration chart**. The default Y axis is logarithmic, with values ranging from less than 1 to about 400 cfs. The default X axis uses a logit scale to plot the chance of exceedance. Values range from greater than zero to less than one, with a value of 0.9 meaning that 90% of the flows are higher than this value. The logit scale stretches out the extreme high and extreme low values so that the distance from the center to one standard deviation is approximately the same as from two standard deviations out to three standard deviations. This approximates the probability scale that Flow Duration charts are often plotted on.

Comparing the two sites, you can see that the orange urban site has lower baseflows than the blue rural site, but also has higher peak flows.

[8]:
```
# let's compare stats for the two sites!
Q.describe()
```

[8]:

	Rural	Urban
count	1097.000000	1097.000000
mean	12.005588	10.456764
std	10.935054	27.932960
min	0.540000	0.170000
25%	5.750000	1.400000
50%	10.000000	2.510000
75%	14.100000	5.820000
max	130.000000	405.000000

16.1.2 Further study

Now that we've used flow duration charts to compare an urban stream to a rural stream, try comparing:

- a large stream to a small stream (choose two sites on the same river!)
- a desert stream to a forested stream
- a high-gradient mountain stream to a low gradient lowland stream

Before you find appropriate sites to compare, think about how these different conditions will affect the hydrology of the watershed. What impact does infiltration have on the 'flashiness' of a stream? How quickly will a large watershed respond to a storm compared to a small stream?

Good luck!

16.2 Comparing Three Nested Watersheds

In this notebook we'll examine three sites along the Gwynns Falls, in Maryland. 'Falls' in this case is a local word for 'River', like in the Jones Falls or Gunpowder Falls. It doesn't refer to a waterfall!

Because the three sites are in a row on the same river, the upstream site has a watershed that is nested inside of the middle site, which is nested inside the downstream site, which has the largest watershed.

[1]:
```
import hydrofunctions as hf
%matplotlib inline
```

The three stream gauges are:

- the farthest upstream is GWYNNS FALLS NEAR DELIGHT, MD, dv01589197
- the middle of stream is GWYNNS FALLS AT VILLA NOVA, MD, dv01589300
- the farthest downstream is GWYNNS FALLS AT WASHINGTON BLVD AT BALTIMORE, MD, dv01589352

[2]:
```
streamid = ['01589197', '01589300','01589352']
# request data for our two sites for a three-year period.
sites = hf.NWIS(streamid, 'dv', start_date='2001-01-01', end_date='2003-12-31')
sites
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01589197%2C01589300%2C01589352&startDT=2001-01-01&endDT=2003-12-31
```

[2]:
```
USGS:01589197: GWYNNS FALLS NEAR DELIGHT, MD
    00060: <Day>  Discharge, cubic feet per second
USGS:01589300: GWYNNS FALLS AT VILLA NOVA, MD
    00060: <Day>  Discharge, cubic feet per second
USGS:01589352: GWYNNS FALLS AT WASHINGTON BLVD AT BALTIMORE, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2001-01-01 00:00:00+00:00
End:   2003-12-31 00:00:00+00:00
```

[3]:
```
#create a dataframe of the sites
Q = sites.df('discharge')
#rename the columns
Q.columns=['Upper', 'Middle', 'Lower']
#show the first few rows of the data
Q.head()
```

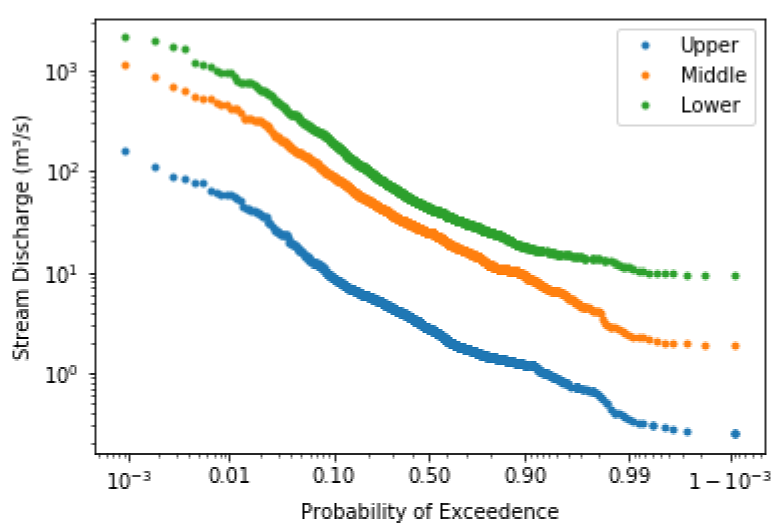
[3]:
```
                              Upper  Middle  Lower
datetimeUTC
2001-01-01 00:00:00+00:00      1.8    18.0   29.0
2001-01-02 00:00:00+00:00      1.8    17.0   28.0
2001-01-03 00:00:00+00:00      1.8    16.0   27.0
2001-01-04 00:00:00+00:00      1.8    17.0   31.0
2001-01-05 00:00:00+00:00      1.8    16.0   31.0
```

[4]:
```
#look at the descriptive statistics for each part of the stream.
#Note that the mean and standard deviation increase as you move down the stream
Q.describe()
```

[4]:
```
             Upper       Middle        Lower
count  1095.000000  1095.000000  1095.000000
mean      5.211306    44.910210    92.074055
std      10.177445    79.268103   173.157576
min       0.250000     1.860000     8.730000
25%       1.590000    14.000000    27.800000
50%       2.720000    24.400000    43.000000
75%       5.000000    42.850000    80.800000
max     161.000000  1140.000000  2140.000000
```

[5]:
```
#create a flow duration for each of the streams
hf.flow_duration(Q)
```

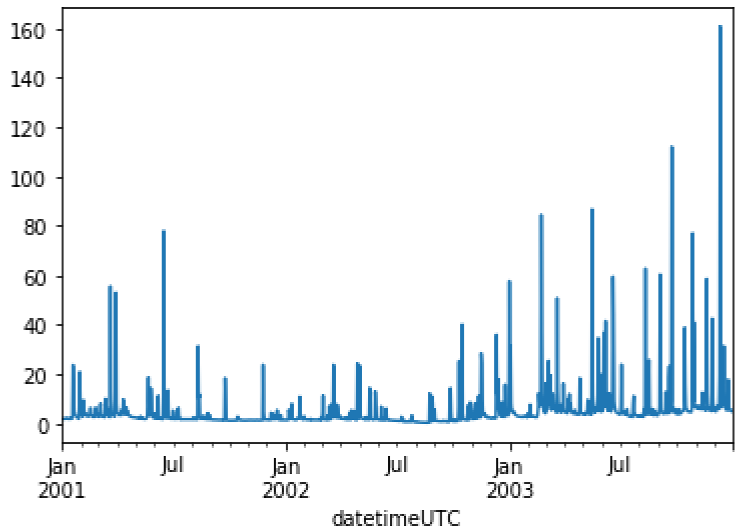
[5]:
```
(<Figure size 432x288 with 1 Axes>,
 <matplotlib.axes._subplots.AxesSubplot at 0x1fe91e24978>)
```



[6]:
```
# create a hydrograph of the upper portion of the stream.
# the .loc method selects a subset of a dataframe.
# The first item selects rows, with ':' selecting every row.
# The second item selects columns, with 'Upper' selecting the column with the 'Upper' label.
# The .plot() method plots the values in the columns on the y axis, with the rows as the
↪x axis.
Q.loc[:, 'Upper'].plot()
```

```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

[6]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1fe920d60b8>
```



16.3 Comparing Different Stream Environments

This Jupyter Notebook compares four streams in different environments in the U.S. Using hydrofunctions, we are able to plot the flow duration graphs for all four streams and compare them.

[1]:
```
import hydrofunctions as hf
%matplotlib inline
```

Choose four streams from different environments from HydroCloud. Import data for three years.

In this example, all four streams are in places with low development:

- Colorado Western Slopes: ROARING FORK RIVER NEAR ASPEN, CO.
- California Mendicino National Park: MAD R AB RUTH RES NR FOREST GLEN CA
- White Mountains, NH: EAST BRANCH PEMIGEWASSET RIVER AT LINCOLN, NH
- PINTO CREEK NEAR MIAMI, AZ

[2]:
```
streams = ['09073400','11480390','01074520','09498502']
sites = hf.NWIS(streams, 'dv', start_date='2001-01-01', end_date='2003-12-31')
sites
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=09073400%2C11480390%2C01074520%2C09498502&startDT=2001-01-01&endDT=2003-12-31
```

[2]:
```
USGS:01074520: EAST BRANCH PEMIGEWASSET RIVER AT LINCOLN, NH
    00060: <Day>  Discharge, cubic feet per second
USGS:09073400: ROARING FORK RIVER NEAR ASPEN, CO.
```

(continues on next page)

(continued from previous page)

```
    00060: <Day>  Discharge, cubic feet per second
USGS:09498502: PINTO CREEK NEAR MIAMI, AZ
    00060: <Day>  Discharge, cubic feet per second
USGS:11480390: MAD R AB RUTH RES NR FOREST GLEN CA
    00060: <Day>  Discharge, cubic feet per second
Start: 2001-01-01 00:00:00+00:00
End:   2003-12-31 00:00:00+00:00
```

[3]:
```
#Create a dataframe of the four sites
Q = sites.df('discharge')
#Show the first few lines of the dataframe
Q.head()
```

[3]:
```
                           USGS:01074520:00060:00003  \
datetimeUTC
2001-01-01 00:00:00+00:00                      160.0
2001-01-02 00:00:00+00:00                      138.0
2001-01-03 00:00:00+00:00                      132.0
2001-01-04 00:00:00+00:00                      125.0
2001-01-05 00:00:00+00:00                      130.0

                           USGS:09073400:00060:00003  \
datetimeUTC
2001-01-01 00:00:00+00:00                       22.0
2001-01-02 00:00:00+00:00                       23.0
2001-01-03 00:00:00+00:00                       21.0
2001-01-04 00:00:00+00:00                       23.0
2001-01-05 00:00:00+00:00                       24.0

                           USGS:09498502:00060:00003  \
datetimeUTC
2001-01-01 00:00:00+00:00                        2.7
2001-01-02 00:00:00+00:00                        2.7
2001-01-03 00:00:00+00:00                        2.7
2001-01-04 00:00:00+00:00                        2.7
2001-01-05 00:00:00+00:00                        2.7

                           USGS:11480390:00060:00003
datetimeUTC
2001-01-01 00:00:00+00:00                       21.0
2001-01-02 00:00:00+00:00                       19.0
2001-01-03 00:00:00+00:00                       16.0
2001-01-04 00:00:00+00:00                       15.0
2001-01-05 00:00:00+00:00                       13.0
```

[4]:
```
# rename the columns based on the names of the sites from HydroCloud
Q.columns=['White Mountains National Park', 'White River National Forest', 'Tonto␣
↪National Forest', 'Mendicino National Park']
# show the first few rows of the data to confirm the changes
Q.head()
```

[4]:
```
                           White Mountains National Park  \
```

(continues on next page)

(continued from previous page)

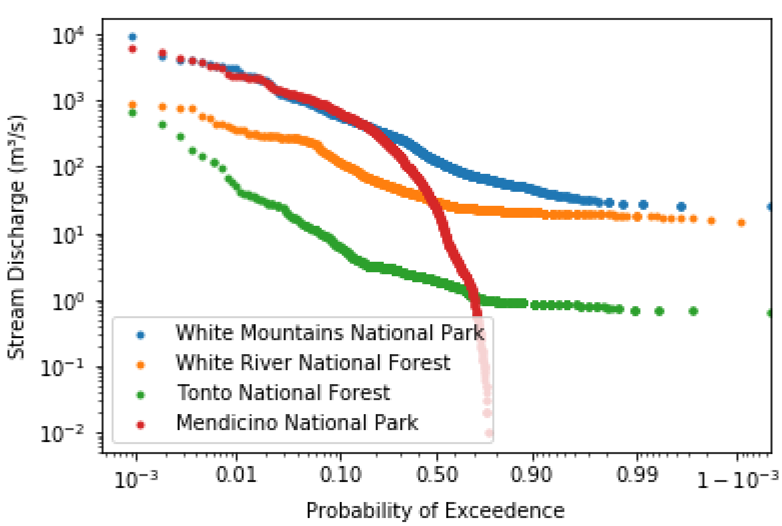
```
datetimeUTC
2001-01-01 00:00:00+00:00                          160.0
2001-01-02 00:00:00+00:00                          138.0
2001-01-03 00:00:00+00:00                          132.0
2001-01-04 00:00:00+00:00                          125.0
2001-01-05 00:00:00+00:00                          130.0

                           White River National Forest  Tonto National Forest  \
datetimeUTC
2001-01-01 00:00:00+00:00                         22.0                    2.7
2001-01-02 00:00:00+00:00                         23.0                    2.7
2001-01-03 00:00:00+00:00                         21.0                    2.7
2001-01-04 00:00:00+00:00                         23.0                    2.7
2001-01-05 00:00:00+00:00                         24.0                    2.7

                           Mendicino National Park
datetimeUTC
2001-01-01 00:00:00+00:00                     21.0
2001-01-02 00:00:00+00:00                     19.0
2001-01-03 00:00:00+00:00                     16.0
2001-01-04 00:00:00+00:00                     15.0
2001-01-05 00:00:00+00:00                     13.0
```

[5]:
```
#use the built-in functions from hydrofunctions to create a flow duration graph for the␣
↪dataframe.
hf.flow_duration(Q)
```

[5]:
```
(<Figure size 432x288 with 1 Axes>,
 <matplotlib.axes._subplots.AxesSubplot at 0x23235ac5860>)
```



[6]:
```
#Pull the stats for each of the four sites.
Q.describe()
```

[6]:
```
       White Mountains National Park  White River National Forest  \
count                    1095.000000                  1095.000000
```

(continues on next page)

(continued from previous page)

```
mean                     282.006941                   57.492785
std                      526.176775                   81.064415
min                       25.000000                   14.000000
25%                       66.300000                   22.700000
50%                      123.000000                   29.000000
75%                      305.500000                   51.850000
max                     9090.000000                  916.000000

       Tonto National Forest  Mendicino National Park
count            1095.000000              1095.000000
mean                5.144813               221.987370
std                28.089932               514.204177
min                 0.670000                 0.000000
25%                 1.000000                 0.085000
50%                 1.920000                22.500000
75%                 3.000000               212.500000
max               691.000000              6270.000000
```

16.3.1 Analysis

Based on the flow duration chart and the descriptive statistics, the largest two sites are in Mendicino and the White Mountains of New Hampshire. However, if you look at the red line for the Mendicino site, it trails off and drops to zero between the 70% and 80% mark. It appears that this river had no water in it for approximately 22% of the days during these three years!

The other two sites at White River and in the Tonto National Forest seem to be same size at higher flows, but the Tonto site, in Arizona, tends to have lower low flows.

16.4 Daily Mean vs. Instant

The NWIS provides continuous discharge data in two different formats: 'Daily Mean Discharge' and 'Instantaneous Values'. The Daily Mean Discharge is the average discharge for an entire day, from midnight, local time to midnight local time. The Instantaneous Discharge is also known as 'Real Time' data. It is the current discharge at the time of the measurement. In this case, measurements can occur as often as every five minutes, but most often is every fifteen minutes. Both Daily Mean and Instantaneous discharge values are given in units of 'cubic feet per second'.

On this page I illustrate the differences in the two datasets and show applications for each.

[1]:
```
# Start our Jupyter Notebook the usual way:
import hydrofunctions as hf
import matplotlib.pyplot as plt
print(hf.__version__)
%matplotlib inline
```

```
0.2.0
```

16.4.1 Loading Daily Values

First, let's load a month of Daily Discharge data using the 'DV', or Daily Values service.

[2]:
```
daily = hf.NWIS('01589330', 'dv', '2018-06-01', '2018-07-01', file='DeadRun_Daily.parquet
↪')
```

```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01589330&startDT=2018-06-01&endDT=2018-07-01
Saving data to DeadRun_Daily.parquet
```

When we list the dataset, you can see that the discharge is collected at a daily interval.

[3]:
```
daily
```

[3]:
```
USGS:01589330: DEAD RUN AT FRANKLINTOWN, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2018-06-01 00:00:00+00:00
End:   2018-07-01 00:00:00+00:00
```

16.4.2 Loading Instantaneous Values

Next, let's collect data for the same station at the same time, but we'll request the 'iv' Instantaneous Values. Note that these get collected every 5 Minutes according to the listing.

[4]:
```
instant = hf.NWIS('01589330', 'iv', '2018-06-01', '2018-07-01', file='DeadRun_Instant.
↪parquet')
instant
```

```
Requested data from https://nwis.waterservices.usgs.gov/nwis/iv/?format=json%2C1.1&
↪sites=01589330&startDT=2018-06-01&endDT=2018-07-01
Saving data to DeadRun_Instant.parquet
```

[4]:
```
USGS:01589330: DEAD RUN AT FRANKLINTOWN, MD
    00060: <5 * Minutes>  Discharge, cubic feet per second
    00065: <5 * Minutes>  Gage height, feet
Start: 2018-06-01 04:00:00+00:00
End:   2018-07-02 03:55:00+00:00
```

Now, we'll put the discharge data into two dataframes, one for the Daily data, and one for the Instantaneous data. The .head() method will list the first five lines of each table.

[5]:
```
# This table needs to be shifted four hours: The data are averaged over 24 hours,
# midnight to midnight, local time (+4 UTC).
D = daily.df('q').shift(4, freq='H')
D.head()
```

[5]:

datetimeUTC	USGS:01589330:00060:00003
2018-06-01 04:00:00+00:00	6.76
2018-06-02 04:00:00+00:00	13.60
2018-06-03 04:00:00+00:00	179.00
2018-06-04 04:00:00+00:00	16.20
2018-06-05 04:00:00+00:00	11.80

[6]:
```
I = instant.df('q')
I.head()
```

[6]:
```
                          USGS:01589330:00060:00000
datetimeUTC
2018-06-01 04:00:00+00:00                      23.9
2018-06-01 04:05:00+00:00                      23.1
2018-06-01 04:10:00+00:00                      22.3
2018-06-01 04:15:00+00:00                      21.5
2018-06-01 04:20:00+00:00                      20.8
```

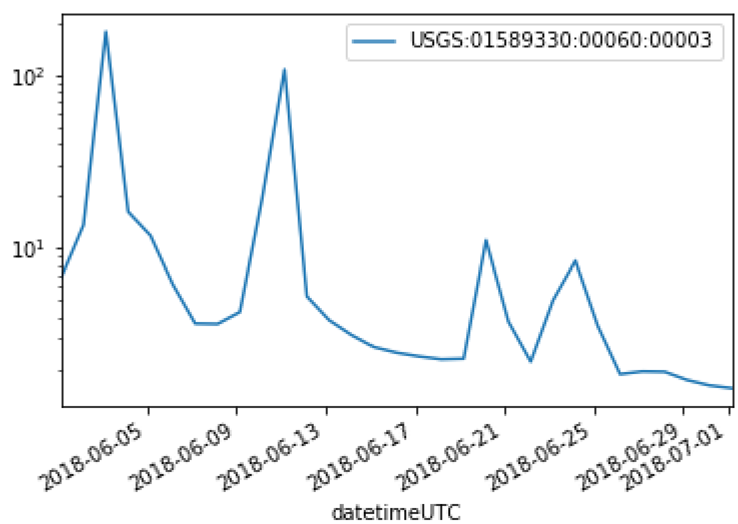
16.4.3 Plotting discharge

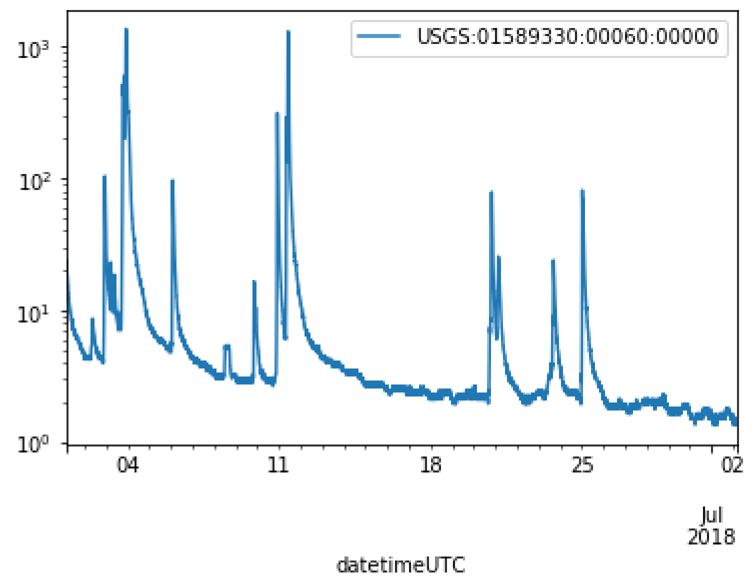
A simple plot of the data shows that the two hydrographs are similar, but of course the Daily data were collected less often. Also, note the the peaks are lower for the daily data.

[7]:
```
# Discharge data can vary over several orders of magnitude, so we usually plot it on a␣
↪logarithmic scale.
D.plot(logy=True)
I.plot(logy=True)
```

```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

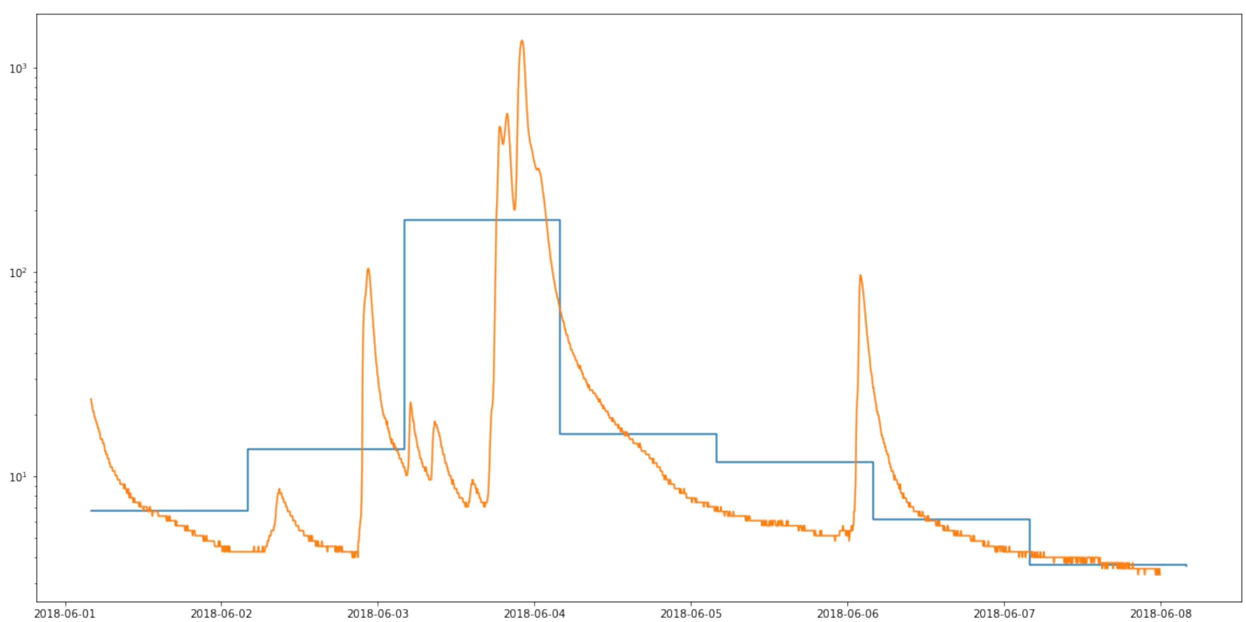
[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a2b8ae9198>
```





That was nice and all, but it would be better if we could plot our data on the same graph. Also, since the daily mean discharage data represent the mean discharge over the course of an entire day, these data are better plotted with a step function.

[8]:
```
# A week of data
fig, ax = plt.subplots(figsize=(20, 10))
ax.step(D.loc['2018-06-01':'2018-06-08'].index.values, D.loc['2018-06-01':'2018-06-08'].
↪values, where='post')
ax.plot(I.loc['2018-06-01':'2018-06-07'])
plt.yscale('log')
```



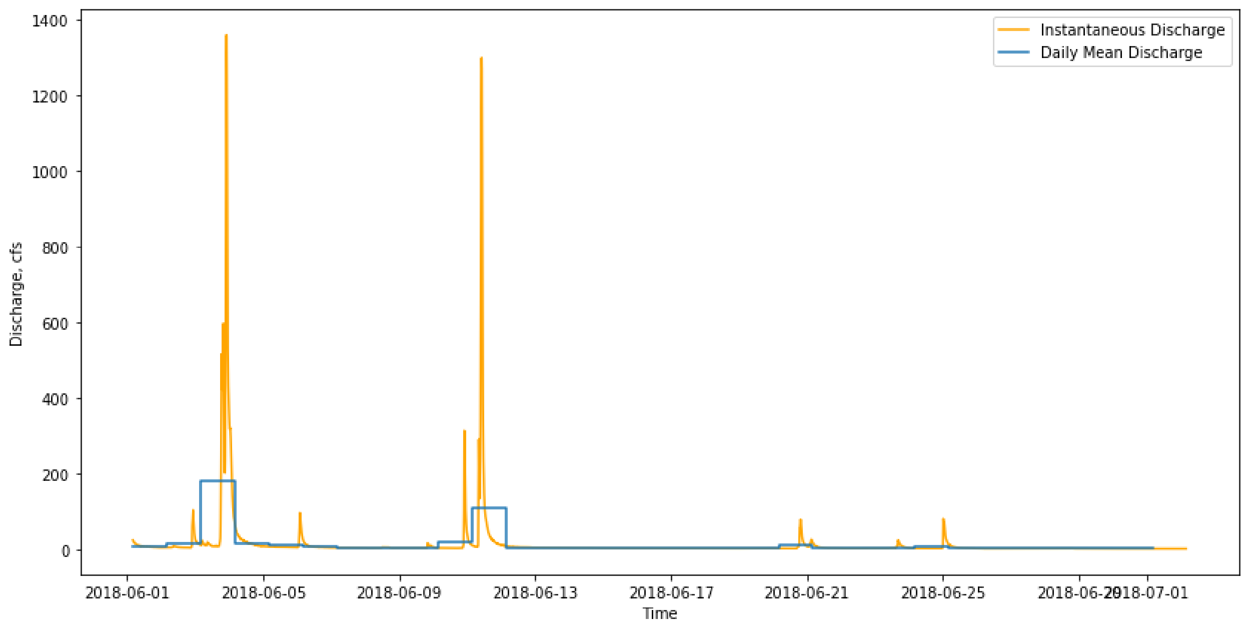
[9]:
```
# Plot the entire month-long dataset.
fig, ax = plt.subplots(figsize=(14, 7))
```

(continues on next page)

(continued from previous page)

```
ax.plot(I, "orange", label="Instantaneous Discharge")
ax.step(D.index.values, D.values, where='post', label="Daily Mean Discharge")
plt.xlabel('Time')
plt.ylabel('Discharge, cfs')
plt.legend()
# I've commented out the command to plot this on a logarithmic scale, but you can try it.
#plt.yscale('log')
```

[9]:
```
<matplotlib.legend.Legend at 0x1a2ba7cd6a0>
```



16.4.4 Why use Daily Mean or Instantaneous values?

The Instantaneous Values are the 'raw' data that gets collected at each stream gauge site. They get processed to produce the Daily Mean Discharge values. The Daily Mean discharge data then become the 'official' data for each USGS stream gauge. These data get error-checked.

The Daily Mean discharge data are great for situations where you want to know the baseflow of a watershed, or the volume of water that is entering a reservoir or if you want to learn about the long-term hydrology of a watershed. Daily values get used for:

- Flow Duration charts
- Water balance models
- Water quality models

The Instantaneous Values are better for examining the shape and timing of flood waves. They get used for:

- Determining the size of floods at their peak,
- and the timing of the peak
- these are compiled into the peak discharge data, which you can access with the command: `hf.peak('01589330')

16.5 Hysteresis in River Systems

Hysteresis is a condition where a dependent variable is controlled not only by the ***value*** of an independent variable, but also by its ***direction***. For example, streams carry a higher sediment load when a flood is receding than when it is rising. In some cases, this hysteresis may be caused by a lag, where the dependent variable responds to the independent variable after a short wait. This may be the case with sediment load, where it takes longer for sediment to move downstream than water, but it may also be due to some sort of non-linear process. For example, stream banks may be stable when flood waters are rising, but they become saturated and then collapse as the discharge and the water levels decrease.

Let's illustrate these ideas with some data!

[1]:
```
import hydrofunctions as hf
print(hf.__version__)
import matplotlib.pyplot as plt
import matplotlib
matplotlib.style.use('seaborn-notebook')
%matplotlib inline
```

```
0.2.1
```

16.5.1 Hystersis in flood waves

First, let's compare a flood wave that travels downstream from one stream gauge to another with little change. We'll use two gauges along the West Branch of the Susquehanna: Karthaus (upstream) and Williamsport (downstream).

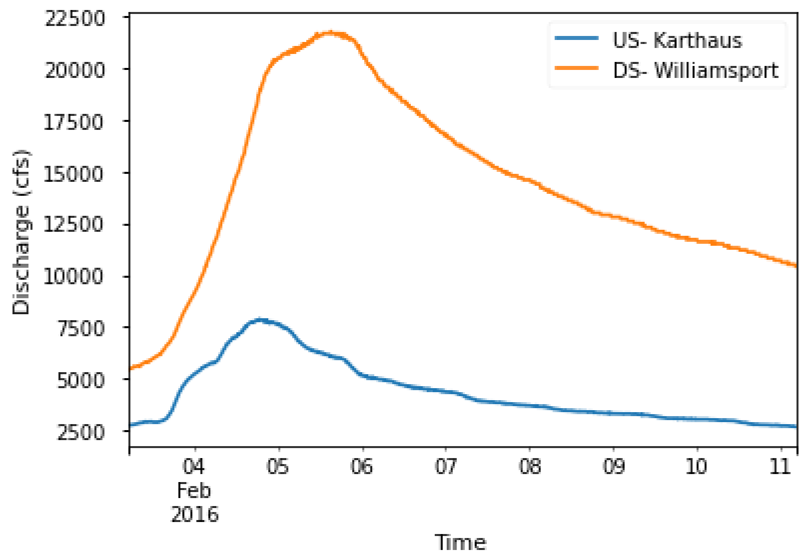
[2]:
```
Karthaus = '01542500'
Williamsport = '01551500'
request1 = hf.NWIS([Karthaus, Williamsport], 'iv', start_date='2016-02-03', end_date=
↪'2016-02-10')
```

```
Requested data from https://nwis.waterservices.usgs.gov/nwis/iv/?format=json%2C1.1&
↪sites=01542500%2C01551500&startDT=2016-02-03&endDT=2016-02-10
```

When we plot the discharge data over time, the flood wave at Williamsport looks similar to how it looked upstream at Karthaus a day earlier. The only difference is that the discharge is higher and the lag. It took about 21 hours for this flood to travel the 200 km between the two sites.

[3]:
```
flood_data = request1.df('q').copy()
flood_data.rename(columns={'USGS:01542500:00060:00000':'US- Karthaus', 'USGS:01551500:
↪00060:00000': 'DS- Williamsport'}, inplace=True)
flood_data.plot(xlabel='Time', ylabel='Discharge (cfs)')
```

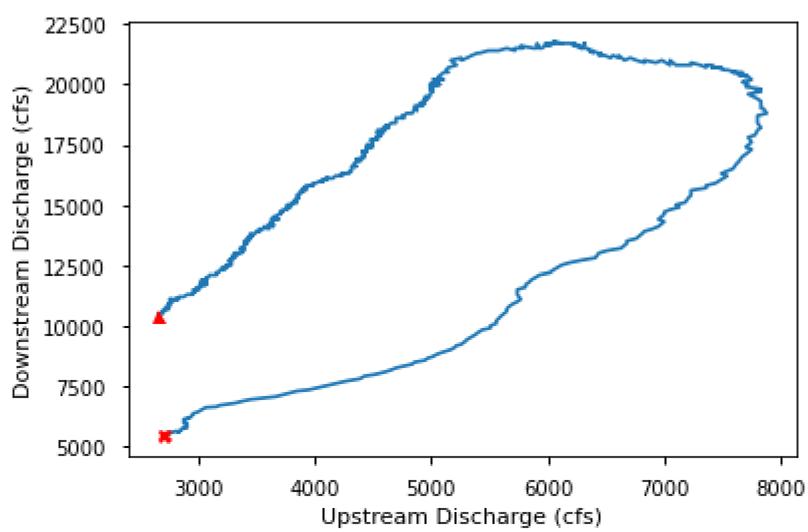
[3]:
```
<AxesSubplot:xlabel='Time', ylabel='Discharge (cfs)'>
```



Now that we've seen the discharge data plotted over time, let's plot the discharge at each site against each other. In this case, I've plotted the upstream discharge at Karthaus on the X axis, and the downstream discharge at Williamsport on the Y axis. I've placed a small 'x' on the first measurement and a triangle on the last measurement so you can tell where the series began and ended.

[4]:
```
fig, ax = plt.subplots()
x = flood_data.loc[:,'US- Karthaus'].values
y = flood_data.loc[:,'DS- Williamsport'].values
ax.plot(x,y)
ax.plot(x[0], y[0], 'rX') # plot an 'X' at the beginning
ax.plot(x[-1], y[-1], 'r^') # plot a triangle at the end
ax.set_xlabel('Upstream Discharge (cfs)')
ax.set_ylabel('Downstream Discharge (cfs)')
```

[4]:
```
Text(0, 0.5, 'Downstream Discharge (cfs)')
```



In the graph above, the values move counter-clockwise. It moves this way because the X-axis (the upstream site) increases before the Y-axis (the downstream site) due to the lag between sites. This is hysteresis! If you had plotted all of the points without paying attention to their order, the relationship between the the variables would have been much less clear!

16.5.2 Hysteresis in sediment load

Our next example of hysteresis occurs in the relationship between discharge and sediment load.

[5]:
```
# To find sites that collect sediment load data, use a query like this:
# sus_sed_sites = hf.NWIS(stateCd='PA', parameterCd='80155')
# More information on sediment data: https://cida.usgs.gov/sediment/helpGuide.jsp
# To determine the dates that sediment load was collected at a site, use:
# https://waterdata.usgs.gov/nwis/inventory/?site_no=01545500&agency_cd=USGS
```

[6]:
```
# We'll request data from another site along the West Branch of the Susquehanna at Renovo,
↪ PA.
renovo = hf.NWIS('01545500', 'dv', start_date='1967-01-26', end_date='1968-09-28')
renovo
```

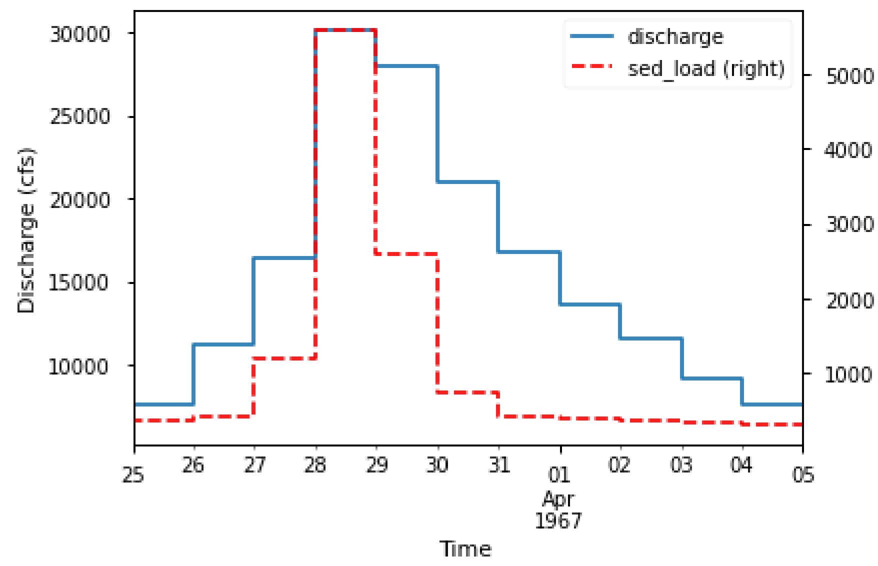
```
Requested data from https://waterservices.usgs.gov/nwis/dv/?format=json%2C1.1&
↪sites=01545500&startDT=1967-01-26&endDT=1968-09-28
```

[6]:
```
USGS:01545500: West Branch Susquehanna River at Renovo, PA
    00060: <Day>  Discharge, cubic feet per second
    80154: <Day>  Suspended sediment concentration, milligrams per liter
    80155: <Day>  Suspended sediment discharge, short tons per day
Start: 1967-01-26 00:00:00+00:00
End:   1968-09-28 00:00:00+00:00
```

Now, let’s plot the daily discharge and daily sediment load for a single flood in 1967.

[7]:
```
may = renovo.df('q', '80155').loc['1967-03-25':'1967-04-05']
may.rename(columns={'USGS:01545500:00060:00003':'discharge', 'USGS:01545500:80155:00003':
↪'sed_load'}, inplace=True)
may['discharge'].plot(drawstyle='steps', xlabel='Time', ylabel='Discharge (cfs)',␣
↪legend=True)
may['sed_load'].plot(secondary_y=True, drawstyle='steps', xlabel='Time', style='r--',␣
↪legend=True)
```

[7]:
```
<AxesSubplot:label='00838a01-1f7a-45bb-919c-75695da3ef73'>
```

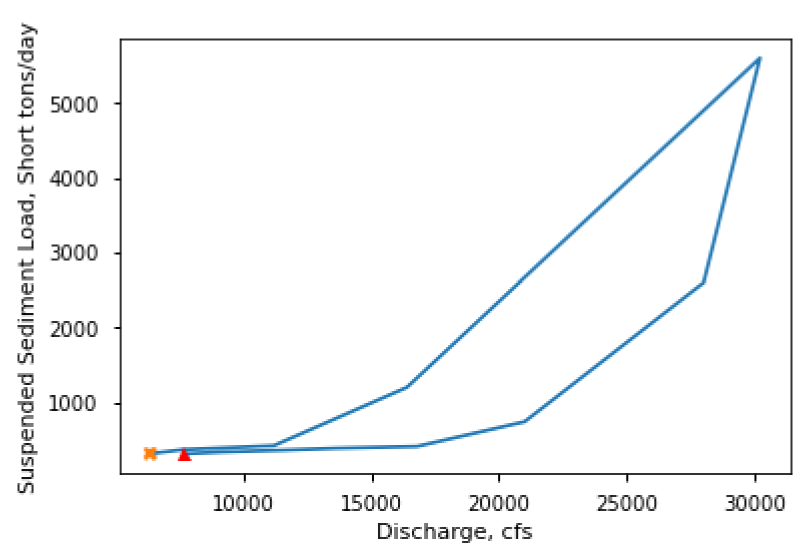


With our two variables plotted on the same time axis, it is possible to see that they peak on the same day, but sediment load increases and decreases faster than discharge does. Let's see how that looks when we plot the variables against each other instead of over time.

In this graph, I've plotted discharge on the X axis, and sediment load on the Y axis. There is an 'X' at the start of the time series, and a triangle at the end.

[8]:
```
Q = may.loc[:,'discharge'].values
load = may.loc[:,'sed_load'].values
fig, ax = plt.subplots()
plt.xlabel('Discharge, cfs')
plt.ylabel('Suspended Sediment Load, Short tons/day')
ax.plot(Q,load)
ax.plot(Q[0], load[0], 'X') # Plot an X at the first value in each series.
ax.plot(Q[-1], load[-1], 'r^') # plot a triangle at the end
```

[8]:
```
[<matplotlib.lines.Line2D at 0x1b256f22a60>]
```



In this case, the hysteresis moves clockwise, contrary to our hypothesis at the top of the page! Is this pattern common? Do the daily mean values hide lags that might otherwise be seen in more frequently sampled data? Or perhaps large rivers behave differently than smaller streams?

Use these techniques to explore a little and let us know what you find out!

16.6 Draw Map Demo

```
[1]: import hydrofunctions as hf
```

```
[2]: hf.draw_map()
```

```
[2]: <IPython.core.display.HTML object>
```

16.7 Example Plots

This notebook illustrates the different types of graph you can produce with Hydrofunctions. We have:

- hydrograph
- flow duration
- cycleplot
- histogram

We'll start with the usual imports:

```
[1]: import hydrofunctions as hf
     import pandas as pd
     %matplotlib inline
     hf.__version__
     pd.__version__
```

```
[1]: '0.24.2'
```

We're going to use two datasets in the following examples. The first dataset was collected at two sites along the Shenandoah River.

```
[2]: sites = ['01634000', '01632000']  # the first is downstream of the second.
     start = '2008-01-01'
     end = '2018-01-01'
     service = 'dv'
     # Request our data.
     request = hf.NWIS(sites, service, start, end, file='graphing-1.parquet')
     request # Verify that the data request went fine.
```

```
Reading data from graphing-1.parquet
```

```
[2]: USGS:01632000: N F SHENANDOAH RIVER AT COOTES STORE, VA
         00010: <Day>  Temperature, water, degrees Celsius
         00060: <Day>  Discharge, cubic feet per second
         00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
     ↪at 25 degrees Celsius
```

(continues on next page)

(continued from previous page)

```
USGS:01634000: N F SHENANDOAH RIVER NEAR STRASBURG, VA
    00060: <Day>  Discharge, cubic feet per second
    00095: <2 * Days>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    00400: <Day>  pH, water, unfiltered, field, standard units
    63680: <Day>  Turbidity, water, unfiltered, monochrome near infra-red LED light, 780-
↪900 nm, detection angle 90 +-2.5 degrees, formazin nephelometric units (FNU)
Start: 2008-01-01 00:00:00+00:00
End:   2018-01-01 00:00:00+00:00
```

This second dataset is for two years of data collected every five minutes, with five different parameters collected. This request may take a while the first time! After that, Hydrofunctions will automatically access the data from the parquet file.

[3]:
```
site = '01581752'
request2 = hf.NWIS(site, 'iv', '2016-01-01', '2018-12-31', file='graphing-2.parquet')
request2
```

```
Reading data from graphing-2.parquet
```

[3]:
```
USGS:01581752: PLUMTREE RUN NEAR BEL AIR, MD
    00010: <5 * Minutes>  Temperature, water, degrees Celsius
    00060: <5 * Minutes>  Discharge, cubic feet per second
    00065: <5 * Minutes>  Gage height, feet
    00095: <5 * Minutes>  Specific conductance, water, unfiltered, microsiemens per␣
↪centimeter at 25 degrees Celsius
    63680: <5 * Minutes>  Turbidity, water, unfiltered, monochrome near infra-red LED␣
↪light, 780-900 nm, detection angle 90 +-2.5 degrees, formazin nephelometric units (FNU)
Start: 2016-01-01 05:00:00+00:00
End:   2019-01-01 04:55:00+00:00
```

Create two nice clean dataframes that we'll plot with.

[4]:
```
Q = request.df('q') # Q stands for discharge
T = request2.df('00010') # T stands for water temperature.
```

Rename the columns to something easier to remember.

[5]:
```
Q = Q.rename(columns={"USGS:01632000:00060:00003": "Upstream", "USGS:01634000:00060:00003
↪": "Downstream"})
Q.head(2)
```

[5]:
```
                           Upstream  Downstream
datetimeUTC
2008-01-01 00:00:00+00:00     112.0       330.0
2008-01-02 00:00:00+00:00     109.0       320.0
```

[6]:
```
T = T.rename(columns={"USGS:01581752:00010:00000": "Plumtree"})
T.head(2)
```

[6]:
```
                           Plumtree
datetimeUTC
2016-01-01 05:00:00+00:00       8.7
2016-01-01 05:05:00+00:00       8.7
```

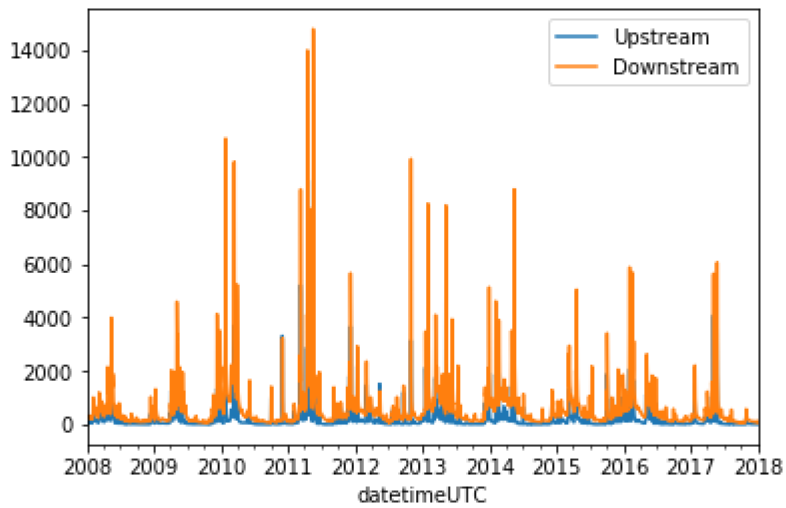
16.7.1 Plotting a hydrograph

Hydrographs can be produced simply by using the built-in .plot() method of our dataframe.

[7]:
```
Q.plot()
```

```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a8d3fcbe48>
```

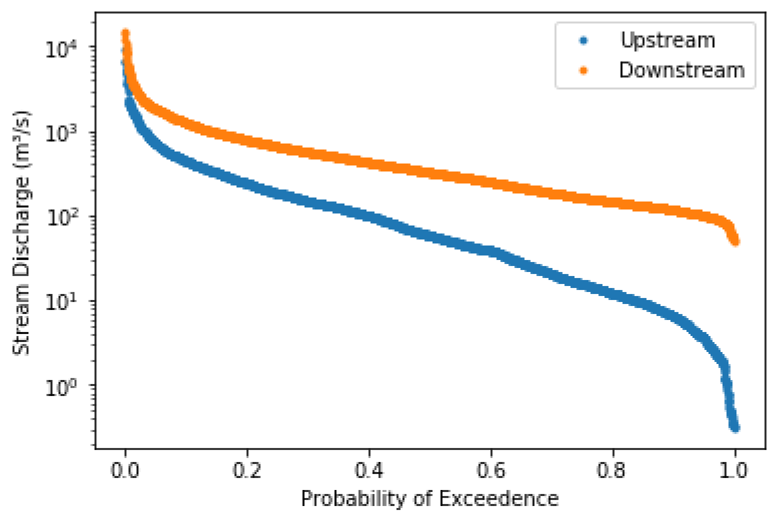


16.7.2 Plotting a flow duration chart

Flow duration charts are function included in Hydrofunctions. We'll use the 'linear' option to scale the X axis.

[8]:
```
hf.flow_duration(Q, xscale='linear')
```

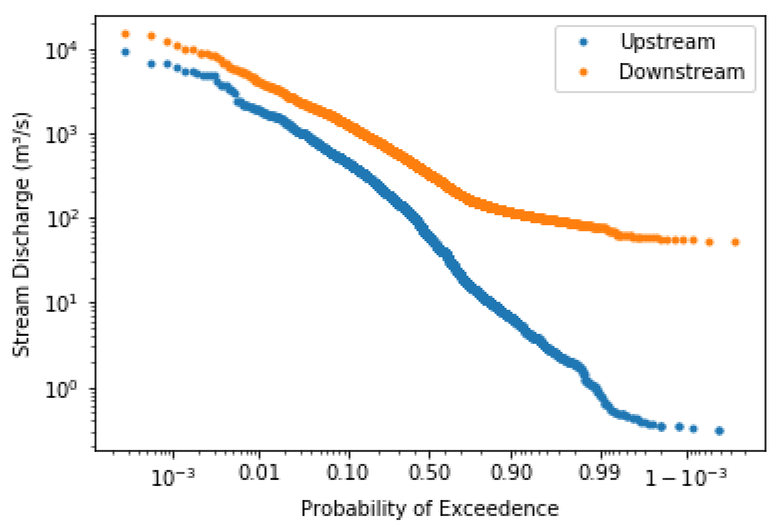
[8]:
```
(<Figure size 432x288 with 1 Axes>,
 <matplotlib.axes._subplots.AxesSubplot at 0x1a8d41faa90>)
```



Now let's plot the same data using the default 'logit' scale

```
[9]: hf.flow_duration(Q, xscale='logit')
```

```
[9]: (<Figure size 432x288 with 1 Axes>,
      <matplotlib.axes._subplots.AxesSubplot at 0x1a8d43d6c88>)
```



16.7.3 Plotting a cycleplot chart

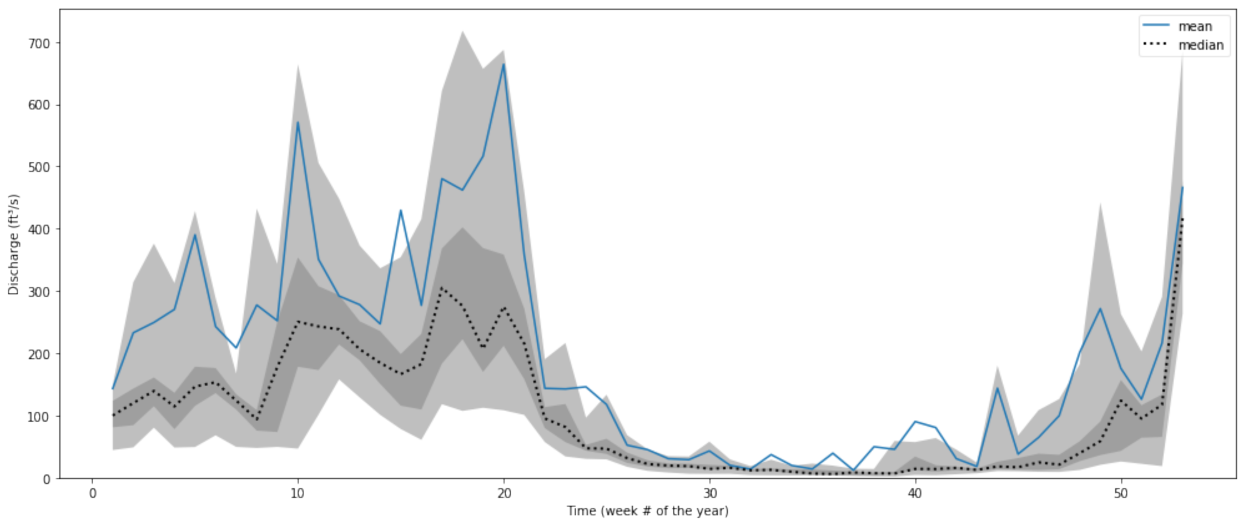
The cycleplot helps to illustrate various natural cycles in your dataset. It plots a single column of data so that the data value is on the Y axis, and time is on the X axis. The time axis is set to show either an annual cycle (shown below), a weekly cycle, or a diurnal cycle (default).

In our example below, the ten years of daily data are plotted over the course of a year, and all of the values within a week are averaged and plotted by week number. The median and mean for each week are shown with lines, while the 20th, 40th, 60th, and 80th percentile are shown with shaded fills.

For our dataset, you can see the lower flows that start occuring around week 25 (early June) and last through the summer to week 45 (start of November). You can also bin values by day ('annual-day') or by month ('annual-month'). The smaller the bins, the greater the variation.

[10]:
```
hf.cycleplot(Q.loc[:,'Upstream'], 'annual-week')
```

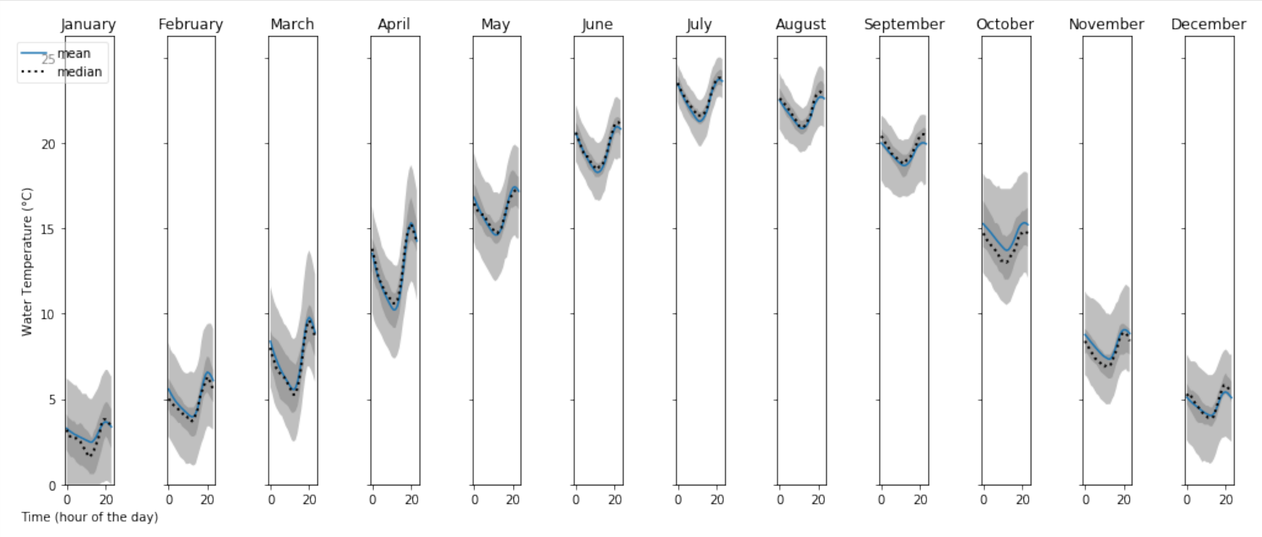
[10]:
```
(<Figure size 1008x432 with 1 Axes>,
 array([<matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D588E518>],
       dtype=object))
```



In this chart, we plot the hourly temperature for each month of the year. Note that the times are UTC, so midnight for this site should occur at hour 6.

[11]:
```
hf.cycleplot(T, 'diurnal', compare='month', y_label="Water Temperature (°C)")
```

[11]:
```
(<Figure size 1008x432 with 12 Axes>,
 array([<matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D58505C0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D4391860>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D580EB38>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5889E10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D58D8128>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5A02400>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5A2D5F8>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5A53898>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5A7CBA8>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5AA5E80>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5C85198>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5CAC470>],
       dtype=object))
```



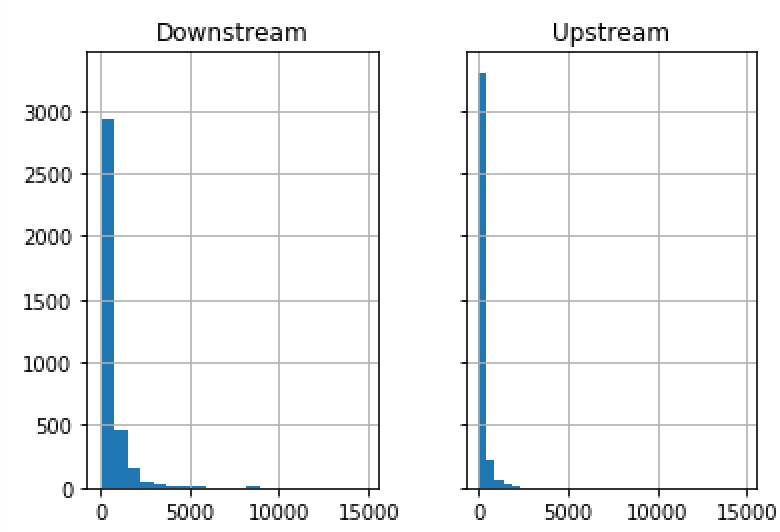
16.7.4 Plotting a histogram

Dataframes have a built-in histogram function.

We'll plot each histogram with 20 bins (default is 10;) and we will have both sites share the same scale for the x axis (discharge) and y axis (count)

[12]:
```
Q.hist(bins=20, sharex=True, sharey=True)
```

[12]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D679AEB8>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x000001A8D5E132E8>]],
      dtype=object)
```

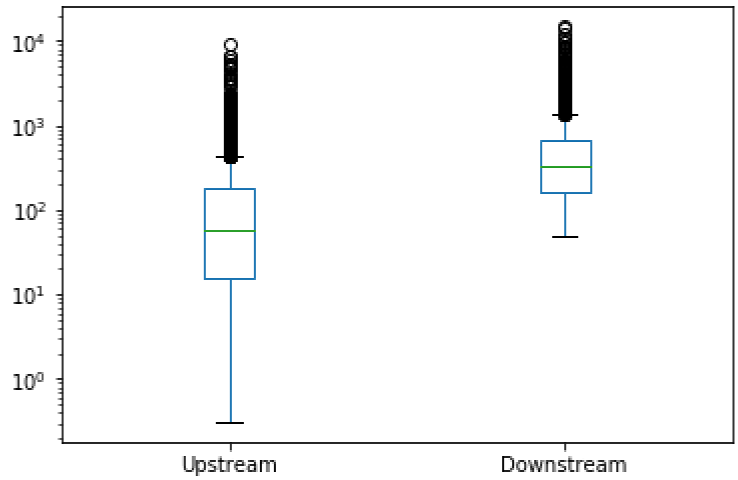


16.7.5 Box plots

Box plots are a great way to illustrate a distribution. In this chart, we'll use boxplots to compare the two sites along the Shenandoah River:

[13]:
```
Q.plot.box(logy=True)
```

[13]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a8d60c3ac8>
```



16.8 A Hydrofunctions Tutorial

This guide will step you through the basics of using hydrofunctions. Read more in our User's Guide, or visit us on GitHub!

16.8.1 Installation

The first step before using hydrofunctions is to get it installed on your system. For scientific computing, we highly recommend using the free, open-source Anaconda distribution to load and manage all of your Python tools and packages. Once you have downloaded and installed Anaconda, or if you already have Python set up on your computer, your next step is to use the pip tool from your operating system's command line to download hydrofunctions.

In Linux: `$ pip install hydrofunctions`

In Windows: `C:\MyPythonWorkspace\> pip install hydrofunctions`

If you have any difficulties, visit our Installation page in the User's Guide.

16.8.2 Getting started in Python

From here on out, we will assume that you have installed hydrofunctions and you are working at a Python command prompt, perhaps in ipython or in a Jupyter notebook.

[1]:
```
# The first step is to import hydrofunctions so that we can use it here.

import hydrofunctions as hf
# This second line allows us to automatically plot diagrams in this notebook.
%matplotlib inline
```

Get data for a USGS streamflow gage

The USGS runs an amazing web service called the National Water Information System. Our first task is to download daily mean discharge data for a stream called Herring Run. Set the start date and the end date for our download, and use the site number for Herring Run ('01585200') to specify which stream gage we want to collect data from. Once we request the data, it will be saved to a file. If the file is already present, we'll just use that instead of requesting it from the NWIS.

You can visit the NWIS website or use hydrocloud.org to find the site number for a stream gage near you.

[2]:
```
start = '2017-06-01'
end = '2017-07-14'
herring = hf.NWIS('01585200', 'dv', start, end, file='herring_july.parquet')
herring # This last command will print out a description of what we have.
```

```
Reading data from herring_july.parquet
```

[2]:
```
USGS:01585200: WEST BRANCH HERRING RUN AT IDLEWYLDE, MD
    00060: <Day>  Discharge, cubic feet per second
Start: 2017-06-01 00:00:00+00:00
End:   2017-07-14 00:00:00+00:00
```

16.8.3 Viewing our data

There are several ways to view our data. Try herring.json() or better still, use a Pandas dataframe:

[3]:
```
herring.df()
```

[3]:
```
                          USGS:01585200:00060:00003_qualifiers  \
datetimeUTC
2017-06-01 00:00:00+00:00                                    A
2017-06-02 00:00:00+00:00                                    A
2017-06-03 00:00:00+00:00                                    A
2017-06-04 00:00:00+00:00                                    A
2017-06-05 00:00:00+00:00                                    A
2017-06-06 00:00:00+00:00                                    A
2017-06-07 00:00:00+00:00                                    A
2017-06-08 00:00:00+00:00                                    A
2017-06-09 00:00:00+00:00                                    A
2017-06-10 00:00:00+00:00                                    A
2017-06-11 00:00:00+00:00                                    A
2017-06-12 00:00:00+00:00                                    A
```

(continues on next page)

(continued from previous page)

```
2017-06-13 00:00:00+00:00                                    A
2017-06-14 00:00:00+00:00                                    A
2017-06-15 00:00:00+00:00                                    A
2017-06-16 00:00:00+00:00                                    A
2017-06-17 00:00:00+00:00                                    A
2017-06-18 00:00:00+00:00                                    A
2017-06-19 00:00:00+00:00                                    A
2017-06-20 00:00:00+00:00                                    A
2017-06-21 00:00:00+00:00                                    A
2017-06-22 00:00:00+00:00                                    A
2017-06-23 00:00:00+00:00                                    A
2017-06-24 00:00:00+00:00                                    A
2017-06-25 00:00:00+00:00                                    A
2017-06-26 00:00:00+00:00                                    A
2017-06-27 00:00:00+00:00                                    A
2017-06-28 00:00:00+00:00                                    A
2017-06-29 00:00:00+00:00                                    A
2017-06-30 00:00:00+00:00                                    A
2017-07-01 00:00:00+00:00                                    A
2017-07-02 00:00:00+00:00                                    A
2017-07-03 00:00:00+00:00                                    A
2017-07-04 00:00:00+00:00                                    A
2017-07-05 00:00:00+00:00                                    A
2017-07-06 00:00:00+00:00                                    A
2017-07-07 00:00:00+00:00                                    A
2017-07-08 00:00:00+00:00                                    A
2017-07-09 00:00:00+00:00                                    A
2017-07-10 00:00:00+00:00                                    A
2017-07-11 00:00:00+00:00                                    A
2017-07-12 00:00:00+00:00                                    A
2017-07-13 00:00:00+00:00                                    A
2017-07-14 00:00:00+00:00                                    A

                           USGS:01585200:00060:00003
datetimeUTC
2017-06-01 00:00:00+00:00                       0.71
2017-06-02 00:00:00+00:00                       0.64
2017-06-03 00:00:00+00:00                       0.61
2017-06-04 00:00:00+00:00                       0.58
2017-06-05 00:00:00+00:00                       1.95
2017-06-06 00:00:00+00:00                       0.66
2017-06-07 00:00:00+00:00                       0.62
2017-06-08 00:00:00+00:00                       0.55
2017-06-09 00:00:00+00:00                       0.51
2017-06-10 00:00:00+00:00                       0.48
2017-06-11 00:00:00+00:00                       0.48
2017-06-12 00:00:00+00:00                       0.45
2017-06-13 00:00:00+00:00                       0.43
2017-06-14 00:00:00+00:00                       0.46
2017-06-15 00:00:00+00:00                       0.38
2017-06-16 00:00:00+00:00                       0.39
2017-06-17 00:00:00+00:00                       0.39
```

(continues on next page)

(continued from previous page)

```
2017-06-18 00:00:00+00:00                        0.36
2017-06-19 00:00:00+00:00                        7.48
2017-06-20 00:00:00+00:00                        1.08
2017-06-21 00:00:00+00:00                        1.30
2017-06-22 00:00:00+00:00                        0.47
2017-06-23 00:00:00+00:00                        1.22
2017-06-24 00:00:00+00:00                        3.04
2017-06-25 00:00:00+00:00                        0.37
2017-06-26 00:00:00+00:00                        0.32
2017-06-27 00:00:00+00:00                        0.31
2017-06-28 00:00:00+00:00                        0.28
2017-06-29 00:00:00+00:00                        0.29
2017-06-30 00:00:00+00:00                        0.27
2017-07-01 00:00:00+00:00                        6.20
2017-07-02 00:00:00+00:00                        0.51
2017-07-03 00:00:00+00:00                        0.35
2017-07-04 00:00:00+00:00                        2.63
2017-07-05 00:00:00+00:00                        0.44
2017-07-06 00:00:00+00:00                       15.60
2017-07-07 00:00:00+00:00                        1.58
2017-07-08 00:00:00+00:00                        0.45
2017-07-09 00:00:00+00:00                        0.35
2017-07-10 00:00:00+00:00                        0.33
2017-07-11 00:00:00+00:00                        0.33
2017-07-12 00:00:00+00:00                        0.30
2017-07-13 00:00:00+00:00                        0.28
2017-07-14 00:00:00+00:00                        7.55
```

Pandas' dataframes give you access to hundreds of useful methods, such as .describe() and .plot():

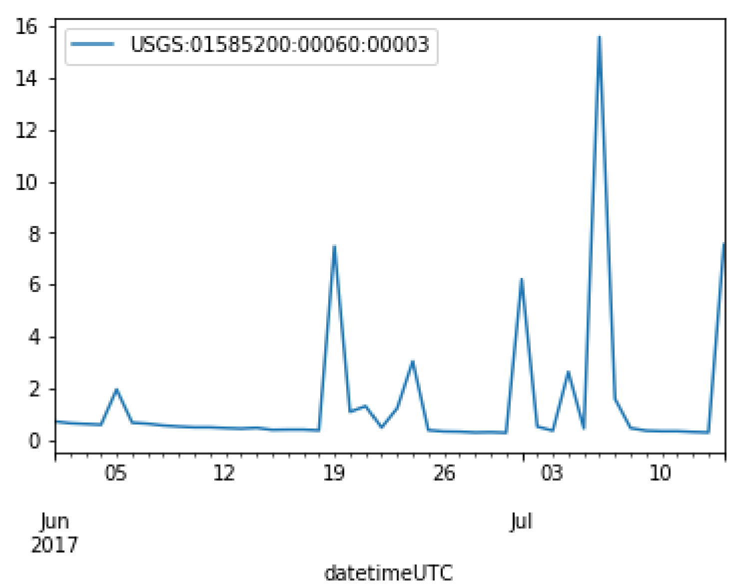
[4]:
```
herring.df().describe()
```

[4]:
```
       USGS:01585200:00060:00003
count                  44.000000
mean                    1.454091
std                     2.792200
min                     0.270000
25%                     0.357500
50%                     0.475000
75%                     0.802500
max                    15.600000
```

[5]:
```
herring.df().plot()
```

```
C:\Users\Marty\Anaconda3\envs\py37hfdev\lib\site-packages\pandas\core\arrays\datetimes.
↪py:1172: UserWarning: Converting to PeriodArray/Index representation will drop␣
↪timezone information.
  "will drop timezone information.", UserWarning)
```

[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x22f6664b198>
```



16.8.4 Multiple sites, other parameters

It's possible to load data from several different sites at the same time, and you aren't limited to just stream discharge.

Requests can use lists of sites:

[6]:
```
sites = ['380616075380701','394008077005601']
```

The NWIS can deliver data as daily mean values ('dv') or as instantaneous values ('iv') that can get collected as often as every five minutes!

[7]:
```
service = 'iv'
```

Depending on the site, the USGS collects groundwater levels ('72019'), stage ('00065'), precipitation, and more!

[8]:
```
pcode = '72019'
```

Now we'll create a new dataset called 'groundwater' using the values we set up above.

Since one of the parameters gets collected every 30 minutes, and the other gets collected every 15 minutes, Hydrofunctions will interpolate values for every 15 minutes for every parameter we've requested. These interpolated values will be marked with a special `hf.interpolate` flag in the qualifiers column.

[9]:
```
groundwater = hf.NWIS(sites, service, '2018-01-01', '2018-01-31', parameterCd=pcode,␣
↪file='groundwater.parquet')
groundwater
```

```
Reading data from groundwater.parquet
```

[9]:
```
USGS:380616075380701: SO Cf   2
    72019: <15 * Minutes>  Depth to water level, feet below land surface
USGS:394008077005601: CL Ad  47
    72019: <30 * Minutes>  Depth to water level, feet below land surface
Start: 2018-01-01 05:00:00+00:00
End:   2018-02-01 04:45:00+00:00
```

Calculate the mean for every data column:

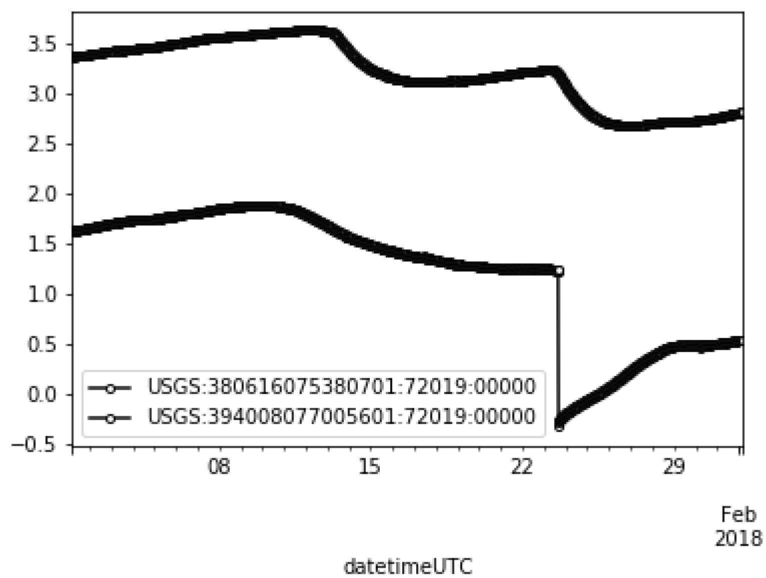
[10]:
```
groundwater.df().mean()
```

[10]:
```
USGS:380616075380701:72019:00000    1.215141
USGS:394008077005601:72019:00000    3.206111
dtype: float64
```

View the data in a specially styled graph!

[11]:
```
groundwater.df().plot(marker='o', mfc='white', ms=4, mec='black', color='black')
```

[11]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x22f667facc0>
```



16.8.5 Learning more

hydrofunctions comes with a variety of built-in help functions that you can access from the command line, in addition to our online User's Guide.

Jupyter Notebooks provide additional helpful shortcuts, such as code completion. This will list all of the available methods for an object just by hitting like this: `herring.<TAB>` this is equivalent to using dir(herring) to list all of the methods available to you.

Typing help() or dir() for different objects allows you to access additional information. `help(hf.NWIS)` is equivalent to just using a question mark like this: `?hf.NWIS`

[12]:
```
help(hf.NWIS)
```

```
Help on class NWIS in module hydrofunctions.station:

class NWIS(Station)
 |  NWIS(site=None, service='dv', start_date=None, end_date=None, stateCd=None,␣
↪countyCd=None, bBox=None, parameterCd='all', period=None, file=None)
 |
 |  A class for working with data from the USGS NWIS service.
```

(continues on next page)

(continued from previous page)

```
|
|   description
|
|   Args:
|       site (str or list of strings):
|           a valid site is '01585200' or ['01585200', '01646502']. Default is
|           None. If site is not specified, you will need to select sites using
|           stateCd or countyCd.
|
|       service (str):
|           can either be 'iv' or 'dv' for instantaneous or daily data.
|           'dv'(default): daily values. Mean value for an entire day.
|           'iv': instantaneous value measured at this time. Also known
|           as 'Real-time data'. Can be measured as often as every
|           five minutes by the USGS. 15 minutes is more typical.
|
|       start_date (str):
|          should take on the form 'yyyy-mm-dd'
|
|       end_date (str):
|           should take on the form 'yyyy-mm-dd'
|
|       stateCd (str):
|           a valid two-letter state postal abbreviation, such as 'MD'. Default
|           is None. Selects all stations in this state. Because this type of
|           site selection returns a large number of sites, you should limit
|           the amount of data requested for each site.
|
|       countyCd (str or list of strings):
|           a valid county FIPS code. Default is None. Requests all stations
|           within the county or list of counties. See https://en.wikipedia.org/wiki/
↪FIPS_county_code
|           for an explanation of FIPS codes.
|
|       bBox (str, list, or tuple):
|           a set of coordinates that defines a bounding box.
|               * Coordinates are in decimal degrees.
|               * Longitude values are negative (west of the prime meridian).
|               * Latitude values are positive (north of the equator).
|               * comma-delimited, no spaces, if provided as a string.
|               * The order of the boundaries should be: "West,South,East,North"
|               * Example: "-83.000000,36.500000,-81.000000,38.500000"
|
|       parameterCd (str or list of strings):
|           NWIS parameter code. Usually a five digit code. Default is 'all'.
|           A valid code can also be given as a list: parameterCd=['00060','00065']
|           This will request data for this parameter.
|
|               * if value is 'all', or no value is submitted, then NWIS will
↪     return every parameter collected at this site. (default option)
|               * stage: '00065'
|               * discharge: '00060'
```

(continues on next page)

(continued from previous page)

```
|               * not all sites collect all parameters!
|               * See https://nwis.waterdata.usgs.gov/usa/nwis/pmcodes for full list
|
|       period (str):
|           NWIS period code. Default is None.
|               * Format is "PxxD", where xx is the number of days before ␣
↪ today, with a maximum of 999 days accepted.
|               * Either use start_date or period, but not both.
|
|   Method resolution order:
|       NWIS
|       Station
|       builtins.object
|
|   Methods defined here:
|
|   __init__(self, site=None, service='dv', start_date=None, end_date=None, stateCd=None,
↪ countyCd=None, bBox=None, parameterCd='all', period=None, file=None)
|       Initialize self.  See help(type(self)) for accurate signature.
|
|   __repr__(self)
|       Return repr(self).
|
|   df(self, *args)
|       Return a subset of columns from the dataframe.
|
|       Args:
|           '': If no args are provided, the entire dataframe will be returned.
|
|           str 'all': the entire dataframe will be returned.
|
|           str 'data': all of the parameters will be returned, with no flags.
|
|           str 'flags': Only the _qualifier flags will be returned. Unless the ␣
↪   flags arg is provided, only data columns will be returned. Visit             https:/
↪/waterdata.usgs.gov/usa/nwis/uv?codes_help#dv_cd1 to see a             more complete␣
↪listing of possible codes.
|
|           str 'discharge' or 'q': discharge columns ('00060') will be returned.
|
|           str 'stage': Gauge height columns ('00065') will be returned.
|
|           int any five digit number: any matching parameter columns will be returned.
↪'00065' returns stage, for example.
|
|           int any eight to twelve digit number: any matching stations will be returned.
|
|   get_data(self)
|
|   read(self, file)
|
|   save(self, file)
```

(continues on next page)

(continued from previous page)

```
 |
 |  ----------------------------------------------------------------------
 |  Data descriptors inherited from Station:
 |
 |  __dict__
 |      dictionary for instance variables (if defined)
 |
 |  __weakref__
 |      list of weak references to the object (if defined)
 |
 |  ----------------------------------------------------------------------
 |  Data and other attributes inherited from Station:
 |
 |  station_dict = {}
```

16.8.6 Advanced techniques

Download data for a large number of sites

[13]:
```
sites = ['07227500', '07228000', '07235000', '07295500', '07297910', '07298500',
↪'07299540',
         '07299670', '07299890', '07300000', '07301300', '07301410', '07308200',
↪'07308500', '07311600',
         '07311630', '07311700', '07311782', '07311783', '07311800', '07311900',
↪'07312100', '07312200',
         '07312500', '07312700', '07314500', '07314900', '07315200', '07315500',
↪'07342465', '07342480',
         '07342500', '07343000', '07343200', '07343500', '07344210', '07344500',
↪'07346000']
mult = hf.NWIS(sites, "dv", "2018-01-01", "2018-01-31", file='mult.parquet')
print('No. sites: {}'.format(len(sites)))
```

```
Reading data from mult.parquet
No. sites: 38
```

This will calculate the mean value for each site.

[14]:
```
mult.df().mean()
```

[14]:
```
USGS:07227500:00010:00001       6.848387
USGS:07227500:00010:00002       0.661290
USGS:07227500:00010:00003       3.254839
USGS:07227500:00060:00003      51.880645
USGS:07227500:00065:00003       1.233710
USGS:07227500:00095:00001    5968.333333
USGS:07227500:00095:00002    5648.571429
USGS:07227500:00095:00003    5788.333333
USGS:07228000:00060:00003      57.938710
USGS:07228000:00065:00003       1.429032
USGS:07235000:00060:00001       0.657742
```

(continues on next page)

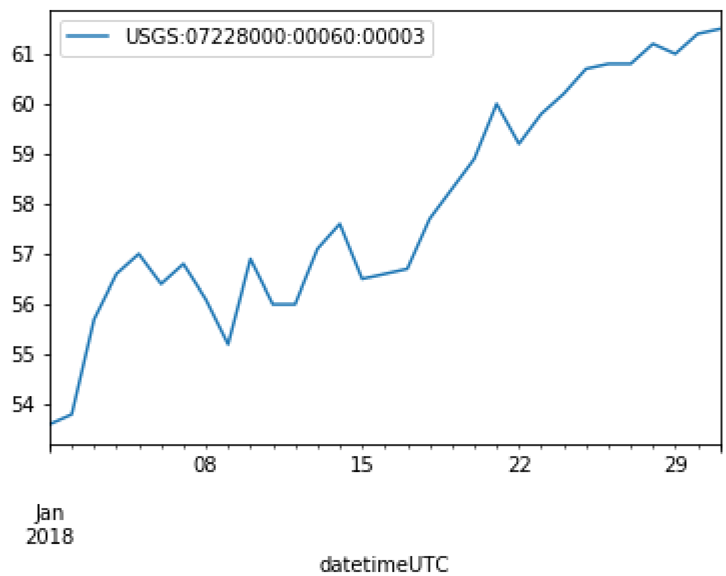
(continued from previous page)

```
USGS:07235000:00060:00002       0.462581
USGS:07235000:00060:00003       0.558065
USGS:07235000:00065:00003       4.381613
USGS:07295500:00060:00001       0.000000
USGS:07295500:00060:00002       0.000000
USGS:07295500:00060:00003       0.000000
USGS:07295500:00065:00001       0.606452
USGS:07295500:00065:00002       0.600000
USGS:07295500:00065:00003       0.600000
USGS:07297910:00060:00001      24.383871
USGS:07297910:00060:00002      14.901290
USGS:07297910:00060:00003      19.103226
USGS:07297910:00065:00001       6.003148
USGS:07297910:00065:00002       5.941111
USGS:07297910:00065:00003       5.967778
USGS:07298500:00060:00001      10.042581
USGS:07298500:00060:00002       7.213548
USGS:07298500:00060:00003       8.730968
USGS:07298500:00065:00001       4.830000
                                   ...
USGS:07342500:00065:00002       2.810645
USGS:07342500:00065:00003       2.835161
USGS:07343000:00060:00001      68.566452
USGS:07343000:00060:00002      14.491935
USGS:07343000:00060:00003      36.442258
USGS:07343000:00065:00001       1.601935
USGS:07343000:00065:00002       1.318065
USGS:07343000:00065:00003       1.468065
USGS:07343200:00060:00003      76.200000
USGS:07343500:00060:00001      52.474194
USGS:07343500:00060:00002      34.600000
USGS:07343500:00060:00003      43.445161
USGS:07343500:00065:00001       2.986452
USGS:07343500:00065:00002       2.609839
USGS:07343500:00065:00003       2.794032
USGS:07344210:00060:00001    1304.306452
USGS:07344210:00060:00002    1208.258065
USGS:07344210:00065:00001      13.347419
USGS:07344210:00065:00002      12.885161
USGS:07344210:00065:00003      13.119032
USGS:07344500:00060:00001      10.138065
USGS:07344500:00060:00002       6.139677
USGS:07344500:00060:00003       7.929032
USGS:07344500:00065:00001       5.570323
USGS:07344500:00065:00002       5.401613
USGS:07344500:00065:00003       5.482258
USGS:07346000:00060:00001     287.387097
USGS:07346000:00060:00002     283.548387
USGS:07346000:00060:00003     285.387097
USGS:07346000:00065:00003       8.185806
Length: 164, dtype: float64
```

Plot just the discharge data for one site in the list:

```
[15]: mult.df('07228000', 'discharge').plot()
```

```
[15]: <matplotlib.axes._subplots.AxesSubplot at 0x22f66758780>
```



List some of the data available to you in a dataframe.

```
[16]: mult
```

```
[16]: USGS:07227500: Canadian Rv nr Amarillo, TX
    00010: <Day>  Temperature, water, degrees Celsius
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
    00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
↪at 25 degrees Celsius
USGS:07228000: Canadian Rv nr Canadian, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07235000: Wolf Ck at Lipscomb, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07295500: Tierra Blanca Ck abv Buffalo Lk nr Umbarger, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07297910: Pr Dog Twn Fk Red Rv nr Wayside, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07298500: Pr Dog Twn Fk Red Rv nr Brice, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07299540: Pr Dog Twn Fk Red Rv nr Childress, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
```

(continues on next page)

(continued from previous page)

```
USGS:07299670: Groesbeck Ck at SH 6 nr Quanah, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07299890: Lelia Lk Ck bl Bell Ck nr Hedley, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07300000: Salt Fk Red Rv nr Wellington, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07301300: N Fk Red Rv nr Shamrock, TX
    00060: <Day>  Discharge, cubic feet per second
USGS:07301410: Sweetwater Ck nr Kelton, TX
    00060: <Day>  Discharge, cubic feet per second
USGS:07308200: Pease Rv nr Vernon, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07308500: Red Rv nr Burkburnett, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07311700: N Wichita Rv nr Truscott, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07311800: S Wichita Rv nr Benjamin, TX
    00010: <Day>  Temperature, water, degrees Celsius
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
    00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
↪at 25 degrees Celsius
USGS:07311900: Wichita Rv nr Seymour, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07312100: Wichita Rv nr Mabelle, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07312200: Beaver Ck nr Electra, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <2 * Days>  Gage height, feet
USGS:07312500: Wichita Rv at Wichita Falls, TX
    00010: <Day>  Temperature, water, degrees Celsius
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
    00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
↪at 25 degrees Celsius
USGS:07312700: Wichita Rv nr Charlie, TX
    00010: <Day>  Temperature, water, degrees Celsius
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
    00095: <Day>  Specific conductance, water, unfiltered, microsiemens per centimeter␣
↪at 25 degrees Celsius
USGS:07314500: Little Wichita Rv nr Archer City, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
```

(continues on next page)

(continued from previous page)

```
USGS:07314900: Little Wichita Rv abv Henrietta, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07315200: E Fk Little Wichita Rv nr Henrietta, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07315500: Red Rv nr Terral, OK
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07342465: S Sulphur Rv at Commerce, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07342480: Middle Sulphur Rv at Commerce, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07342500: S Sulphur Rv nr Cooper, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07343000: N Sulphur Rv nr Cooper, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07343200: Sulphur Rv nr Talco, TX
    00060: <Day>  Discharge, cubic feet per second
USGS:07343500: White Oak Ck nr Talco, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07344210: Sulphur Rv nr Texarkana, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07344500: Big Cypress Ck nr Pittsburg, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
USGS:07346000: Big Cypress Bayou nr Jefferson, TX
    00060: <Day>  Discharge, cubic feet per second
    00065: <Day>  Gage height, feet
Start: 2018-01-01 00:00:00+00:00
End:   2018-01-31 00:00:00+00:00
```

Create a table of discharge data

`.head()` only show the first five
`.tail()` only show the last five

[17]:
```
mult.df('discharge').head()
```

[17]:
```
                           USGS:07227500:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                       43.0
2018-01-02 00:00:00+00:00                       52.9
```

(continues on next page)

(continued from previous page)

```
2018-01-03 00:00:00+00:00                          51.7
2018-01-04 00:00:00+00:00                          50.7
2018-01-05 00:00:00+00:00                          53.8

                           USGS:07228000:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                          53.6
2018-01-02 00:00:00+00:00                          53.8
2018-01-03 00:00:00+00:00                          55.7
2018-01-04 00:00:00+00:00                          56.6
2018-01-05 00:00:00+00:00                          57.0

                           USGS:07235000:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                          0.55
2018-01-02 00:00:00+00:00                          0.61
2018-01-03 00:00:00+00:00                          0.55
2018-01-04 00:00:00+00:00                          0.51
2018-01-05 00:00:00+00:00                          0.51

                           USGS:07235000:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                          0.40
2018-01-02 00:00:00+00:00                          0.36
2018-01-03 00:00:00+00:00                          0.46
2018-01-04 00:00:00+00:00                          0.46
2018-01-05 00:00:00+00:00                          0.46

                           USGS:07235000:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                          0.46
2018-01-02 00:00:00+00:00                          0.48
2018-01-03 00:00:00+00:00                          0.51
2018-01-04 00:00:00+00:00                          0.48
2018-01-05 00:00:00+00:00                          0.49

                           USGS:07295500:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                           0.0
2018-01-02 00:00:00+00:00                           0.0
2018-01-03 00:00:00+00:00                           0.0
2018-01-04 00:00:00+00:00                           0.0
2018-01-05 00:00:00+00:00                           0.0

                           USGS:07295500:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                           0.0
2018-01-02 00:00:00+00:00                           0.0
2018-01-03 00:00:00+00:00                           0.0
2018-01-04 00:00:00+00:00                           0.0
2018-01-05 00:00:00+00:00                           0.0
```

(continues on next page)

(continued from previous page)

```
                          USGS:07295500:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                        0.0
2018-01-02 00:00:00+00:00                        0.0
2018-01-03 00:00:00+00:00                        0.0
2018-01-04 00:00:00+00:00                        0.0
2018-01-05 00:00:00+00:00                        0.0

                          USGS:07297910:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                       30.9
2018-01-02 00:00:00+00:00                       35.9
2018-01-03 00:00:00+00:00                       43.3
2018-01-04 00:00:00+00:00                       30.9
2018-01-05 00:00:00+00:00                       30.9

                          USGS:07297910:00060:00002  ...  \
datetimeUTC                                          ...
2018-01-01 00:00:00+00:00                       25.9  ...
2018-01-02 00:00:00+00:00                       30.9  ...
2018-01-03 00:00:00+00:00                       18.8  ...
2018-01-04 00:00:00+00:00                       14.7  ...
2018-01-05 00:00:00+00:00                       16.0  ...

                          USGS:07343500:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                       35.8
2018-01-02 00:00:00+00:00                       32.5
2018-01-03 00:00:00+00:00                       30.4
2018-01-04 00:00:00+00:00                       27.1
2018-01-05 00:00:00+00:00                       23.6

                          USGS:07343500:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                       37.7
2018-01-02 00:00:00+00:00                       34.0
2018-01-03 00:00:00+00:00                       31.6
2018-01-04 00:00:00+00:00                       28.8
2018-01-05 00:00:00+00:00                       25.5

                          USGS:07344210:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                      774.0
2018-01-02 00:00:00+00:00                     1360.0
2018-01-03 00:00:00+00:00                     1510.0
2018-01-04 00:00:00+00:00                     1530.0
2018-01-05 00:00:00+00:00                     1530.0

                          USGS:07344210:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                      756.0
2018-01-02 00:00:00+00:00                      763.0
```

(continues on next page)

(continued from previous page)

```
2018-01-03 00:00:00+00:00                      1360.0
2018-01-04 00:00:00+00:00                      1510.0
2018-01-05 00:00:00+00:00                      1520.0

                           USGS:07344500:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                        4.82
2018-01-02 00:00:00+00:00                        5.27
2018-01-03 00:00:00+00:00                        6.10
2018-01-04 00:00:00+00:00                        7.17
2018-01-05 00:00:00+00:00                        7.39

                           USGS:07344500:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                        4.00
2018-01-02 00:00:00+00:00                        3.70
2018-01-03 00:00:00+00:00                        4.97
2018-01-04 00:00:00+00:00                        6.10
2018-01-05 00:00:00+00:00                        6.61

                           USGS:07344500:00060:00003  \
datetimeUTC
2018-01-01 00:00:00+00:00                        4.44
2018-01-02 00:00:00+00:00                        4.32
2018-01-03 00:00:00+00:00                        5.29
2018-01-04 00:00:00+00:00                        6.66
2018-01-05 00:00:00+00:00                        7.02

                           USGS:07346000:00060:00001  \
datetimeUTC
2018-01-01 00:00:00+00:00                       285.0
2018-01-02 00:00:00+00:00                       285.0
2018-01-03 00:00:00+00:00                       285.0
2018-01-04 00:00:00+00:00                       284.0
2018-01-05 00:00:00+00:00                       284.0

                           USGS:07346000:00060:00002  \
datetimeUTC
2018-01-01 00:00:00+00:00                       283.0
2018-01-02 00:00:00+00:00                       284.0
2018-01-03 00:00:00+00:00                       284.0
2018-01-04 00:00:00+00:00                       283.0
2018-01-05 00:00:00+00:00                       282.0

                           USGS:07346000:00060:00003
datetimeUTC
2018-01-01 00:00:00+00:00                       284.0
2018-01-02 00:00:00+00:00                       285.0
2018-01-03 00:00:00+00:00                       284.0
2018-01-04 00:00:00+00:00                       283.0
2018-01-05 00:00:00+00:00                       283.0
```

(continues on next page)

(continued from previous page)

```
[5 rows x 73 columns]
```

Download all streamflow data for the state of Virginia

[18]:
```
# Use this carefully! You can easily request more data than you will know what to do␣
↪with.
start = "2017-01-01"
end = "2017-12-31"
param = '00060'
virginia = hf.NWIS(None, "dv", start, end, parameterCd=param, stateCd='va', file=
↪'virginia.parquet')
```

```
Reading data from virginia.parquet
```

[19]:
```
# Calculate the mean for each site.
virginia.df('discharge').mean()
```

[19]:
```
USGS:01613900:00060:00003      12.901726
USGS:01615000:00060:00003      41.219260
USGS:01616100:00060:00003       8.948493
USGS:01620500:00060:00003      17.857973
USGS:01621050:00060:00003       3.844548
USGS:01622000:00060:00003     277.692055
USGS:01625000:00060:00003     245.340548
USGS:01626000:00060:00003     106.207123
USGS:01626850:00060:00003     143.700000
USGS:01627500:00060:00003     184.487397
USGS:01628500:00060:00003     748.202740
USGS:01629500:00060:00003     990.643836
USGS:01631000:00060:00003    1114.293151
USGS:01632000:00060:00003     124.048740
USGS:01632082:00060:00003      17.520329
USGS:01632900:00060:00003      39.136493
USGS:01633000:00060:00003     218.096712
USGS:01634000:00060:00003     361.363562
USGS:01634500:00060:00003      77.218795
USGS:01635500:00060:00003      48.196192
USGS:01636316:00060:00003      16.360575
USGS:01636690:00060:00003      11.838795
USGS:01638350:00060:00003      25.122712
USGS:01638420:00060:00003      19.026932
USGS:01638480:00060:00003      66.770411
USGS:01643590:00060:00003       5.074493
USGS:01643700:00060:00003      98.993425
USGS:01643805:00060:00003      31.099041
USGS:01643880:00060:00003      42.124575
USGS:01644000:00060:00003     271.511233
                                 ...
USGS:02076000:00060:00003    2715.035616
USGS:02077000:00060:00003     495.344384
USGS:02079640:00060:00003      30.671205
```

(continues on next page)

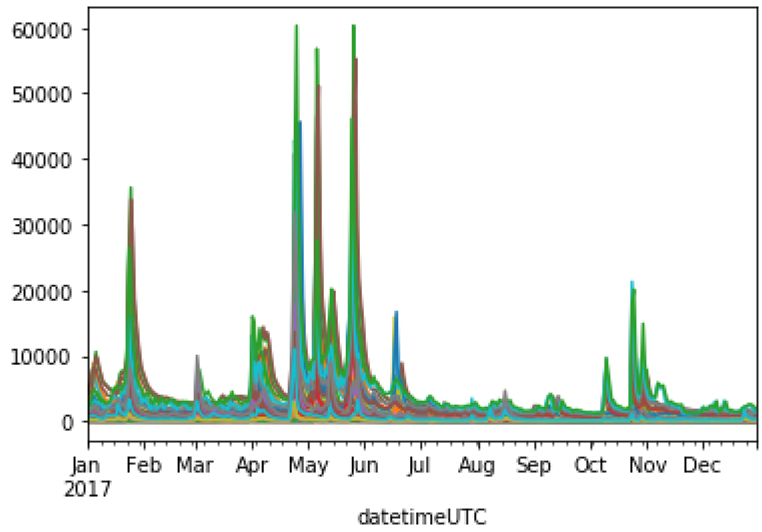
(continued from previous page)

```
USGS:03164000:00060:00003    1809.479452
USGS:03165000:00060:00003      69.004110
USGS:03165500:00060:00003    2113.663014
USGS:03167000:00060:00003     268.287671
USGS:03168000:00060:00003    3159.178082
USGS:03170000:00060:00003     366.520548
USGS:03171000:00060:00003    3966.383562
USGS:03173000:00060:00003     335.324932
USGS:03175500:00060:00003     321.495616
USGS:03176500:00060:00003    5021.671233
USGS:03177710:00060:00003      62.864110
USGS:03207800:00060:00003     385.552877
USGS:03208500:00060:00003     330.526575
USGS:03208950:00060:00003      85.020000
USGS:03209000:00060:00003     294.153973
USGS:03471500:00060:00003     109.356712
USGS:03473000:00060:00003     501.227123
USGS:03474000:00060:00003     171.507123
USGS:03475000:00060:00003     254.800822
USGS:03478400:00060:00003      34.246301
USGS:03488000:00060:00003     296.555890
USGS:03490000:00060:00003     923.371507
USGS:03524000:00060:00003     700.853151
USGS:03524500:00060:00003     143.473973
USGS:03527000:00060:00003    1510.515068
USGS:03529500:00060:00003     187.819178
USGS:03531500:00060:00003     554.865479
Length: 190, dtype: float64
```

Plot all streamflow data for the state of Virginia

[20]:
```
# There are so many sites that we can't read them all!
virginia.df('q').plot(legend=None)
```

[20]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x22f67bb6588>
```



Download all streamflow data for Fairfax and Prince William counties in the state of Virginia

[21]:
```
start = "2017-01-01"
end = "2017-12-31"
county = hf.NWIS(None, "dv", start, end, parameterCd='00060', countyCd=['51059', '51061
↪'], file='PG.parquet')
```

```
Reading data from PG.parquet
```

[22]:
```
county.df('data').head()
```

[22]:
```
                           USGS:01645704:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                       1.89
2017-01-02 00:00:00+00:00                       4.69
2017-01-03 00:00:00+00:00                      33.60
2017-01-04 00:00:00+00:00                      13.40
2017-01-05 00:00:00+00:00                       6.06

                           USGS:01645762:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                       0.92
2017-01-02 00:00:00+00:00                       1.44
2017-01-03 00:00:00+00:00                       9.79
2017-01-04 00:00:00+00:00                       2.87
2017-01-05 00:00:00+00:00                       1.74

                           USGS:01646000:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                       19.0
2017-01-02 00:00:00+00:00                       28.8
2017-01-03 00:00:00+00:00                      298.0
2017-01-04 00:00:00+00:00                       95.6
```

(continues on next page)

(continued from previous page)

```
2017-01-05 00:00:00+00:00                    41.4

                          USGS:01646305:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                    0.40
2017-01-02 00:00:00+00:00                    2.17
2017-01-03 00:00:00+00:00                   20.10
2017-01-04 00:00:00+00:00                    1.26
2017-01-05 00:00:00+00:00                    0.73

                          USGS:01654000:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                    3.88
2017-01-02 00:00:00+00:00                   18.30
2017-01-03 00:00:00+00:00                  225.00
2017-01-04 00:00:00+00:00                   24.10
2017-01-05 00:00:00+00:00                    9.77

                          USGS:01654500:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                    0.49
2017-01-02 00:00:00+00:00                    2.02
2017-01-03 00:00:00+00:00                   23.10
2017-01-04 00:00:00+00:00                    2.17
2017-01-05 00:00:00+00:00                    1.14

                          USGS:01655794:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                     NaN
2017-01-02 00:00:00+00:00                     NaN
2017-01-03 00:00:00+00:00                     NaN
2017-01-04 00:00:00+00:00                     NaN
2017-01-05 00:00:00+00:00                     NaN

                          USGS:01656000:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                    9.55
2017-01-02 00:00:00+00:00                    9.70
2017-01-03 00:00:00+00:00                  128.00
2017-01-04 00:00:00+00:00                   90.50
2017-01-05 00:00:00+00:00                   36.50

                          USGS:01656903:00060:00003  \
datetimeUTC
2017-01-01 00:00:00+00:00                    1.12
2017-01-02 00:00:00+00:00                    2.18
2017-01-03 00:00:00+00:00                   23.50
2017-01-04 00:00:00+00:00                    7.46
2017-01-05 00:00:00+00:00                    3.20

                          USGS:01661977:00060:00003  \
datetimeUTC
```

(continues on next page)

(continued from previous page)

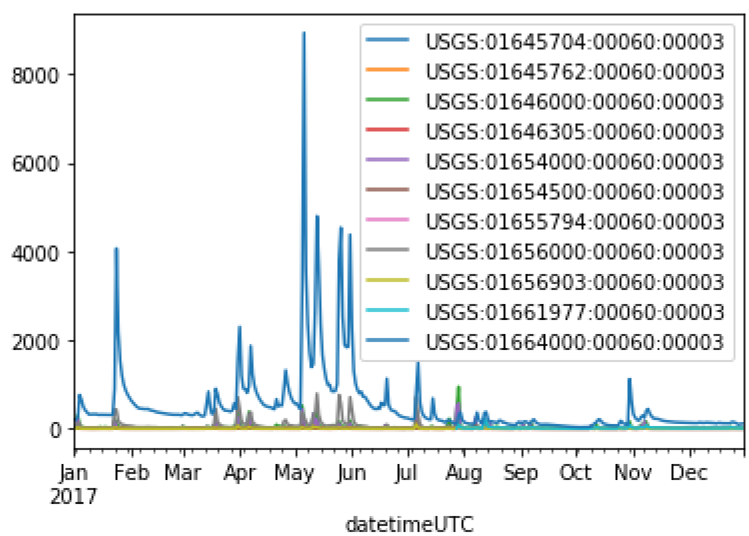
```
2017-01-01 00:00:00+00:00                              NaN
2017-01-02 00:00:00+00:00                              NaN
2017-01-03 00:00:00+00:00                              NaN
2017-01-04 00:00:00+00:00                              NaN
2017-01-05 00:00:00+00:00                              NaN

                           USGS:01664000:00060:00003
datetimeUTC
2017-01-01 00:00:00+00:00                            205.0
2017-01-02 00:00:00+00:00                            200.0
2017-01-03 00:00:00+00:00                            256.0
2017-01-04 00:00:00+00:00                            761.0
2017-01-05 00:00:00+00:00                            650.0
```

Plot all streamflow data for Fairfax and Prince William counties in the state of Virginia

[23]:
```
county.df('data').plot()
```

[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x22f67bd87b8>
```



16.8.7 Thanks for using hydrofunctions!

We would love to hear your comments and suggestions!

CHAPTER

SEVENTEEN

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

h

INDEX

C

D

E

F

G

H

M

N

P

R

S