An introduction to Python Programming for Research

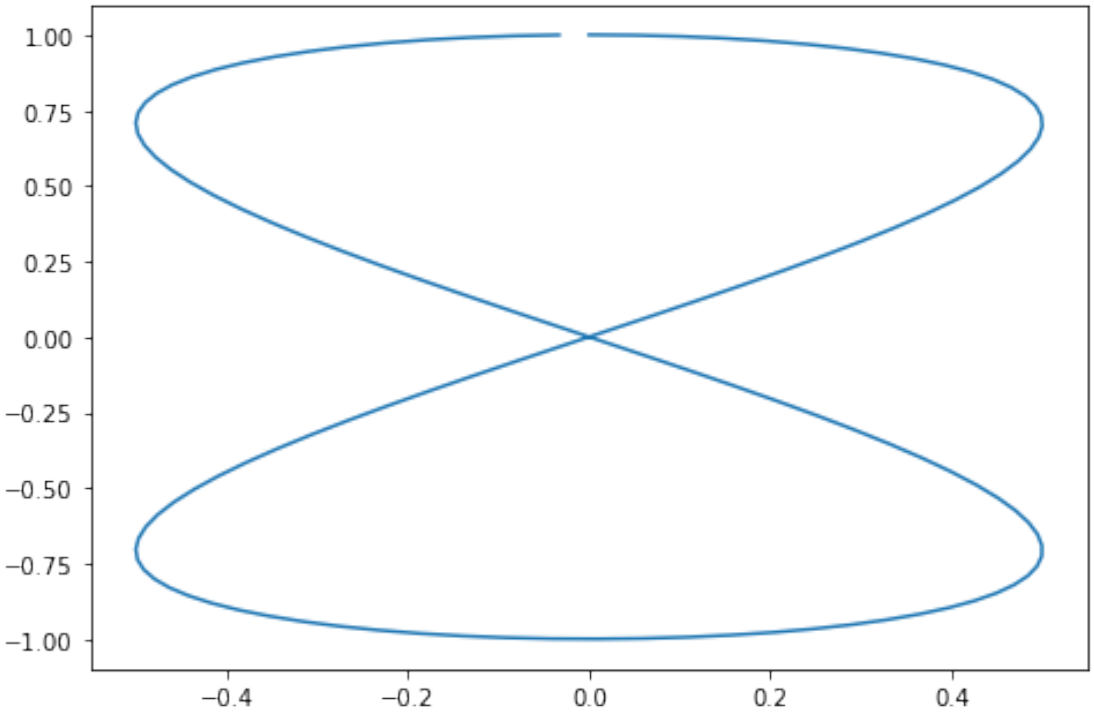
James Hetherington

November 4, 2019

Contents

In [1]:
```
### Make plot
%matplotlib inline
import math

import numpy as np
import matplotlib.pyplot as plt

theta = np.arange(0, 4 * math.pi, 0.1)
eight = plt.figure()
axes = eight.add_axes([0, 0, 1, 1])
axes.plot(0.5 * np.sin(theta), np.cos(theta / 2))
```

Out[1]: [<matplotlib.lines.Line2D at 0x114cd3cf8>]



In [2]:
```
print("This cell is a code cell")
```

```
This cell is a code cell
```

In [3]:
```
%%bash
# Above line tells Python to execute this cell as *shell code*
# not Python, as if we were in a command line
# This is called a 'cell magic'

python -c "print(2 * 4)"
```

```
8
```

In [4]:
```
%%bash
echo "print(2 * 4)" > eight.py
python eight.py
```

```
8
```

In [5]:
```
%%writefile fourteen.py
#! /usr/bin/env python
print(2 * 7)
```

```
Overwriting fourteen.py
```

In [6]:
```
%%bash
chmod u+x fourteen.py
./fourteen.py
```

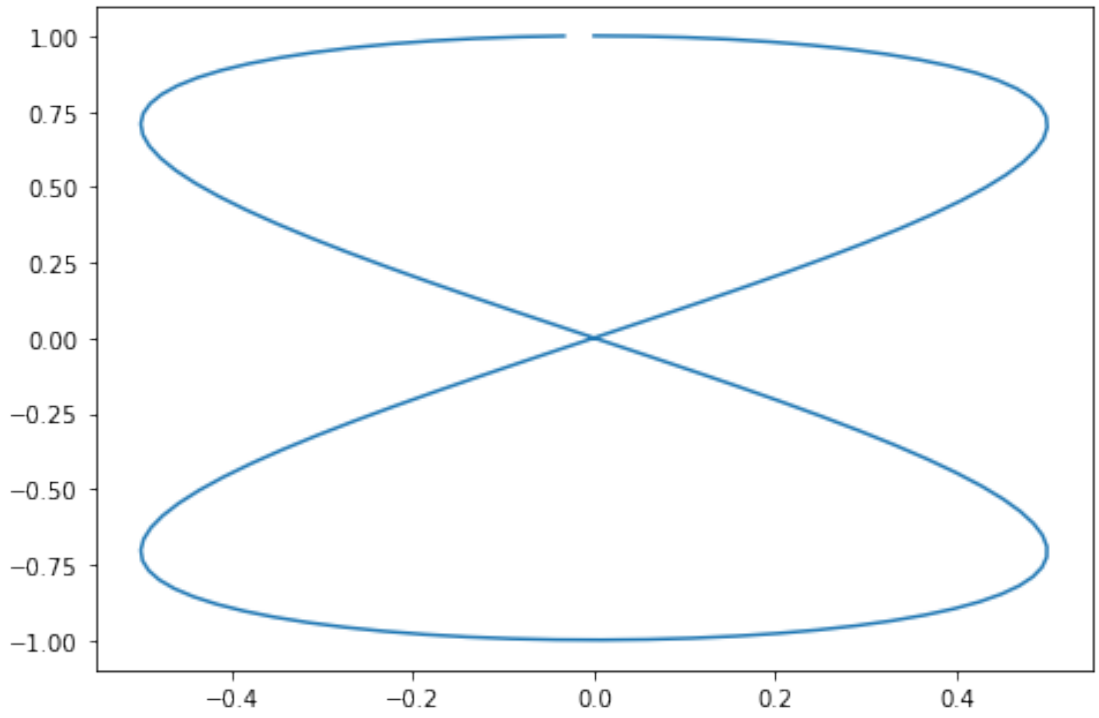
```
14
```

In [7]:
```
%%writefile draw_eight.py
# Above line tells the notebook to treat the rest of this
# cell as content for a file on disk.
import math

import numpy as np
import matplotlib.pyplot as plt

def make_figure():
    theta = np.arange(0, 4 * math.pi, 0.1)
    eight = plt.figure()
    axes = eight.add_axes([0, 0, 1, 1])
    axes.plot(0.5 * np.sin(theta), np.cos(theta / 2))
    return eight
```

```
Overwriting draw_eight.py
```

In [8]:
```
import draw_eight # Load the library file we just wrote to disk
```

In [9]:
```
image = draw_eight.make_figure()
```




```
In [1]: import geopy # A python library for investigating geographic information.
        # https://pypi.org/project/geopy/

In [2]: geocoder = geopy.geocoders.Nominatim(user_agent="my-application")
        geocoder.geocode('Cambridge', exactly_one=False)
```

```
---------------------------------------------------------------------------

timeout                                   Traceback (most recent call last)

/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.p
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.p
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
```

```
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-2-dc34efe900db> in <module>
      1 geocoder = geopy.geocoders.Nominatim(user_agent="my-application")
----> 2 geocoder.geocode('Cambridge', exactly_one=False)


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_on
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389
```

```
/usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
376             elif isinstance(error, URLError):
377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
379                 elif "unreachable" in message:
380                     raise GeocoderUnavailable('Service not available')


GeocoderTimedOut: Service timed out
```

```
In [3]: print("This runs") # print "This doesn't"
        # print This doesn't either
```

```
This runs
```

```
In [4]: def geolocate(place):
          return geocoder.geocode(place, exactly_one = False)[0][1]
```

```
In [5]: geolocate('Cambridge')
```

```
---------------------------------------------------------------------------

timeout                                   Traceback (most recent call last)

/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
1318             except OSError as err: # timeout error


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
1230


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
1276


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
1225


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
1015         del self._buffer[:]
-> 1016         self.send(msg)
1017
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    542         result = self._call_chain(self.handle_open, protocol, protocol +
```

```
--> 543                                     '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-5-ccb6d38c8bab> in <module>
----> 1 geolocate('Cambridge')


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')
```

```
GeocoderTimedOut: Service timed out
```

```
In [6]: london_location = geolocate("London")
        print(london_location)
```

```
---------------------------------------------------------------------------

timeout                                   Traceback (most recent call last)

/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
```

```
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.p
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.p
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-6-e33090ca51bc> in <module>
----> 1 london_location = geolocate("London")
      2 print(london_location)


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out
```

In [7]:
```
import requests
def request_map_at(lat, long, satellite=True,
                   zoom=12, size=(400, 400)):
    base = "https://static-maps.yandex.ru/1.x/?"

    params = dict(
        z = zoom,
        size = str(size[0]) + "," + str(size[1]),
        ll = str(long) + "," + str(lat),
        l = "sat" if satellite else "map",
```

```
            lang = "en_US"
        )

        return requests.get(base,params=params)
```

```
In [8]: map_response = request_map_at(51.5072, -0.1275)
```

```
In [9]: url = map_response.url
        print(url[0:50])
        print(url[50:100])
        print(url[100:])
```

```
https://static-maps.yandex.ru/1.x/?z=12&size=400%2
C400&ll=-0.1275%2C51.5072&l=sat&lang=en_US
```

```
In [10]: from nose.tools import assert_in

         assert_in("https://static-maps.yandex.ru/1.x/?", url)
         assert_in("ll=-0.1275%2C51.5072", url)
         assert_in("z=12", url)
         assert_in("size=400%2C400", url)
```

```
        ---------------------------------------------------------------------------

        ModuleNotFoundError                       Traceback (most recent call last)

        <ipython-input-10-8ded406521fd> in <module>
    ----> 1 from nose.tools import assert_in
          2
          3 assert_in("https://static-maps.yandex.ru/1.x/?", url)
          4 assert_in("ll=-0.1275%2C51.5072", url)
          5 assert_in("z=12", url)


        ModuleNotFoundError: No module named 'nose'
```

```
In [11]: map_response.content[0:20]
```

```
Out[11]: b'\xff\xd8\xff\xe0\x00\x10JFIF\x00\x01\x01\x01\x00H\x00H\x00\x00'
```

```
In [12]: def map_at(*args, **kwargs):
             return request_map_at(*args, **kwargs).content
```

```
In [13]: import IPython
         map_png = map_at(*london_location)
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-13-d69fb2ebab72> in <module>
```

```
      1 import IPython
----> 2 map_png = map_at(*london_location)


NameError: name 'london_location' is not defined
```

```
In [14]: print("The type of our map result is actually a: ", type(map_png))
```

```
---------------------------------------------------------------------------

NameError                                 Traceback (most recent call last)

<ipython-input-14-6c68dc8fc8d1> in <module>
----> 1 print("The type of our map result is actually a: ", type(map_png))


NameError: name 'map_png' is not defined
```

```
In [15]: IPython.core.display.Image(map_png)
```

```
---------------------------------------------------------------------------

NameError                                 Traceback (most recent call last)

<ipython-input-15-ac666969e449> in <module>
----> 1 IPython.core.display.Image(map_png)


NameError: name 'map_png' is not defined
```

```
In [16]: IPython.core.display.Image(map_at(*geolocate("New Delhi")))
```

```
---------------------------------------------------------------------------

timeout                                   Traceback (most recent call last)

/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
```

```
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle

    timeout: timed out

During handling of the above exception, another exception occurred:

    URLError                                  Traceback (most recent call last)
```

```
    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-16-9124101779f1> in <module>
----> 1 IPython.core.display.Image(map_at(*geolocate("New Delhi")))


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]
```

```
/usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
385
386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
388         )
389


/usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
376             elif isinstance(error, URLError):
377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
379                 elif "unreachable" in message:
380                     raise GeocoderUnavailable('Service not available')


GeocoderTimedOut: Service timed out
```

```
In [17]: from io import BytesIO # A library to convert between files and strings
         import numpy as np # A library to deal with matrices
         import imageio # A library to deal with images
```

```
---------------------------------------------------------------------------

ModuleNotFoundError                       Traceback (most recent call last)

<ipython-input-17-31fe9dfd527f> in <module>
  1 from io import BytesIO # A library to convert between files and strings
  2 import numpy as np # A library to deal with matrices
----> 3 import imageio # A library to deal with images


ModuleNotFoundError: No module named 'imageio'
```

```
In [18]: def is_green(pixels):
             threshold = 1.1
             greener_than_red = pixels[:,:,1] > threshold * pixels[:,:,0]
             greener_than_blue = pixels[:,:,1] > threshold * pixels[:,:,2]
             green = np.logical_and(greener_than_red, greener_than_blue)
             return green

In [19]: def count_green_in_png(data):
             f = BytesIO(data)
             pixels = imageio.imread(f) # Get our PNG image as a numpy array
             return np.sum(is_green(pixels))

In [20]: print(count_green_in_png( map_at(*london_location) ))
```

```
---------------------------------------------------------------------------

NameError                                 Traceback (most recent call last)
```

```
    <ipython-input-20-1df2d88d5544> in <module>
----> 1 print(count_green_in_png( map_at(*london_location) ))


    NameError: name 'london_location' is not defined
```

```
In [21]: def location_sequence(start, end, steps):
             lats = np.linspace(start[0], end[0], steps) # "Linearly spaced" data
             longs = np.linspace(start[1], end[1], steps)
             return np.vstack([lats, longs]).transpose()
```

```
In [22]: location_sequence(geolocate("London"), geolocate("Cambridge"), 5)
```

```
    ---------------------------------------------------------------------------

    timeout                                   Traceback (most recent call last)

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
```

```
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-22-ed53afe2376e> in <module>
----> 1 location_sequence(geolocate("London"), geolocate("Cambridge"), 5)


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_on
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out
```

```
In [23]: def show_green_in_png(data):
             pixels = imageio.imread(BytesIO(data)) # Get our PNG image as rows of pixels
             green = is_green(pixels)
```

```
            out = green[:, :, np.newaxis] * np.array([0, 1, 0])[np.newaxis, np.newaxis, :]


            buffer = BytesIO()
            result = imageio.imwrite(buffer, out, format='png')
            return buffer.getvalue()
```

```
In [24]: IPython.core.display.Image(
             map_at(*london_location, satellite=True)
         )
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-24-84d560d5795b> in <module>
          1 IPython.core.display.Image(
        ----> 2     map_at(*london_location, satellite=True)
          3 )


        NameError: name 'london_location' is not defined
```

```
In [25]: IPython.core.display.Image(
             show_green_in_png(
                 map_at(
                     *london_location,
                     satellite=True)))
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-25-ba1938f843d6> in <module>
          2     show_green_in_png(
          3         map_at(
        ----> 4             *london_location,
          5             satellite=True)))


        NameError: name 'london_location' is not defined
```

```
In [26]: for location in location_sequence(geolocate("London"),
                                           geolocate("Birmingham"),
                                           4):
             IPython.core.display.display(
                 IPython.core.display.Image(map_at(*location)))
```

```
        ---------------------------------------------------------------------------
```

```
timeout                                   Traceback (most recent call last)

/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
1316                h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                          encode_chunked=req.has_header('Transfer-encoding'))
1318            except OSError as err: # timeout error


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
1230


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
1276


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
1225


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
1015         del self._buffer[:]
-> 1016         self.send(msg)
1017


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
955             if self.auto_open:
--> 956                 self.connect()
957             else:


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
1383
-> 1384             super().connect()
1385


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


/usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
726     if err is not None:
--> 727         raise err
```

```
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.p
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>
```

```
During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-26-b3877d0d28cf> in <module>
----> 1 for location in location_sequence(geolocate("London"),
      2                                   geolocate("Birmingham"),
      3                                   4):
      4     IPython.core.display.display(
      5         IPython.core.display.Image(map_at(*location)))


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out

In [27]: [count_green_in_png(map_at(*location))
                  for location in
                       location_sequence(geolocate("London"),
                                         geolocate("Birmingham"),
                                         10)]


    ---------------------------------------------------------------------------

    timeout                                   Traceback (most recent call last)

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle
```

```
    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-27-b5d8a75e50ec> in <module>
```

```
      1 [count_green_in_png(map_at(*location))
      2             for location in
----> 3                  location_sequence(geolocate("London"),
      4                                    geolocate("Birmingham"),
      5                                    10)]


    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out
```

```
In [28]: import matplotlib.pyplot as plt
         %matplotlib inline

In [29]: plt.plot([count_green_in_png(map_at(*location))
                   for location in
                     location_sequence(geolocate("London"),
                                       geolocate("Birmingham"),
                                       10)])
```

```
    ---------------------------------------------------------------------------

    timeout                                   Traceback (most recent call last)

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230
```

```
    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
   1276


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:
```

```
    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, time
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-29-e7d26b5362c3> in <module>
      1 plt.plot([count_green_in_png(map_at(*location))
      2             for location in
----> 3               location_sequence(geolocate("London"),
      4                                 geolocate("Birmingham"),
      5                                 10)])
```

```
    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out
```

```
In [30]: def green_between(start, end,steps):
             return [count_green_in_png( map_at(*location) )
                     for location in location_sequence(
                                     geolocate(start),
                                     geolocate(end),
                                     steps)]
```

```
In [31]: plt.plot(green_between('New York', 'Chicago', 20))
```

```
    ---------------------------------------------------------------------------

    timeout                                   Traceback (most recent call last)

    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/r
   1316                 h.request(req.get_method(), req.selector, req.data, headers,
-> 1317                           encode_chunked=req.has_header('Transfer-encoding'))
   1318             except OSError as err: # timeout error


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1228         """Send a complete request to the server."""
-> 1229         self._send_request(method, url, body, headers, encode_chunked)
   1230


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cli
   1274             body = _encode(body, 'body')
-> 1275         self.endheaders(body, encode_chunked=encode_chunked)
```

```
   1276


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1223             raise CannotSendHeader()
-> 1224         self._send_output(message_body, encode_chunked=encode_chunked)
   1225


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1015         del self._buffer[:]
-> 1016         self.send(msg)
   1017


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    955             if self.auto_open:
--> 956                 self.connect()
    957             else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
   1383
-> 1384             super().connect()
   1385


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/http/cl
    927         self.sock = self._create_connection(
--> 928             (self.host,self.port), self.timeout, self.source_address)
    929         self.sock.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    726     if err is not None:
--> 727         raise err
    728     else:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/socket.
    715                 sock.bind(source_address)
--> 716             sock.connect(sa)
    717             # Break explicitly a reference cycle


    timeout: timed out


During handling of the above exception, another exception occurred:


    URLError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
```

```
    354         try:
--> 355             page = requester(req, timeout=timeout, **kwargs)
    356         except Exception as error:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    524
--> 525         response = self._open(req, data)
    526


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    542         result = self._call_chain(self.handle_open, protocol, protocol +
--> 543                                   '_open', req)
    544         if result:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
    502             func = getattr(handler, meth_name)
--> 503             result = func(*args)
    504             if result is not None:


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1359             return self.do_open(http.client.HTTPSConnection, req,
-> 1360                 context=self._context, check_hostname=self._check_hostname)
   1361


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/urllib/
   1318             except OSError as err: # timeout error
-> 1319                 raise URLError(err)
   1320             r = h.getresponse()


    URLError: <urlopen error timed out>


During handling of the above exception, another exception occurred:


    GeocoderTimedOut                          Traceback (most recent call last)

    <ipython-input-31-eea54607c6ae> in <module>
----> 1 plt.plot(green_between('New York', 'Chicago', 20))


    <ipython-input-30-7a2c39ca1229> in green_between(start, end, steps)
      2     return [count_green_in_png( map_at(*location) )
      3             for location in location_sequence(
----> 4                             geolocate(start),
      5                             geolocate(end),
      6                             steps)]
```

```
    <ipython-input-4-2394ca7f2ca5> in geolocate(place)
      1 def geolocate(place):
----> 2   return geocoder.geocode(place, exactly_one = False)[0][1]


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/osm.py in geocode(self, query, exactly_o
    385
    386         return self._parse_json(
--> 387             self._call_geocoder(url, timeout=timeout), exactly_one
    388         )
    389


    /usr/local/lib/python3.7/site-packages/geopy/geocoders/base.py in _call_geocoder(self, url, tim
    376             elif isinstance(error, URLError):
    377                 if "timed out" in message:
--> 378                     raise GeocoderTimedOut('Service timed out')
    379                 elif "unreachable" in message:
    380                     raise GeocoderUnavailable('Service not available')


    GeocoderTimedOut: Service timed out
```

```
In [1]: 2*3

Out[1]: 6

In [2]: six = 2*3

In [3]: print(six)

6


In [4]: print(seven)


    ---------------------------------------------------------------------------

    NameError                                 Traceback (most recent call last)

    <ipython-input-4-25c0309421cb> in <module>
----> 1 print(seven)


    NameError: name 'seven' is not defined


In [5]: nothing = None

In [6]: print(nothing)

None
```

```
In [7]: type(None)
Out[7]: NoneType
In [8]: print(5*six)
30

In [9]: scary = six*six*six
In [10]: print(scary)
216

In [11]: scary = 25
In [12]: print(scary)
25

In [13]: name = "James"
In [14]: nom = name
In [15]: print(nom)
James

In [16]: print(name)
James

In [17]: nom = "Hetherington"
In [18]: print(name)
James

In [19]: print(nom)
Hetherington

In [20]: name = "Jim"
In [21]: type(name)
Out[21]: str
In [22]: z = 3+1j
In [23]: dir(z)
```

```
Out[23]: ['__abs__',
          '__add__',
          '__bool__',
          '__class__',
          '__delattr__',
          '__dir__',
          '__divmod__',
          '__doc__',
          '__eq__',
          '__float__',
          '__floordiv__',
          '__format__',
          '__ge__',
          '__getattribute__',
          '__getnewargs__',
          '__gt__',
          '__hash__',
          '__init__',
          '__init_subclass__',
          '__int__',
          '__le__',
          '__lt__',
          '__mod__',
          '__mul__',
          '__ne__',
          '__neg__',
          '__new__',
          '__pos__',
          '__pow__',
          '__radd__',
          '__rdivmod__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__rfloordiv__',
          '__rmod__',
          '__rmul__',
          '__rpow__',
          '__rsub__',
          '__rtruediv__',
          '__setattr__',
          '__sizeof__',
          '__str__',
          '__sub__',
          '__subclasshook__',
          '__truediv__',
          'conjugate',
          'imag',
          'real']

In [24]: type(z)

Out[24]: complex

In [25]: z.real
```

```
Out[25]: 3.0

In [26]: z.imag

Out[26]: 1.0

In [27]: z.wrong

         ---------------------------------------------------------------------------

         AttributeError                            Traceback (most recent call last)

         <ipython-input-27-0cc5a8ef8f99> in <module>
     ----> 1 z.wrong


         AttributeError: 'complex' object has no attribute 'wrong'


In [28]: z2 = 5-6j
          print("Gets to here")
          print(z.wrong)
          print("Didn't get to here")

Gets to here



         ---------------------------------------------------------------------------

         AttributeError                            Traceback (most recent call last)

         <ipython-input-28-88a4fd40cc7a> in <module>
           1 z2 = 5-6j
           2 print("Gets to here")
     ----> 3 print(z.wrong)
           4 print("Didn't get to here")


         AttributeError: 'complex' object has no attribute 'wrong'


In [29]: number = 0

In [30]: print(number)

0


In [31]: number = number + 1

In [32]: print(number)

1
```

```
In [1]: len("pneumonoultramicroscopicsilicovolcanoconiosis")

Out[1]: 45

In [2]: sorted("Python")

Out[2]: ['P', 'h', 'n', 'o', 't', 'y']

In [3]: len('Jim')*8

Out[3]: 24

In [4]: x = len('Mike')
        y = len('Bob')
        z = x+y

In [5]: print(z)

7


In [6]: "shout".upper()

Out[6]: 'SHOUT'

In [7]: x = 5

In [8]: type(x)

Out[8]: int

In [9]: x.upper()

        ---------------------------------------------------------------------------

        AttributeError                            Traceback (most recent call last)

        <ipython-input-9-328ac508ff1b> in <module>
    ----> 1 x.upper()


        AttributeError: 'int' object has no attribute 'upper'


In [10]: x.wrong


        ---------------------------------------------------------------------------

        AttributeError                            Traceback (most recent call last)

        <ipython-input-10-29321da545fa> in <module>
    ----> 1 x.wrong


        AttributeError: 'int' object has no attribute 'wrong'
```

```
In [11]: z = 1+5j

In [12]: z.real

Out[12]: 1.0

In [13]: z.conjugate()

Out[13]: (1-5j)

In [14]: z.conjugate

Out[14]: <function complex.conjugate>

In [15]: z.conjugate

Out[15]: <function complex.conjugate>

In [16]: type(z.conjugate)

Out[16]: builtin_function_or_method

In [17]: somefunc=z.conjugate

In [18]: somefunc()

Out[18]: (1-5j)

In [19]: sorted([1,5,3,4])

Out[19]: [1, 3, 4, 5]

In [20]: magic = sorted

In [21]: type(magic)

Out[21]: builtin_function_or_method

In [22]: magic(["Technology", "Advanced"])

Out[22]: ['Advanced', 'Technology']

In [23]: help(sorted)

Help on built-in function sorted in module builtins:

sorted(iterable, /, *, key=None, reverse=False)
    Return a new list containing all items from the iterable in ascending order.

    A custom key function can be supplied to customize the sort order, and the
    reverse flag can be set to request the result in descending order.


In [24]: dir("Hexxo")
```

```
Out[24]: ['__add__',
          '__class__',
          '__contains__',
          '__delattr__',
          '__dir__',
          '__doc__',
          '__eq__',
          '__format__',
          '__ge__',
          '__getattribute__',
          '__getitem__',
          '__getnewargs__',
          '__gt__',
          '__hash__',
          '__init__',
          '__init_subclass__',
          '__iter__',
          '__le__',
          '__len__',
          '__lt__',
          '__mod__',
          '__mul__',
          '__ne__',
          '__new__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__rmod__',
          '__rmul__',
          '__setattr__',
          '__sizeof__',
          '__str__',
          '__subclasshook__',
          'capitalize',
          'casefold',
          'center',
          'count',
          'encode',
          'endswith',
          'expandtabs',
          'find',
          'format',
          'format_map',
          'index',
          'isalnum',
          'isalpha',
          'isascii',
          'isdecimal',
          'isdigit',
          'isidentifier',
          'islower',
          'isnumeric',
          'isprintable',
          'isspace',
```

```
          'istitle',
          'isupper',
          'join',
          'ljust',
          'lower',
          'lstrip',
          'maketrans',
          'partition',
          'replace',
          'rfind',
          'rindex',
          'rjust',
          'rpartition',
          'rsplit',
          'rstrip',
          'split',
          'splitlines',
          'startswith',
          'strip',
          'swapcase',
          'title',
          'translate',
          'upper',
          'zfill']
```

```
In [25]: "Hexxo".replace("x", "l")
```

```
Out[25]: 'Hello'
```

```
In [26]: help("FIsh".replace)
```

```
Help on built-in function replace:

replace(old, new, count=-1, /) method of builtins.str instance
    Return a copy with all occurrences of substring old replaced by new.

      count
        Maximum number of occurrences to replace.
        -1 (the default value) means replace all occurrences.

    If the optional argument count is given, only the first count occurrences are
    replaced.
```

```
In [27]: x = 2 + 3
```

```
In [28]: print(x)
```

```
5
```

```
In [29]: x.__add__(7)
```

```
Out[29]: 12
```

```
In [30]: "Hello" + "Goodbye"
```

```
Out[30]: 'HelloGoodbye'

In [31]: [2, 3, 4] + [5, 6]

Out[31]: [2, 3, 4, 5, 6]

In [32]: 7-2

Out[32]: 5

In [33]: [2, 3, 4] - [5, 6]

         ---------------------------------------------------------------------------

         TypeError                                 Traceback (most recent call last)

         <ipython-input-33-5b64b789ad11> in <module>
     ----> 1 [2, 3, 4] - [5, 6]


         TypeError: unsupported operand type(s) for -: 'list' and 'list'

In [34]: [2, 3, 4] + 5

         ---------------------------------------------------------------------------

         TypeError                                 Traceback (most recent call last)

         <ipython-input-34-67b01a5c24ab> in <module>
     ----> 1 [2, 3, 4] + 5


         TypeError: can only concatenate list (not "int") to list

In [35]: [2, 3, 4] + [5]

Out[35]: [2, 3, 4, 5]

In [36]: print(2+3*4)

14

In [37]: print((2+3)*4)

20


In [1]: type(5)

Out[1]: int

In [2]: one = 1
        ten = 10
        one_float = 1.0
        ten_float = 10.
```

```
In [3]: tenth= one_float/ten_float

In [4]: tenth

Out[4]: 0.1

In [5]: type(one)

Out[5]: int

In [6]: type(one_float)

Out[6]: float

In [7]: print(one//ten)

0


In [8]: one_float/ten_float

Out[8]: 0.1

In [9]: print(type(one/ten))

<class 'float'>


In [10]: type(tenth)

Out[10]: float

In [11]: 10//3

Out[11]: 3

In [12]: 10.0/3

Out[12]: 3.3333333333333335

In [13]: 10/3.0

Out[13]: 3.3333333333333335

In [14]: x = float(5)
         type(x)

Out[14]: float

In [15]: 10/float(3)

Out[15]: 3.3333333333333335

In [16]: N = 10000.0
         sum([1/N]*int(N))

Out[16]: 0.9999999999999062

In [17]: given = "James"
         family = "Hetherngton"
         full = given + " " + family
```

```
In [18]: print(full.upper())

JAMES HETHERNGTON

In [19]: ten, one

Out[19]: (10, 1)

In [20]: print(ten + one)

11

In [21]: print(float(str(ten) + str(one)))

101.0

In [22]: "    Hello  ".strip()

Out[22]: 'Hello'

In [23]: "James's Class"

Out[23]: "James's Class"

In [24]: '"Wow!", said Bob.'

Out[24]: '"Wow!", said Bob.'

In [25]: [1, 3, 7]

Out[25]: [1, 3, 7]

In [26]: type([1, 3, 7])

Out[26]: list

In [27]: various_things = [1, 2, "banana", 3.4, [1,2] ]

In [28]: various_things[2]

Out[28]: 'banana'

In [29]: index = 0
         various_things[index]

Out[29]: 1

In [30]: name = ["James", "Philip", "John", "Hetherington"]
         print("==".join(name))

James==Philip==John==Hetherington

In [31]: "Ernst Stavro Blofeld".split(" ")

Out[31]: ['Ernst', 'Stavro', 'Blofeld']
```

```
In [32]: "Ernst Stavro Blofeld".split("o")

Out[32]: ['Ernst Stavr', ' Bl', 'feld']

In [33]: "->".join("John Ronald Reuel Tolkein".split(" "))

Out[33]: 'John->Ronald->Reuel->Tolkein'

In [34]: identity = [[1, 0], [0, 1]]

In [35]: identity[0][0]

Out[35]: 1

In [36]: range(5)

Out[36]: range(0, 5)

In [37]: count_to_five = range(5)
         print(list(count_to_five))

[0, 1, 2, 3, 4]


In [38]: print(count_to_five[1])

1


In [39]: print("James"[2])

m


In [40]: count_to_five = range(5)

In [41]: count_to_five[1:3]

Out[41]: range(1, 3)

In [42]: "Hello World"[4:8]

Out[42]: 'o Wo'

In [43]: len(various_things)

Out[43]: 5

In [44]: len("Python")

Out[44]: 6

In [45]: name

Out[45]: ['James', 'Philip', 'John', 'Hetherington']

In [46]: "John" in name

Out[46]: True

In [47]: 3 in count_to_five
```

```
Out[47]: True

In [48]: mylist = ['Hello', 'World']
         a, b = mylist
         print(b)

World


In [49]: range(4)

Out[49]: range(0, 4)

In [50]: zero, one, two, three = range(4)

In [51]: two

Out[51]: 2

In [52]: zero, one, two, three = range(7)

         ---------------------------------------------------------------------------

         ValueError                                Traceback (most recent call last)

         <ipython-input-52-3331a3ab5222> in <module>
     ----> 1 zero, one, two, three = range(7)


         ValueError: too many values to unpack (expected 4)


In [53]: zero, one, two, three = range(2)


         ---------------------------------------------------------------------------

         ValueError                                Traceback (most recent call last)

         <ipython-input-53-8575e9410b1d> in <module>
     ----> 1 zero, one, two, three = range(2)


         ValueError: not enough values to unpack (expected 4, got 2)


In [54]: head, *tail = range(4)
         print("head is", head)
         print("tail is", tail)

head is 0
tail is [1, 2, 3]


In [55]: one, *two, three = range(10)
```

```
In [56]: print("one is", one)
         print("two is", two)
         print("three is", three)
```

```
one is 0
two is [1, 2, 3, 4, 5, 6, 7, 8]
three is 9
```

```
In [1]: 'Dog' in ['Cat', 'Dog', 'Horse']
```

```
Out[1]: True
```

```
In [2]: 'Bird' in ['Cat', 'Dog', 'Horse']
```

```
Out[2]: False
```

```
In [3]: 2 in range(5)
```

```
Out[3]: True
```

```
In [4]: 99 in range(5)
```

```
Out[4]: False
```

```
In [5]: name = "James Philip John Hetherington".split(" ")
        print(name)
```

```
['James', 'Philip', 'John', 'Hetherington']
```

```
In [6]: name[0] = "Dr"
        name[1:3] = ["Griffiths-"]
        name.append("PhD")

        print(" ".join(name))
```

```
Dr Griffiths- Hetherington PhD
```

```
In [7]: x = 0,
        type(x)
```

```
Out[7]: tuple
```

```
In [8]: my_tuple = ("Hello", "World")
        my_tuple[0] = "Goodbye"
```

```
        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-8-242e9dae76d3> in <module>
          1 my_tuple = ("Hello", "World")
    ----> 2 my_tuple[0] = "Goodbye"


        TypeError: 'tuple' object does not support item assignment
```

```
In [9]: type(my_tuple)

Out[9]: tuple

In [10]: fish = "Hake"
         fish[0] = 'R'


         ---------------------------------------------------------------------------

         TypeError                                 Traceback (most recent call last)

         <ipython-input-10-7127277fc72e> in <module>
           1 fish = "Hake"
     ----> 2 fish[0] = 'R'


         TypeError: 'str' object does not support item assignment


In [11]: fish = "Rake" ## OK!

In [12]: x = list(range(3))
         x

Out[12]: [0, 1, 2]

In [13]: y = x
         y

Out[13]: [0, 1, 2]

In [14]: z = x[0:3]
         y[1] = "Gotcha!"

In [15]: x

Out[15]: [0, 'Gotcha!', 2]

In [16]: y

Out[16]: [0, 'Gotcha!', 2]

In [17]: z

Out[17]: [0, 1, 2]

In [18]: z[2] = "Really?"

In [19]: x

Out[19]: [0, 'Gotcha!', 2]

In [20]: y

Out[20]: [0, 'Gotcha!', 2]

In [21]: z
```

```
Out[21]: [0, 1, 'Really?']

In [22]: x = [['a', 'b'] , 'c']
         y = x
         z = x[0:2]

In [23]: x[0][1] = 'd'
         z[1] = 'e'

In [24]: x

Out[24]: [['a', 'd'], 'c']

In [25]: y

Out[25]: [['a', 'd'], 'c']

In [26]: z

Out[26]: [['a', 'd'], 'e']

In [27]: [1, 2] == [1, 2]

Out[27]: True

In [28]: [1, 2] is [1, 2]

Out[28]: False

In [29]: "Hello" == "Hello"

Out[29]: True

In [30]: "Hello" is "Hello"

Out[30]: True

In [31]: x = range(3)
         y = x
         z = x[:]

In [32]: x == y

Out[32]: True

In [33]: x is y

Out[33]: True

In [34]: x == z

Out[34]: True

In [35]: x is z

Out[35]: False

In [1]: names="Martin Luther King".split(" ")

In [2]: names[1]
```

```
Out[2]: 'Luther'

In [3]: me = { "name": "James", "age": 39,
               "Jobs": ["Programmer", "Teacher"] }

In [4]: me

Out[4]: {'name': 'James', 'age': 39, 'Jobs': ['Programmer', 'Teacher']}

In [5]: me['Jobs']

Out[5]: ['Programmer', 'Teacher']

In [6]: me['age']

Out[6]: 39

In [7]: type(me)

Out[7]: dict

In [8]: me.keys()

Out[8]: dict_keys(['name', 'age', 'Jobs'])

In [9]: me.values()

Out[9]: dict_values(['James', 39, ['Programmer', 'Teacher']])

In [10]: 'Jobs' in me

Out[10]: True

In [11]: 'James' in me

Out[11]: False

In [12]: 'James' in me.values()

Out[12]: True

In [13]: good_match = {
             ("Lamb", "Mint"): True,
             ("Bacon", "Chocolate"): False
            }

In [14]: illegal = {
             ["Lamb", "Mint"]: True,
             ["Bacon", "Chocolate"]: False
            }


         ---------------------------------------------------------------------------

         TypeError                                 Traceback (most recent call last)

         <ipython-input-14-514a4c981e6d> in <module>
           1 illegal = {
           2     ["Lamb", "Mint"]: True,
     ----> 3     ["Bacon", "Chocolate"]: False
           4    }


         TypeError: unhashable type: 'list'
```

```
In [15]: my_dict = {'0': 0, '1':1, '2': 2, '3': 3, '4': 4}
         print(my_dict)
         print(my_dict.values())
```

```
{'0': 0, '1': 1, '2': 2, '3': 3, '4': 4}
dict_values([0, 1, 2, 3, 4])
```

```
In [16]: name = "James Hetherington"
         unique_letters = set(name)

In [17]: unique_letters

Out[17]: {' ', 'H', 'J', 'a', 'e', 'g', 'h', 'i', 'm', 'n', 'o', 'r', 's', 't'}

In [18]: primes_below_ten = { 2, 3, 5, 7}

In [19]: type(unique_letters)

Out[19]: set

In [20]: type(primes_below_ten)

Out[20]: set

In [21]: unique_letters

Out[21]: {' ', 'H', 'J', 'a', 'e', 'g', 'h', 'i', 'm', 'n', 'o', 'r', 's', 't'}

In [22]: "".join(unique_letters)

Out[22]: 'enhJHgm trioas'

In [23]: x = set("Hello")
         y = set("Goodbye")

In [24]: x & y # Intersection

Out[24]: {'e', 'o'}

In [25]: x | y # Union

Out[25]: {'G', 'H', 'b', 'd', 'e', 'l', 'o', 'y'}

In [26]: y - x # y intersection with complement of x: letters in Goodbye but not in Hello

Out[26]: {'G', 'b', 'd', 'y'}

In [1]: UCL = {
            'City': 'London',
            'Street': 'Gower Street',
            'Postcode': 'WC1E 6BT'
        }

In [2]: James = {
            'City': 'London',
            'Street': 'Waterson Street',
            'Postcode': 'E2 8HH'
        }
```

```
In [3]: addresses = [UCL, James]

In [4]: addresses

Out[4]: [{'City': 'London', 'Street': 'Gower Street', 'Postcode': 'WC1E 6BT'},
         {'City': 'London', 'Street': 'Waterson Street', 'Postcode': 'E2 8HH'}]

In [5]: UCL['people'] = ['Clare','James', 'Owain']

In [6]: James['people'] = ['Sue', 'James']

In [7]: addresses

Out[7]: [{'City': 'London',
          'Street': 'Gower Street',
          'Postcode': 'WC1E 6BT',
          'people': ['Clare', 'James', 'Owain']},
         {'City': 'London',
          'Street': 'Waterson Street',
          'Postcode': 'E2 8HH',
          'people': ['Sue', 'James']}]

In [8]: UCL['Residential'] = False

In [9]: leaders = [place['people'][0] for place in addresses]
        leaders

Out[9]: ['Clare', 'Sue']

In [1]: house = {
            'living' : {
                'exits': {
                    'north' : 'kitchen',
                    'outside' : 'garden',
                    'upstairs' : 'bedroom'
                },
                'people' : ['James'],
                'capacity' : 2
            },
            'kitchen' : {
                'exits': {
                    'south' : 'living'
                },
                'people' : [],
                'capacity' : 1
            },
            'garden' : {
                'exits': {
                    'inside' : 'living'
                },
                'people' : ['Sue'],
                'capacity' : 3
            },
            'bedroom' : {
                'exits': {
                    'downstairs' : 'living',
```

```
                    'jump' : 'garden'
                },
                'people' : [],
                'capacity' : 1
            }
        }
```

```
In [1]: x = 5

        if x < 0:
            print(x, " is negative")
```

```
In [2]: x = -10

        if x < 0:
            print(x, " is negative")
```

```
-10  is negative
```

```
In [3]: x = 5
        if x < 0:
            print("x is negative")
        else:
            print("x is positive")
```

```
x is positive
```

```
In [4]: x = 5
        if x < 0:
            print("x is negative")
        elif x == 0:
            print("x is zero")
        else:
            print("x is positive")
```

```
x is positive
```

```
In [5]: choice = 'high'

        if choice == 'high':
            print(1)
        elif choice == 'medium':
            print(2)
        else:
            print(3)
```

```
1
```

```
In [6]: 1 > 2
```

```
Out[6]: False
```

```
In [7]: "UCL" > "KCL"
```

```
Out[7]: True

In [8]: "UCL" > "kcl"

Out[8]: False

In [9]: True == "True"

Out[9]: False

In [10]: '1' < 2

        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-10-2ae56e567bff> in <module>
    ----> 1 '1' < 2


        TypeError: '<' not supported between instances of 'str' and 'int'


In [11]: '5' < 2

        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-11-4b266c2a1d9b> in <module>
    ----> 1 '5' < 2


        TypeError: '<' not supported between instances of 'str' and 'int'


In [12]: '1' > 2

        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-12-142f2d5d83a7> in <module>
    ----> 1 '1' > 2


        TypeError: '>' not supported between instances of 'str' and 'int'


In [13]: mytext = "Hello"

In [14]: if mytext:
             print("Mytext is not empty")
```

```
Mytext is not empty
```

```
In [15]: mytext2 = ""

In [16]: if mytext2:
             print("Mytext2 is not empty")

In [17]: x = 3.2
         if not (x>0 and type(x)==int):
             print(x,"is not a positive integer")
```

```
3.2 is not a positive integer
```

```
In [18]: not not "Who's there!" #ăThanks to Mysterious Student

Out[18]: True

In [19]: bool("")

Out[19]: False

In [20]: bool("James")

Out[20]: True

In [21]: bool([])

Out[21]: False

In [22]: bool(['a'])

Out[22]: True

In [23]: bool({})

Out[23]: False

In [24]: bool({'name': 'James'})

Out[24]: True

In [25]: bool(0)

Out[25]: False

In [26]: bool(1)

Out[26]: True

In [27]: [] == False

Out[27]: False

In [28]: bool([]) == False

Out[28]: True

In [29]: x = 2
```

```
In [30]: if x > 0:
             print(x)


           File "<ipython-input-30-61c7132d9aa5>", line 2
         print(x)
             ^
     IndentationError: expected an indented block



In [31]: if x > 0:
                 print(x)

2


In [32]: if x > 0:
                 # print x

             print("Hello")


           File "<ipython-input-32-1045ed7694fb>", line 4
         print("Hello")
             ^
     IndentationError: expected an indented block



In [33]: if x > 0:
                 # print x
                 pass

             print("Hello")

Hello


In [1]: mylist = [3, 7, 15, 2]

In [2]: for whatever in mylist:
             print(whatever**2)

9
49
225
4


In [3]: vowels="aeiou"
        sarcasm = []

        for letter in "Okay":
            if letter.lower() in vowels:
                repetition = 3
```

```
            else:
                repetition = 1

            sarcasm.append(letter*repetition)

        "".join(sarcasm)
```

```
Out[3]: 'OOOkaaay'
```

```
In [4]: import datetime
        now = datetime.datetime.now()

        founded = {"James": 1976, "UCL": 1826, "Cambridge": 1209}

        current_year = now.year

        for thing in founded:
            print(thing, " is ", current_year -  founded[thing], "years old.")
```

```
James  is  43 years old.
UCL  is  193 years old.
Cambridge  is  810 years old.
```

```
In [5]: triples = [
            [4, 11, 15],
            [39, 4, 18]
        ]
```

```
In [6]: for whatever in triples:
            print(whatever)
```

```
[4, 11, 15]
[39, 4, 18]
```

```
In [7]: for first, middle, last in triples:
            print(middle)
```

```
11
4
```

```
In [8]: # A reminder that the words you use for variable names are arbitrary:
        for hedgehog, badger, fox in triples:
            print(badger)
```

```
11
4
```

```
In [9]: things = {"James": [1976, 'Kendal'],
                  "UCL": [1826, 'Bloomsbury'],
                  "Cambridge": [1209, 'Cambridge']}

        print(things.items())
```

```
dict_items([('James', [1976, 'Kendal']), ('UCL', [1826, 'Bloomsbury']), ('Cambridge', [1209, 'Cambridge
```

```
In [10]: for name, year in founded.items():
             print(name, " is ", current_year - year, "years old.")
```

```
James  is  43 years old.
UCL  is  193 years old.
Cambridge  is  810 years old.
```

```
In [11]: for n in range(50):
             if n == 20:
                 break
             if n % 2 == 0:
                 continue
             print(n)
```

```
1
3
5
7
9
11
13
15
17
19
```

```
In [1]: house = {
            'living' : {
                'exits': {
                    'north' : 'kitchen',
                    'outside' : 'garden',
                    'upstairs' : 'bedroom'
                },
                'people' : ['James'],
                'capacity' : 2
            },
            'kitchen' : {
                'exits': {
                    'south' : 'living'
                },
                'people' : [],
                'capacity' : 1
            },
            'garden' : {
                'exits': {
                    'inside' : 'living'
                },
                'people' : ['Sue'],
                'capacity' : 3
            },
            'bedroom' : {
```

```
                'exits': {
                    'downstairs' : 'living',
                    'jump' : 'garden'
                },
                'people' : [],
                'capacity' : 1
            }
        }
```

```
In [2]: capacity = 0
        occupancy = 0
        for name, room in house.items():
            capacity += room['capacity']
            occupancy += len(room['people'])
        print("House can fit {} people, and currently has: {}.".format(capacity,occupancy))
```

```
House can fit 7 people, and currently has: 2.
```

```
In [1]: [2**x for x in range(10)]
```

```
Out[1]: [1, 2, 4, 8, 16, 32, 64, 128, 256, 512]
```

```
In [2]: result = []
        for x in range(10):
            result.append(2**x)

        result
```

```
Out[2]: [1, 2, 4, 8, 16, 32, 64, 128, 256, 512]
```

```
In [3]: [len(str(2**x)) for x in range(10)]
```

```
Out[3]: [1, 1, 1, 1, 2, 2, 2, 3, 3, 3]
```

```
In [4]: [2**x for x in range(30) if x%3 == 0]
```

```
Out[4]: [1, 8, 64, 512, 4096, 32768, 262144, 2097152, 16777216, 134217728]
```

```
In [5]: "".join([letter for letter in "James Hetherington"
                 if letter.lower() not in 'aeiou'])
```

```
Out[5]: 'Jms Hthrngtn'
```

```
In [6]: result = []
        for x in range(30):
            if x%3 == 0:
                result.append(2**x)
        result
```

```
Out[6]: [1, 8, 64, 512, 4096, 32768, 262144, 2097152, 16777216, 134217728]
```

```
In [7]: [x - y for x in range(4) for y in range(4)]
```

```
Out[7]: [0, -1, -2, -3, 1, 0, -1, -2, 2, 1, 0, -1, 3, 2, 1, 0]
```

```
In [8]: [x - y for x in range(4) for y in range(4) if x >= y]
```

```
Out[8]: [0, 1, 0, 2, 1, 0, 3, 2, 1, 0]

In [9]: [[x - y for x in range(4)] for y in range(4)]

Out[9]: [[0, 1, 2, 3], [-1, 0, 1, 2], [-2, -1, 0, 1], [-3, -2, -1, 0]]

In [10]: [x+y for x in ['a','b','c'] for y in ['1','2','3']]

Out[10]: ['a1', 'a2', 'a3', 'b1', 'b2', 'b3', 'c1', 'c2', 'c3']

In [11]: [[x+y for x in ['a','b','c']] for y in ['1','2','3']]

Out[11]: [['a1', 'b1', 'c1'], ['a2', 'b2', 'c2'], ['a3', 'b3', 'c3']]

In [12]: {(str(x))*3: x for x in range(3)}

Out[12]: {'000': 0, '111': 1, '222': 2}

In [13]: any([True, False, True])

Out[13]: True

In [14]: all([True, False, True])

Out[14]: False

In [15]: max([1, 2, 3])

Out[15]: 3

In [16]: sum([1, 2, 3])

Out[16]: 6

In [17]: [str(x) for x in range(10)]

Out[17]: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

In [18]: list(map(str, range(10)))

Out[18]: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

In [19]: {'bedroom': 1, 'garden': 3, 'kitchen': 1, 'living': 2}

Out[19]: {'bedroom': 1, 'garden': 3, 'kitchen': 1, 'living': 2}

In [20]: {'garden': 1, 'living': 1}

Out[20]: {'garden': 1, 'living': 1}

In [1]: house = {
            'living' : {
                'exits': {
                    'north' : 'kitchen',
                    'outside' : 'garden',
                    'upstairs' : 'bedroom'
                },
                'people' : ['James'],
                'capacity' : 2
            },
```

```
            'kitchen' : {
                'exits': {
                    'south' : 'living'
                },
                'people' : [],
                'capacity' : 1
            },
            'garden' : {
                'exits': {
                    'inside' : 'living'
                },
                'people' : ['Sue'],
                'capacity' : 3
            },
            'bedroom' : {
                'exits': {
                    'downstairs' : 'living',
                    'jump' : 'garden'
                },
                'people' : [],
                'capacity' : 1
            }
        }
```

```
In [2]: {name: room['capacity'] for name, room in house.items()}
```

```
Out[2]: {'living': 2, 'kitchen': 1, 'garden': 3, 'bedroom': 1}
```

```
In [3]: {name: len(room['people']) for name, room in house.items() if len(room['people']) > 0}
```

```
Out[3]: {'living': 1, 'garden': 1}
```

```
In [1]: def double(x):
          return x*2

        print(double(5), double([5]), double('five'))
```

```
10 [5, 5] fivefive
```

```
In [2]: def jeeves(name = "Sir"):
            return "Very good, {}".format(name)
```

```
In [3]: jeeves()
```

```
Out[3]: 'Very good, Sir'
```

```
In [4]: jeeves('James')
```

```
Out[4]: 'Very good, James'
```

```
In [5]: def jeeves(greeting="Very good", name="Sir"):
            return "{}, {}".format(greeting, name)
```

```
In [6]: jeeves()
```

```
Out[6]: 'Very good, Sir'
```

```
In [7]: jeeves("Hello")

Out[7]: 'Hello, Sir'

In [8]: jeeves(name = "James")

Out[8]: 'Very good, James'

In [9]: jeeves(greeting="Suits you")

Out[9]: 'Suits you, Sir'

In [10]: jeeves("Hello", "Sailor")

Out[10]: 'Hello, Sailor'

In [11]: def double_inplace(vec):
             vec[:] = [element*2 for element in vec]

         z = list(range(4))
         double_inplace(z)
         print(z)

[0, 2, 4, 6]


In [12]: letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g']
         letters[:] = []

In [13]: def double(vec):
             return [element*2 for element in vec]

In [14]: x = 5
         x = 7
         x = ['a','b','c']
         y = x

In [15]: x

Out[15]: ['a', 'b', 'c']

In [16]: x[:] = ["Hooray!", "Yippee"]

In [17]: y

Out[17]: ['Hooray!', 'Yippee']

In [18]: def extend(to, vec, pad):
             if len(vec) >= to:
                 return # Exit early, list is already long enough.

             vec[:] = vec + [pad]*(to-len(vec))

In [19]: x = list(range(3))
         extend(6, x, 'a')
         print(x)

[0, 1, 2, 'a', 'a', 'a']
```

```
In [20]: z = range(9)
         extend(6, z, 'a')
         print(z)

range(0, 9)


In [21]: def arrow(before, after):
             return str(before) + " -> " + str(after)

         arrow(1, 3)

Out[21]: '1 -> 3'

In [22]: x = [1,-1]
         arrow(*x)

Out[22]: '1 -> -1'

In [23]: charges = {"neutron": 0, "proton": 1, "electron": -1}
         for particle in charges.items():
             print(arrow(*particle))

neutron -> 0
proton -> 1
electron -> -1


In [24]: def doubler(*sequence):
             return [x*2 for x in sequence]

In [25]: doubler(1,2,3)

Out[25]: [2, 4, 6]

In [26]: doubler(5, 2, "Wow!")

Out[26]: [10, 4, 'Wow!Wow!']

In [27]: def arrowify(**args):
             for key, value in args.items():
                 print(key + " -> " + value)

         arrowify(neutron="n", proton="p", electron="e")

neutron -> n
proton -> p
electron -> e


In [28]: def somefunc(a, b, *args, **kwargs):
             print("A:", a)
             print("B:", b)
             print("args:", args)
             print("keyword args", kwargs)

In [29]: somefunc(1, 2, 3, 4, 5, fish="Haddock")
```

```
A: 1
B: 2
args: (3, 4, 5)
keyword args {'fish': 'Haddock'}


In [1]: math.sin(1.6)


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-1-12dcc3af2e0c> in <module>
    ----> 1 math.sin(1.6)


        NameError: name 'math' is not defined


In [2]: import math

In [3]: math.sin(1.6)

Out[3]: 0.9995736030415051

In [4]: type(math)

Out[4]: module

In [5]: dir(math)

Out[5]: ['__doc__',
         '__file__',
         '__loader__',
         '__name__',
         '__package__',
         '__spec__',
         'acos',
         'acosh',
         'asin',
         'asinh',
         'atan',
         'atan2',
         'atanh',
         'ceil',
         'copysign',
         'cos',
         'cosh',
         'degrees',
         'e',
         'erf',
         'erfc',
         'exp',
         'expm1',
         'fabs',
```

```
 'factorial',
 'floor',
 'fmod',
 'frexp',
 'fsum',
 'gamma',
 'gcd',
 'hypot',
 'inf',
 'isclose',
 'isfinite',
 'isinf',
 'isnan',
 'ldexp',
 'lgamma',
 'log',
 'log10',
 'log1p',
 'log2',
 'modf',
 'nan',
 'pi',
 'pow',
 'radians',
 'remainder',
 'sin',
 'sinh',
 'sqrt',
 'tan',
 'tanh',
 'tau',
 'trunc']
```

```
In [6]: math.pi
```

```
Out[6]: 3.141592653589793
```

```
In [7]: print(math.__file__[0:50])
        print(math.__file__[50:])
```

```
/usr/local/Cellar/python/3.7.2_1/Frameworks/Python
.framework/Versions/3.7/lib/python3.7/lib-dynload/math.cpython-37m-darwin.so
```

```
In [8]: import math
        math.sin(math.pi)
```

```
Out[8]: 1.2246467991473532e-16
```

```
In [9]: from math import sin
        sin(math.pi)
```

```
Out[9]: 1.2246467991473532e-16
```

```
In [10]: from math import *
         sin(pi)
```

```
Out[10]: 1.2246467991473532e-16

In [11]: import math as m
         m.cos(0)

Out[11]: 1.0

In [12]: pi = 3
         from math import pi as realpi
         print(sin(pi), sin(realpi))

0.1411200080598672 1.2246467991473532e-16


In [1]: class Room(object):
            pass

In [2]: class Room():
            pass

In [3]: class Room:
            pass

In [4]: zero = int()
        type(zero)

Out[4]: int

In [5]: myroom = Room()
        type(myroom)

Out[5]: __main__.Room

In [6]: myroom.name = "Living"

In [7]: myroom.name

Out[7]: 'Living'

In [8]: myroom.capacity = 3
        myroom.occupants = ["James", "Sue"]

In [9]: class Room(object):
            def overfull(self):
                return len(self.occupants) > self.capacity

In [10]: myroom = Room()
         myroom.capacity = 3
         myroom.occupants = ["James", "Sue"]

In [11]: myroom.overfull()

Out[11]: False

In [12]: myroom.occupants.append(['Clare'])

In [13]: myroom.occupants.append(['Bob'])

In [14]: myroom.overfull()
```

```
Out[14]: True

In [15]: class Room(object):
             def __init__(self, name, exits, capacity, occupants=[]):
                 self.name = name
                 self.occupants = occupants  # Note the default argument, occupants start empty
                 self.exits = exits
                 self.capacity = capacity

             def overfull(self):
                 return len(self.occupants) > self.capacity

In [16]: living = Room("Living Room", {'north': 'garden'}, 3)

In [17]: living.capacity

Out[17]: 3

In [18]: class Maze(object):
             def __init__(self, name):
                 self.name = name
                 self.rooms = {}

             def add_room(self, room):
                 room.maze = self  # The Room needs to know
                                   # which Maze it is a part of
                 self.rooms[room.name] = room

             def occupants(self):
                 return [occupant for room in self.rooms.values()
                         for occupant in room.occupants.values()]

             def wander(self):
                 """Move all the people in a random direction"""
                 for occupant in self.occupants():
                     occupant.wander()

             def describe(self):
                 for room in self.rooms.values():
                     room.describe()

             def step(self):
                 self.describe()
                 print("")
                 self.wander()
                 print("")

             def simulate(self, steps):
                 for _ in range(steps):
                     self.step()

In [19]: class Room(object):
             def __init__(self, name, exits, capacity, maze=None):
                 self.maze = maze
                 self.name = name
```

```
            self.occupants = {}  # Note the default argument, occupants start empty
            self.exits = exits  # Should be a dictionary from directions to room names
            self.capacity = capacity

        def has_space(self):
            return len(self.occupants) < self.capacity

        def available_exits(self):
            return [exit for exit, target in self.exits.items()
                    if self.maze.rooms[target].has_space()]

        def random_valid_exit(self):
            import random
            if not self.available_exits():
                return None
            return random.choice(self.available_exits())

        def destination(self, exit):
            return self.maze.rooms[self.exits[exit]]

        def add_occupant(self, occupant):
            occupant.room = self  # The person needs to know which room it is in
            self.occupants[occupant.name] = occupant

        def delete_occupant(self, occupant):
            del self.occupants[occupant.name]

        def describe(self):
            if self.occupants:
                print(f"{self.name}: " + " ".join(self.occupants.keys()))
```

```
In [20]: class Person(object):
            def __init__(self, name, room=None):
                self.name = name

            def use(self, exit):
                self.room.delete_occupant(self)
                destination = self.room.destination(exit)
                destination.add_occupant(self)
                print("{some} goes {action} to the {where}".format(some=self.name,
                                                                   action=exit,
                                                                   where=destination.name))

            def wander(self):
                exit = self.room.random_valid_exit()
                if exit:
                    self.use(exit)
```

```
In [21]: james = Person('James')
         sue = Person('Sue')
         bob = Person('Bob')
         clare = Person('Clare')
```

```
In [22]: living = Room('livingroom', {'outside': 'garden',
                                      'upstairs': 'bedroom', 'north': 'kitchen'}, 2)
```

```
        kitchen = Room('kitchen', {'south': 'livingroom'}, 1)
        garden = Room('garden', {'inside': 'livingroom'}, 3)
        bedroom = Room('bedroom', {'jump': 'garden', 'downstairs': 'livingroom'}, 1)

In [23]: house = Maze('My House')

In [24]: for room in [living, kitchen, garden, bedroom]:
             house.add_room(room)

In [25]: living.add_occupant(james)

In [26]: garden.add_occupant(sue)
         garden.add_occupant(clare)

In [27]: bedroom.add_occupant(bob)

In [28]: house.simulate(3)

livingroom: James
garden: Sue Clare
bedroom: Bob

James goes north to the kitchen
Sue goes inside to the livingroom
Clare goes inside to the livingroom
Bob goes jump to the garden

livingroom: Sue Clare
kitchen: James
garden: Bob

Sue goes upstairs to the bedroom
Clare goes outside to the garden
James goes south to the livingroom
Bob goes inside to the livingroom

livingroom: James Bob
garden: Clare
bedroom: Sue

James goes north to the kitchen
Bob goes outside to the garden
Clare goes inside to the livingroom
Sue goes downstairs to the livingroom



In [29]: class Maze():
             def __init__(self, name):
                 self.name = name
                 self.rooms = []
                 self.occupants = []

             def add_room(self, name, capacity):
                 result = Room(name, capacity)
                 self.rooms.append(result)
```

```
            return result

        def add_exit(self, name, source, target, reverse=None):
            source.add_exit(name, target)
            if reverse:
                target.add_exit(reverse, source)

        def add_occupant(self, name, room):
            self.occupants.append(Person(name, room))
            room.occupancy += 1

        def wander(self):
            "Move all the people in a random direction"
            for occupant in self.occupants:
                occupant.wander()

        def describe(self):
            for occupant in self.occupants:
                occupant.describe()

        def step(self):
            self.describe()
            print("")
            self.wander()
            print("")

        def simulate(self, steps):
            for _ in range(steps):
                self.step()
```

```
In [30]: class Room:
        def __init__(self, name, capacity):
            self.name = name
            self.capacity = capacity
            self.occupancy = 0
            self.exits = []

        def has_space(self):
            return self.occupancy < self.capacity

        def available_exits(self):
            return [exit for exit in self.exits if exit.valid()]

        def random_valid_exit(self):
            import random
            if not self.available_exits():
                return None
            return random.choice(self.available_exits())

        def add_exit(self, name, target):
            self.exits.append(Exit(name, target))
```

```
In [31]: class Person():
        def __init__(self, name, room=None):
            self.name = name
```

```
            self.room = room

        def use(self, exit):
            self.room.occupancy -= 1
            destination = exit.target
            destination.occupancy += 1
            self.room = destination
            print("{some} goes {action} to the {where}".format(some=self.name,
                                                               action=exit.name,
                                                               where=destination.name))

        def wander(self):
            exit = self.room.random_valid_exit()
            if exit:
                self.use(exit)

        def describe(self):
            print("{who} is in the {where}".format(who=self.name,
                                                   where=self.room.name))
```

```
In [32]: class Exit:
             def __init__(self, name, target):
                 self.name = name
                 self.target = target

             def valid(self):
                 return self.target.has_space()
```

```
In [33]: house = Maze('My New House')
```

```
In [34]: living = house.add_room('livingroom', 2)
         bed = house.add_room('bedroom', 1)
         garden = house.add_room('garden', 3)
         kitchen = house.add_room('kitchen', 1)
```

```
In [35]: house.add_exit('north', living, kitchen, 'south')
```

```
In [36]: house.add_exit('upstairs', living, bed, 'downstairs')
```

```
In [37]: house.add_exit('outside', living, garden, 'inside')
```

```
In [38]: house.add_exit('jump', bed, garden)
```

```
In [39]: house.add_occupant('James', living)
         house.add_occupant('Sue', garden)
         house.add_occupant('Bob', bed)
         house.add_occupant('Clare', garden)
```

```
In [40]: house.simulate(3)
```

```
James is in the livingroom
Sue is in the garden
Bob is in the bedroom
Clare is in the garden

James goes outside to the garden
```

```
Sue goes inside to the livingroom
Bob goes downstairs to the livingroom

James is in the garden
Sue is in the livingroom
Bob is in the livingroom
Clare is in the garden

Sue goes north to the kitchen
Bob goes outside to the garden
Clare goes inside to the livingroom

James is in the garden
Sue is in the kitchen
Bob is in the garden
Clare is in the livingroom

James goes inside to the livingroom
Clare goes upstairs to the bedroom
```

```
In [1]: %%writefile mydata.txt
        A poet once said, 'The whole universe is in a glass of wine.'
        We will probably never know in what sense he meant it,
        for poets do not write to be understood.
        But it is true that if we look at a glass of wine closely enough we see the entire universe.
        There are the things of physics: the twisting liquid which evaporates depending
        on the wind and weather, the reflection in the glass;
        and our imagination adds atoms.
        The glass is a distillation of the earth's rocks,
        and in its composition we see the secrets of the universe's age, and the evolution of stars.
        What strange array of chemicals are in the wine? How did they come to be?
        There are the ferments, the enzymes, the substrates, and the products.
        There in wine is found the great generalization; all life is fermentation.
        Nobody can discover the chemistry of wine without discovering,
        as did Louis Pasteur, the cause of much disease.
        How vivid is the claret, pressing its existence into the consciousness that watches it!
        If our small minds, for some convenience, divide this glass of wine, this universe,
        into parts --
        physics, biology, geology, astronomy, psychology, and so on --
        remember that nature does not know it!

        So let us put it all back together, not forgetting ultimately what it is for.
        Let it give us one more final pleasure; drink it and forget it all!
           - Richard Feynman

Overwriting mydata.txt
```

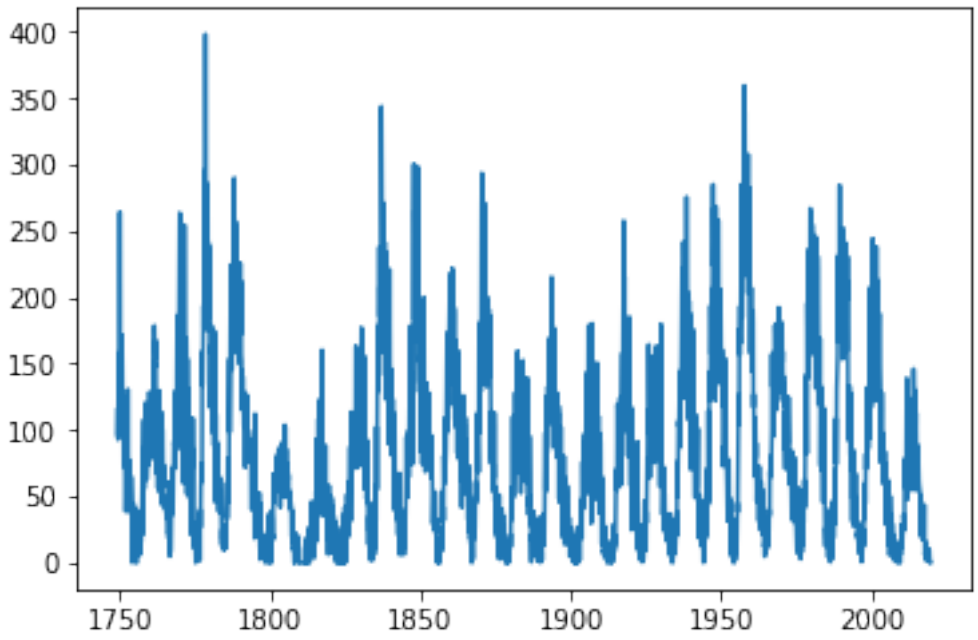
```
In [2]: import os # The 'os' module gives us all the tools we need to search in the file system
        os.getcwd() # Use the 'getcwd' function from the 'os' module to find where we are on disk.

Out[2]: '/Users/edaub/Projects/rsd-engineeringcourse/ch01data'

In [3]: import os
        [x for x in os.listdir(os.getcwd()) if ".txt" in x]
```

```
Out[3]: ['mydata.txt']

In [4]: os.path.dirname(os.getcwd())

Out[4]: '/Users/edaub/Projects/rsd-engineeringcourse'

In [5]: "/".join(os.getcwd().split("/")[:-1])

Out[5]: '/Users/edaub/Projects/rsd-engineeringcourse'

In [6]: myfile = open('mydata.txt')

In [7]: type(myfile)

Out[7]: _io.TextIOWrapper

In [8]: [x for x in myfile]

Out[8]: ["A poet once said, 'The whole universe is in a glass of wine.'\n",
         'We will probably never know in what sense he meant it, \n',
         'for poets do not write to be understood. \n',
         'But it is true that if we look at a glass of wine closely enough we see the entire universe. '
         'There are the things of physics: the twisting liquid which evaporates depending\n',
         'on the wind and weather, the reflection in the glass;\n',
         'and our imagination adds atoms.\n',
         "The glass is a distillation of the earth's rocks,\n",
         "and in its composition we see the secrets of the universe's age, and the evolution of stars. "
         'What strange array of chemicals are in the wine? How did they come to be? \n',
         'There are the ferments, the enzymes, the substrates, and the products.\n',
         'There in wine is found the great generalization; all life is fermentation.\n',
         'Nobody can discover the chemistry of wine without discovering, \n',
         'as did Louis Pasteur, the cause of much disease.\n',
         'How vivid is the claret, pressing its existence into the consciousness that watches it!\n',
         'If our small minds, for some convenience, divide this glass of wine, this universe, \n',
         'into parts -- \n',
         'physics, biology, geology, astronomy, psychology, and so on -- \n',
         'remember that nature does not know it!\n',
         '\n',
         'So let us put it all back together, not forgetting ultimately what it is for.\n',
         'Let it give us one more final pleasure; drink it and forget it all!\n',
         '   - Richard Feynman\n']

In [9]: [x for x in myfile]

Out[9]: []

In [10]: myfile.seek(0)
         [len(x) for x in myfile if 'know' in x]

Out[10]: [56, 39]

In [11]: myfile.seek(0)
         first = myfile.readline()

In [12]: first

Out[12]: "A poet once said, 'The whole universe is in a glass of wine.'\n"
```

```
In [13]: second = myfile.readline()

In [14]: second

Out[14]: 'We will probably never know in what sense he meant it, \n'

In [15]: rest = myfile.read()

In [16]: rest

Out[16]: "for poets do not write to be understood. \nBut it is true that if we look at a glass of wine

In [17]: open('mydata.txt').read()

Out[17]: "A poet once said, 'The whole universe is in a glass of wine.'\nWe will probably never know in

In [18]: myfile.seek(1335)

Out[18]: 1335

In [19]: myfile.read(15)

Out[19]: '\n   - Richard F'

In [20]: mystring = "Hello World\n My name is James"

In [21]: mystring

Out[21]: 'Hello World\n My name is James'

In [22]: mystring.readline()

        ---------------------------------------------------------------------------

        AttributeError                            Traceback (most recent call last)

        <ipython-input-22-8fadd4a635f7> in <module>
    ----> 1 mystring.readline()


        AttributeError: 'str' object has no attribute 'readline'


In [23]: from io import StringIO

In [24]: mystringasafile = StringIO(mystring)

In [25]: mystringasafile.readline()

Out[25]: 'Hello World\n'

In [26]: mystringasafile.readline()

Out[26]: ' My name is James'

In [27]: myfile.close()
```

```
In [28]: with open('mydata.txt') as somefile:
             content = somefile.read()

         content

Out[28]: "A poet once said, 'The whole universe is in a glass of wine.'\nWe will probably never know in

In [29]: with open('mywrittenfile', 'w') as target:
             target.write('Hello')
             target.write('World')

In [30]: with open('mywrittenfile','r') as source:
             print(source.read())

HelloWorld


In [31]: with open('mywrittenfile', 'a') as target:
             target.write('Hello')
             target.write('James')

In [32]: with open('mywrittenfile','r') as source:
             print(source.read())

HelloWorldHelloJames


In [1]: "http://maps.googleapis.com:80/maps/api/staticmap?size=400x400&center=51.51,-0.1275&zoom=12"

Out[1]: 'http://maps.googleapis.com:80/maps/api/staticmap?size=400x400&center=51.51,-0.1275&zoom=12'

In [2]: import requests

In [3]: response = requests.get("http://maps.googleapis.com/maps/api/staticmap",
                                params={
                 'size': '400x400',
                 'center': '51.51,-0.1275',
                 'zoom': 12
             })

In [4]: response.content[0:50]

Out[4]: b'The Google Maps Platform server rejected your requ'

In [5]: spots = requests.get('http://www.sidc.be/silso/INFO/snmtotcsv.php').text

In [6]: spots[0:80]

Out[6]: '1749;01;1749.042;  96.7; -1.0;   -1;1\n1749;02;1749.123; 104.3; -1.0;   -1;1\n1749'

In [7]: lines = spots.split("\n")
        lines[0:5]

Out[7]: ['1749;01;1749.042;  96.7; -1.0;   -1;1',
         '1749;02;1749.123; 104.3; -1.0;   -1;1',
         '1749;03;1749.204; 116.7; -1.0;   -1;1',
         '1749;04;1749.288;  92.8; -1.0;   -1;1',
         '1749;05;1749.371; 141.7; -1.0;   -1;1']
```

```
In [8]: years=[line.split(";")[0] for line in lines]

In [9]: years[0:15]

Out[9]: ['1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1749',
         '1750',
         '1750',
         '1750']

In [1]: import requests
        spots = requests.get('http://www.sidc.be/silso/INFO/snmtotcsv.php')
        spots.text.split('\n')[0]

Out[1]: '1749;01;1749.042;  96.7; -1.0;   -1;1'

In [2]: import numpy as np
        import requests

In [3]: spots = requests.get('http://www.sidc.be/silso/INFO/snmtotcsv.php', stream=True)

In [4]: sunspots = np.genfromtxt(spots.raw, delimiter=';')

In [5]: sunspots[0][3]

Out[5]: 96.7

In [6]: %matplotlib inline

        from matplotlib import pyplot as plt
        plt.plot(sunspots[:,2], sunspots[:,3]) # Numpy syntax to access all
                                               # rows, specified column.

Out[6]: [<matplotlib.lines.Line2D at 0x112e0bc50>]
```



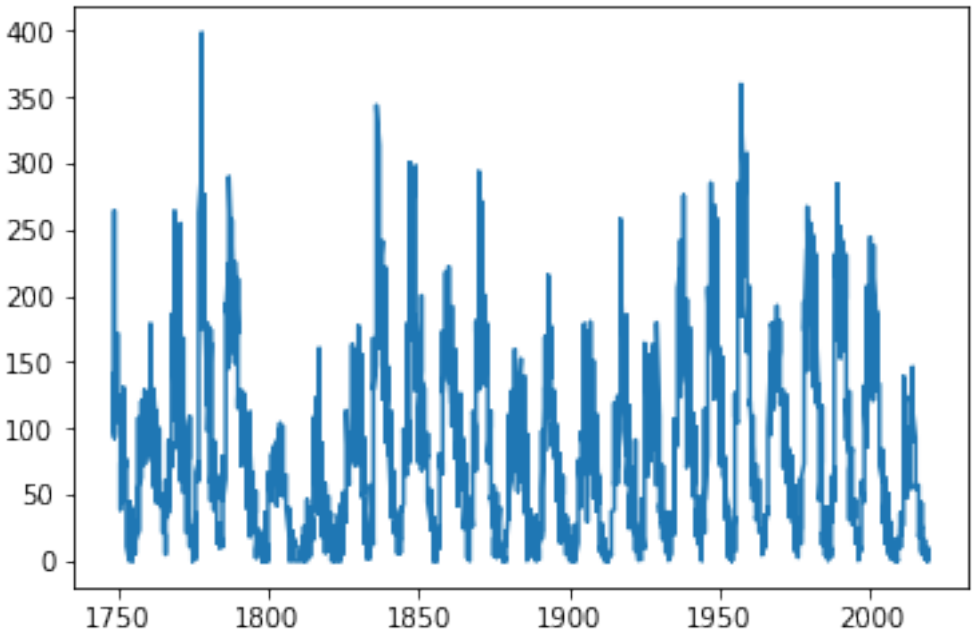

```
In [7]: spots = requests.get('http://www.sidc.be/silso/INFO/snmtotcsv.php', stream=True)

        sunspots = np.genfromtxt(spots.raw, delimiter=';',
                                 names=['year','month','date',
                                 'mean','deviation','observations','definitive'])

In [8]: sunspots

Out[8]: array([(1749.,  1., 1749.042,  96.7, -1. , -1.000e+00, 1.),
               (1749.,  2., 1749.123, 104.3, -1. , -1.000e+00, 1.),
               (1749.,  3., 1749.204, 116.7, -1. , -1.000e+00, 1.), ...,
               (2019.,  8., 2019.623,   0.7,  0.2,  1.157e+03, 0.),
               (2019.,  9., 2019.705,   1.1,  0.1,  9.920e+02, 0.),
               (2019., 10., 2019.79 ,   0.4,  0.1,  8.570e+02, 0.)],
              dtype=[('year', '<f8'), ('month', '<f8'), ('date', '<f8'), ('mean', '<f8'), ('deviation',

In [9]: spots = requests.get('http://www.sidc.be/silso/INFO/snmtotcsv.php', stream=True)

        sunspots = np.genfromtxt(spots.raw, delimiter=';',
                                 names=['year','month','date',
                                 'mean','deviation','observations','definitive'],
                                 dtype=[int, int, float, float, float, int, int])

In [10]: sunspots

Out[10]: array([(1749,  1, 1749.042,  96.7, -1. ,   -1, 1),
                (1749,  2, 1749.123, 104.3, -1. ,   -1, 1),
                (1749,  3, 1749.204, 116.7, -1. ,   -1, 1), ...,
                (2019,  8, 2019.623,   0.7,  0.2, 1157, 0),
                (2019,  9, 2019.705,   1.1,  0.1,  992, 0),
                (2019, 10, 2019.79 ,   0.4,  0.1,  857, 0)],
               dtype=[('year', '<i8'), ('month', '<i8'), ('date', '<f8'), ('mean', '<f8'), ('deviation'
```

```
In [11]: sunspots['year']
```

```
Out[11]: array([1749, 1749, 1749, ..., 2019, 2019, 2019])
```

```
In [12]: plt.plot(sunspots['year'], sunspots['mean'])
```

```
Out[12]: [<matplotlib.lines.Line2D at 0x107197668>]
```




```
In [1]: import json
```

```
In [2]: mydata = {'key': ['value1', 'value2'],
                  'key2': {'key4':'value3'}}
```

```
In [3]: json.dumps(mydata)
```

```
Out[3]: '{"key": ["value1", "value2"], "key2": {"key4": "value3"}}'
```

```
In [4]: print(json.dumps(mydata, indent=4))
```

```
{
    "key": [
        "value1",
        "value2"
    ],
    "key2": {
        "key4": "value3"
    }
}
```

```
In [5]: %%writefile myfile.json
        {
            "somekey": ["a list", "with values"]
        }
```

```
Overwriting myfile.json
```

```
In [6]: with open('myfile.json', 'r') as f:
            mydataasstring = f.read()
```

```
In [7]: mydataasstring
```

```
Out[7]: '{\n    "somekey": ["a list", "with values"]\n}\n'
```

```
In [8]: mydata = json.loads(mydataasstring)
```

```
In [9]: mydata['somekey']
```

```
Out[9]: ['a list', 'with values']
```

```
In [10]: %%writefile myfile.yaml
         somekey:
             - a list # Look, this is a list
             - with values
```

```
Overwriting myfile.yaml
```

```
In [11]: import yaml  # This may need installed as pyyaml
```

```
In [12]: mydata = yaml.load(open('myfile.yaml'))
         print(mydata)
```

```
{'somekey': ['a list', 'with values']}
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/vendor/lib/python3.7/site-packages/ipykernel_launcher.py:1: YAM
  """Entry point for launching an IPython kernel.
```

```
In [13]: print(yaml.safe_dump(mydata))
```

```
somekey:
- a list
- with values
```

```
In [14]: print(yaml.safe_dump(mydata, default_flow_style=False))
```

```
somekey:
- a list
- with values
```

```
In [1]: house = {
            'living': {
                'exits': {
                    'north': 'kitchen',
                    'outside': 'garden',
                    'upstairs': 'bedroom'
```

```
            },
            'people': ['James'],
            'capacity': 2
        },
        'kitchen': {
            'exits': {
                'south': 'living'
            },
            'people': [],
            'capacity': 1
        },
        'garden': {
            'exits': {
                'inside': 'living'
            },
            'people': ['Sue'],
            'capacity': 3
        },
        'bedroom': {
            'exits': {
                'downstairs': 'living',
                'jump': 'garden'
            },
            'people': [],
            'capacity': 1
        }
    }
```

```
In [2]: import json
```

```
In [3]: with open('maze.json', 'w') as json_maze_out:
            json_maze_out.write(json.dumps(house))
```

```
In [4]: %%bash
        cat 'maze.json'
```

```
{"living": {"exits": {"north": "kitchen", "outside": "garden", "upstairs": "bedroom"}, "people": ["James
```

```
In [5]: with open('maze.json') as json_maze_in:
            maze_again = json.load(json_maze_in)
```

```
In [6]: maze_again
```

```
Out[6]: {'living': {'exits': {'north': 'kitchen',
           'outside': 'garden',
           'upstairs': 'bedroom'},
          'people': ['James'],
          'capacity': 2},
         'kitchen': {'exits': {'south': 'living'}, 'people': [], 'capacity': 1},
         'garden': {'exits': {'inside': 'living'}, 'people': ['Sue'], 'capacity': 3},
         'bedroom': {'exits': {'downstairs': 'living', 'jump': 'garden'},
          'people': [],
          'capacity': 1}}
```

```
In [7]: import yaml
```

```
In [8]: with open('maze.yaml', 'w') as yaml_maze_out:
            yaml_maze_out.write(yaml.dump(house))

In [9]: %%bash
        cat 'maze.yaml'

bedroom:
  capacity: 1
  exits:
    downstairs: living
    jump: garden
  people: []
garden:
  capacity: 3
  exits:
    inside: living
  people:
  - Sue
kitchen:
  capacity: 1
  exits:
    south: living
  people: []
living:
  capacity: 2
  exits:
    north: kitchen
    outside: garden
    upstairs: bedroom
  people:
  - James


In [10]: with open('maze.yaml') as yaml_maze_in:
             maze_again = yaml.load(yaml_maze_in)

/usr/local/Cellar/ipython/7.2.0/libexec/vendor/lib/python3.7/site-packages/ipykernel_launcher.py:2: YAM



In [11]: maze_again

Out[11]: {'bedroom': {'capacity': 1,
           'exits': {'downstairs': 'living', 'jump': 'garden'},
           'people': []},
          'garden': {'capacity': 3, 'exits': {'inside': 'living'}, 'people': ['Sue']},
          'kitchen': {'capacity': 1, 'exits': {'south': 'living'}, 'people': []},
          'living': {'capacity': 2,
           'exits': {'north': 'kitchen', 'outside': 'garden', 'upstairs': 'bedroom'},
           'people': ['James']}}

In [1]: import requests
        quakes = requests.get("http://earthquake.usgs.gov/fdsnws/event/1/query.geojson",
                              params={
                                  'starttime': "2000-01-01",
                                  "maxlatitude": "58.723",
```

```
                                    "minlatitude": "50.008",
                                    "maxlongitude": "1.67",
                                    "minlongitude": "-9.756",
                                    "minmagnitude": "1",
                                    "endtime": "2018-10-11",
                                    "orderby": "time-asc"}
                                )
```

```
In [2]: quakes.text[0:100]
```

```
Out[2]: '{"type":"FeatureCollection","metadata":{"generated":1572866320000,"url":"https://earthquake.us
```

```
In [1]: import requests
        quakes = requests.get("http://earthquake.usgs.gov/fdsnws/event/1/query.geojson",
                              params={
                                    'starttime': "2000-01-01",
                                    "maxlatitude": "58.723",
                                    "minlatitude": "50.008",
                                    "maxlongitude": "1.67",
                                    "minlongitude": "-9.756",
                                    "minmagnitude": "1",
                                    "endtime": "2018-10-11",
                                    "orderby": "time-asc"}
                              )
```

```
In [2]: import json
```

```
In [3]: requests_json = json.loads(quakes.text)
```

```
In [4]: type(requests_json)
```

```
Out[4]: dict
```

```
In [5]: requests_json.keys()
```

```
Out[5]: dict_keys(['type', 'metadata', 'features', 'bbox'])
```

```
In [6]: len(requests_json['features'])
```

```
Out[6]: 120
```

```
In [7]: requests_json['features'][0].keys()
```

```
Out[7]: dict_keys(['type', 'properties', 'geometry', 'id'])
```

```
In [8]: requests_json['features'][0]['properties'].keys()
```

```
Out[8]: dict_keys(['mag', 'place', 'time', 'updated', 'tz', 'url', 'detail', 'felt', 'cdi', 'mmi', 'ale
```

```
In [9]: requests_json['features'][0]['properties']['mag']
```

```
Out[9]: 2.6
```

```
In [10]: requests_json['features'][0]['geometry']
```

```
Out[10]: {'type': 'Point', 'coordinates': [-2.81, 54.77, 14]}
```

```
In [11]: quakes = requests_json['features']
```

```
In [12]: largest_so_far = quakes[0]
         for quake in quakes:
             if quake['properties']['mag'] > largest_so_far['properties']['mag']:
                 largest_so_far = quake
         largest_so_far['properties']['mag']
```

```
Out[12]: 4.8
```

```
In [13]: lat = largest_so_far['geometry']['coordinates'][1]
         long = largest_so_far['geometry']['coordinates'][0]
         print("Latitude: {} Longitude: {}".format(lat, long))
```

```
Latitude: 52.52 Longitude: -2.15
```

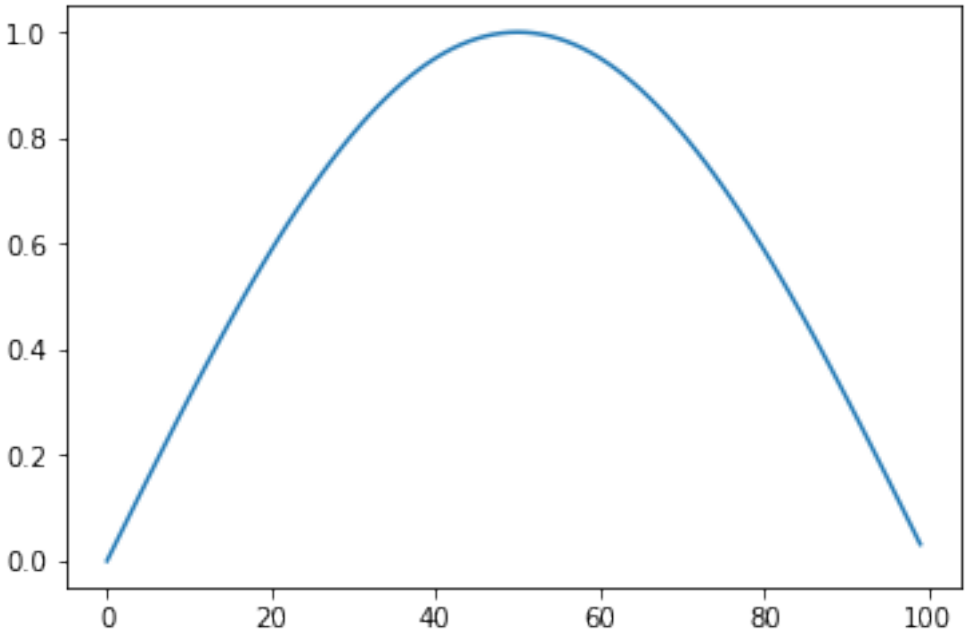
```
In [14]: import requests


         def request_map_at(lat, long, satellite=True,
                            zoom=10, size=(400, 400)):
             base = "https://static-maps.yandex.ru/1.x/?"

             params = dict(
                 z=zoom,
                 size="{},{}".format(size[0], size[1]),
                 ll="{},{}".format(long, lat),
                 l="sat" if satellite else "map",
                 lang="en_US"
             )

             return requests.get(base, params=params)
```

```
In [15]: map_png = request_map_at(lat, long, zoom=10, satellite=False)
```

```
In [16]: from IPython.display import Image
         Image(map_png.content)
```

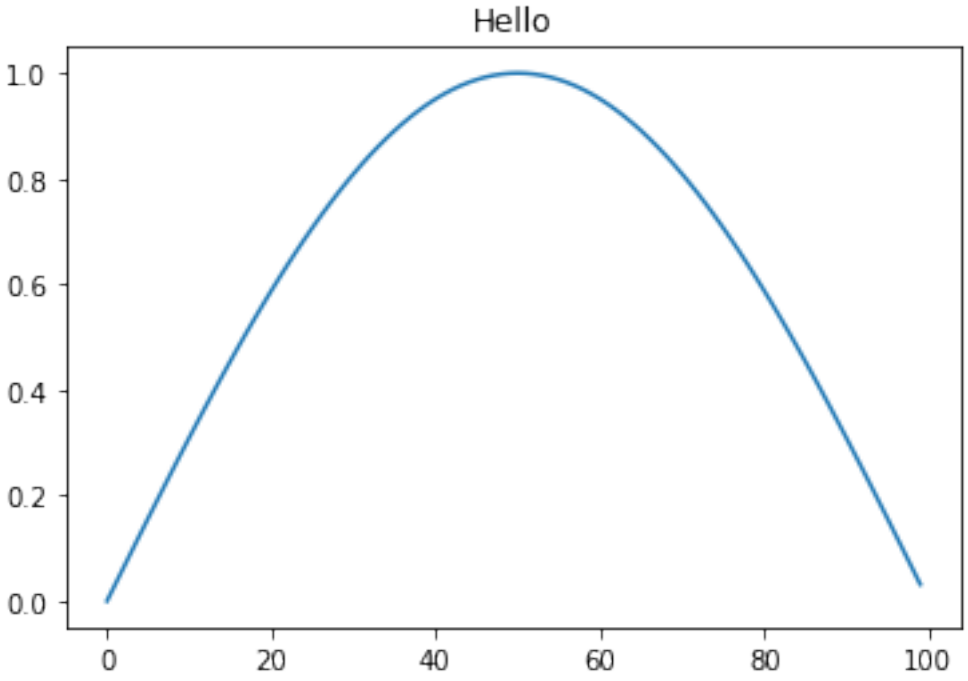
```
Out[16]:
```


```
In [1]: from matplotlib import pyplot as plt

In [2]: %matplotlib inline

In [3]: from math import sin, cos, pi
        myfig = plt.plot([sin(pi*x/100.0) for x in range(100)])
```

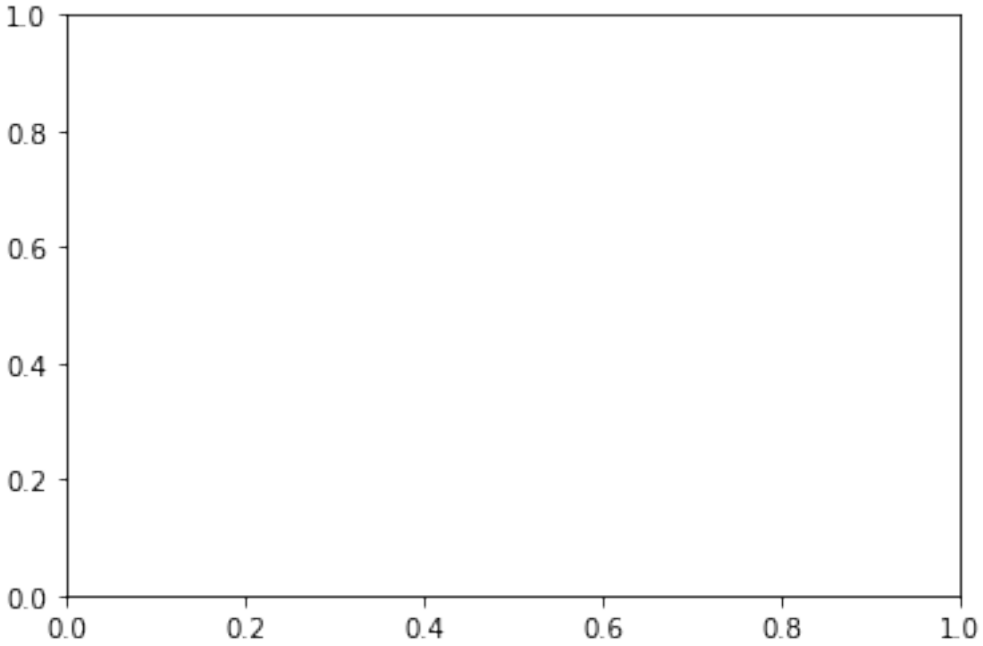



```
In [4]: plt.plot([sin(pi*x/100.0) for x in range(100)])
        plt.title("Hello")
```

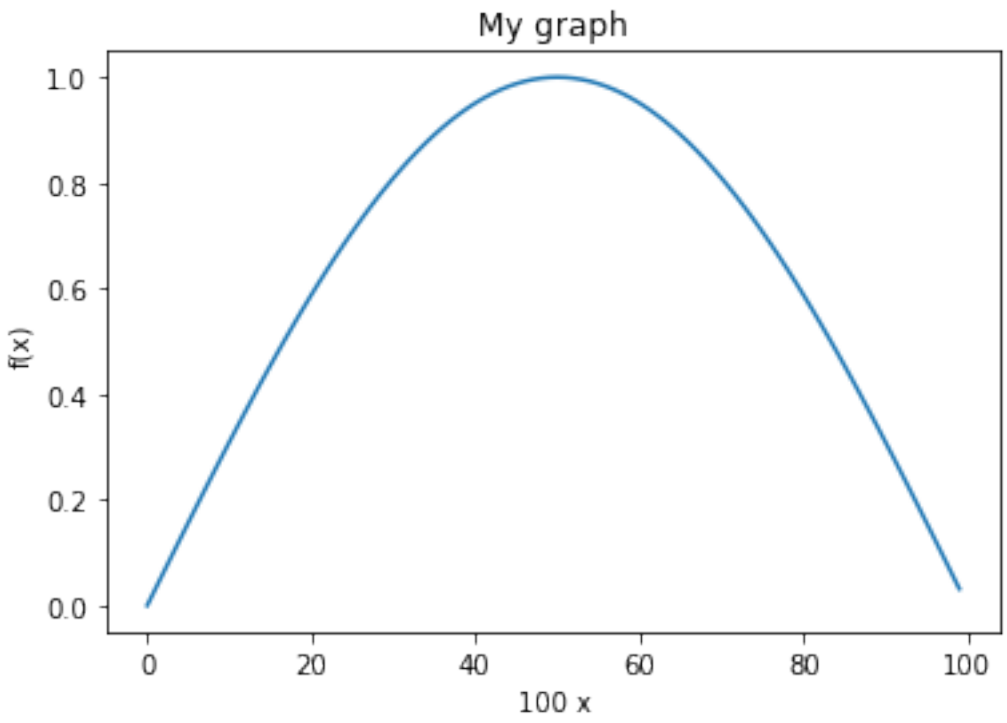
```
Out[4]: Text(0.5, 1.0, 'Hello')
```



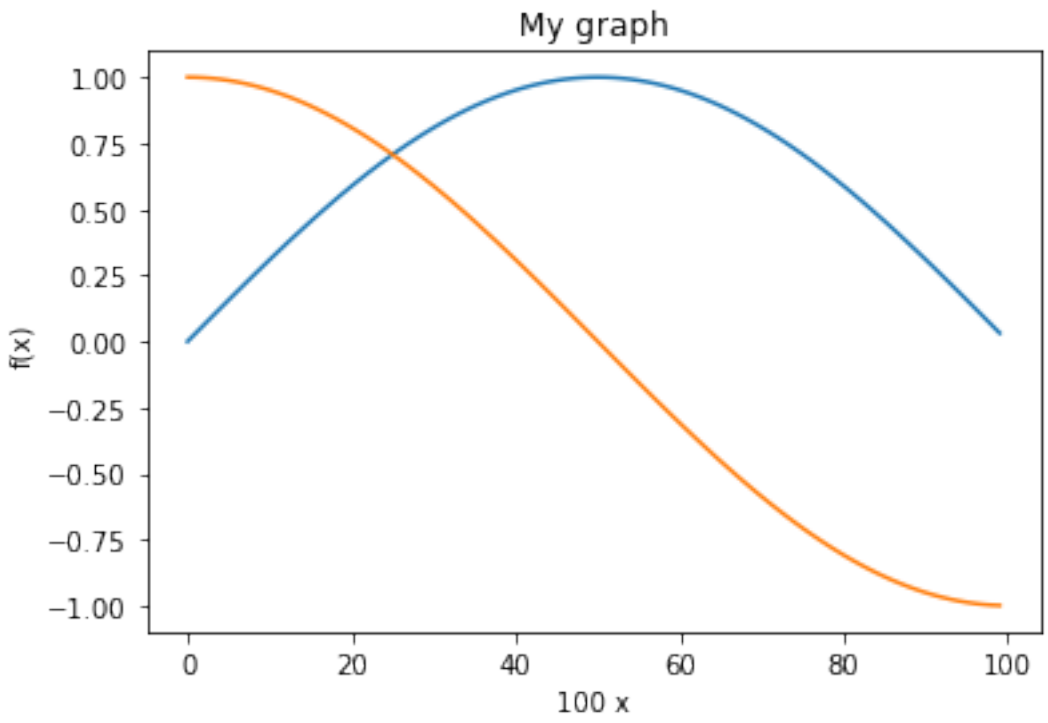
```
In [5]: sine_graph, sine_graph_axes = plt.subplots()
```



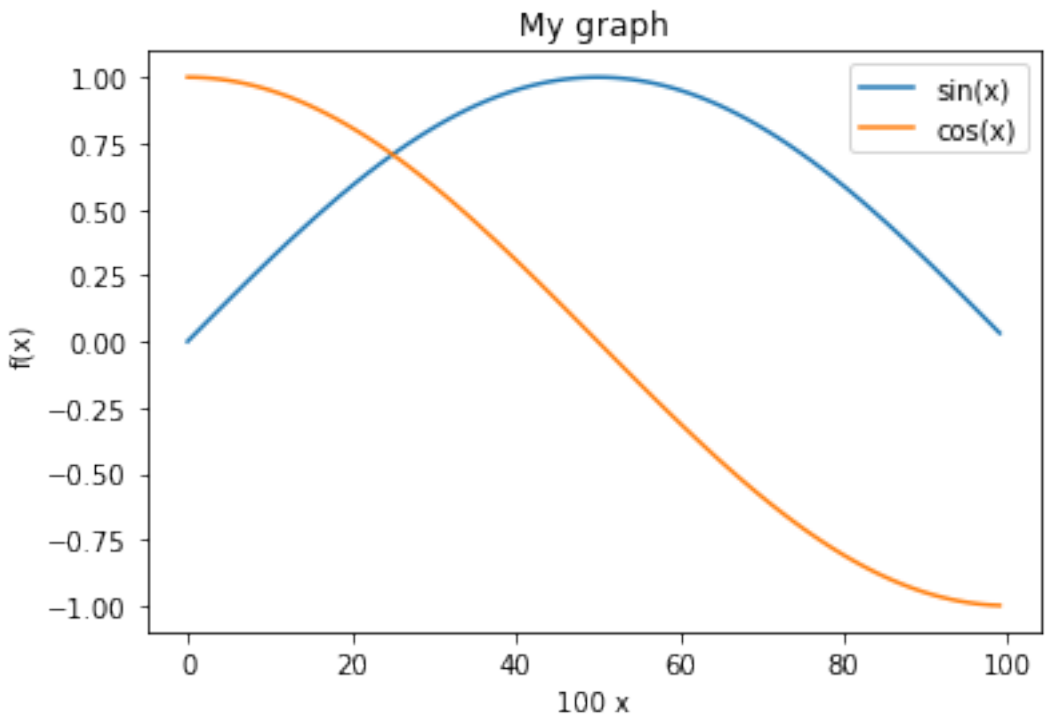

```
In [6]: sine_graph_axes.plot([sin(pi*x/100.0) for x in range(100)], label='sin(x)')
Out[6]: [<matplotlib.lines.Line2D at 0x11126fda0>]
In [7]: sine_graph_axes.set_title("My graph")
Out[7]: Text(0.5, 1.0, 'My graph')
In [8]: sine_graph_axes.set_ylabel("f(x)")
Out[8]: Text(3.200000000000003, 0.5, 'f(x)')
In [9]: sine_graph_axes.set_xlabel("100 x")
Out[9]: Text(0.5, 3.1999999999999993, '100 x')
In [10]: sine_graph
   Out[10]:
```



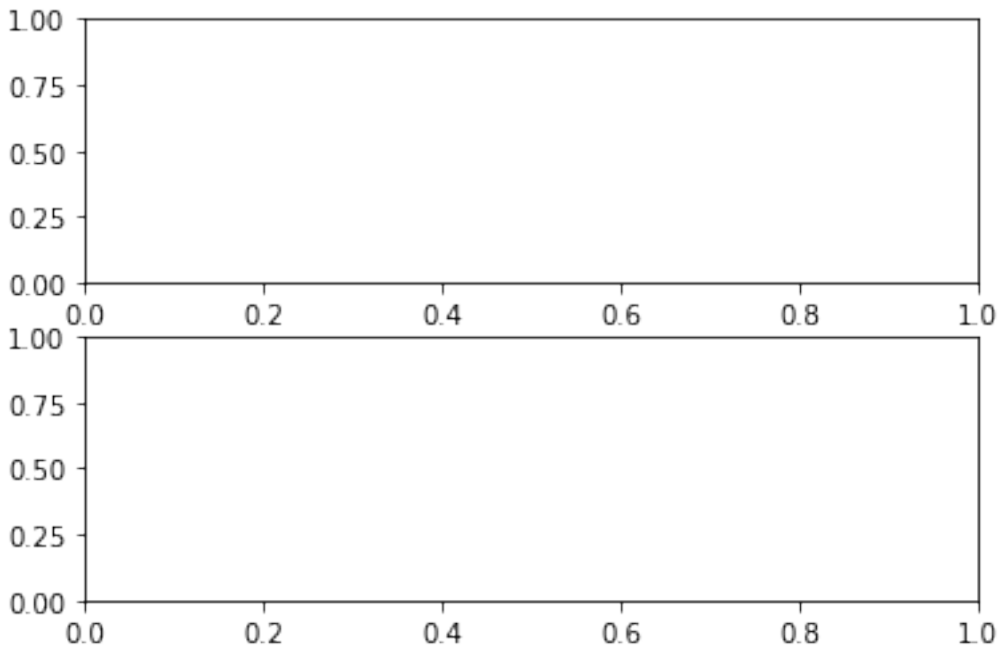

```
In [11]: sine_graph_axes.plot([cos(pi*x/100.0) for x in range(100)], label='cos(x)')

Out[11]: [<matplotlib.lines.Line2D at 0x111295f60>]

In [12]: sine_graph

   Out[12]:
```



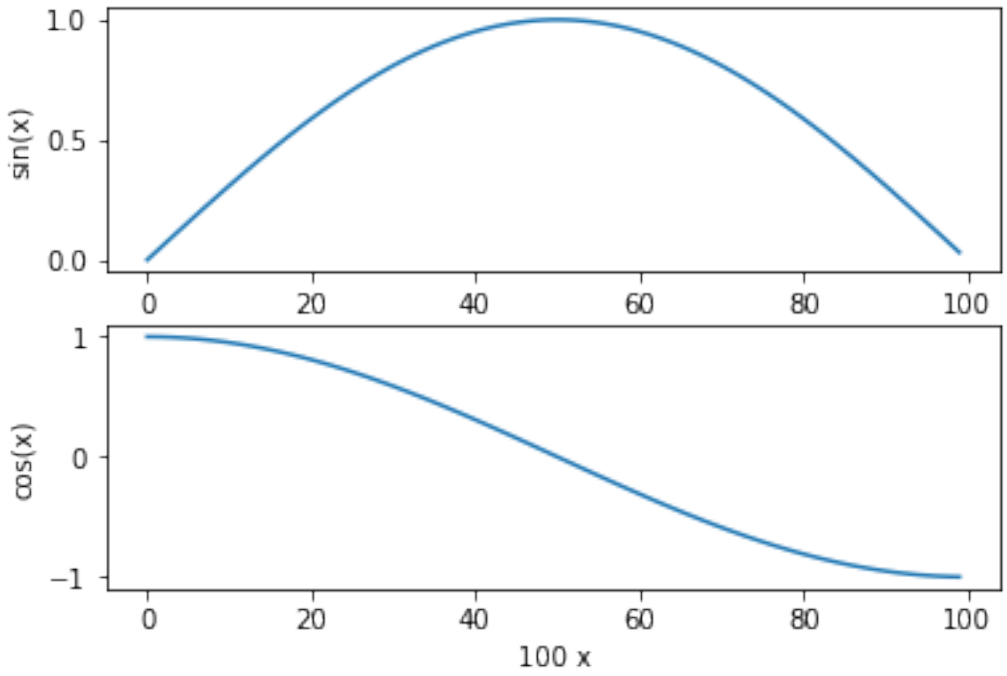

```
In [13]: sine_graph_axes.legend()

Out[13]: <matplotlib.legend.Legend at 0x10e4266a0>

In [14]: sine_graph

   Out[14]:
```



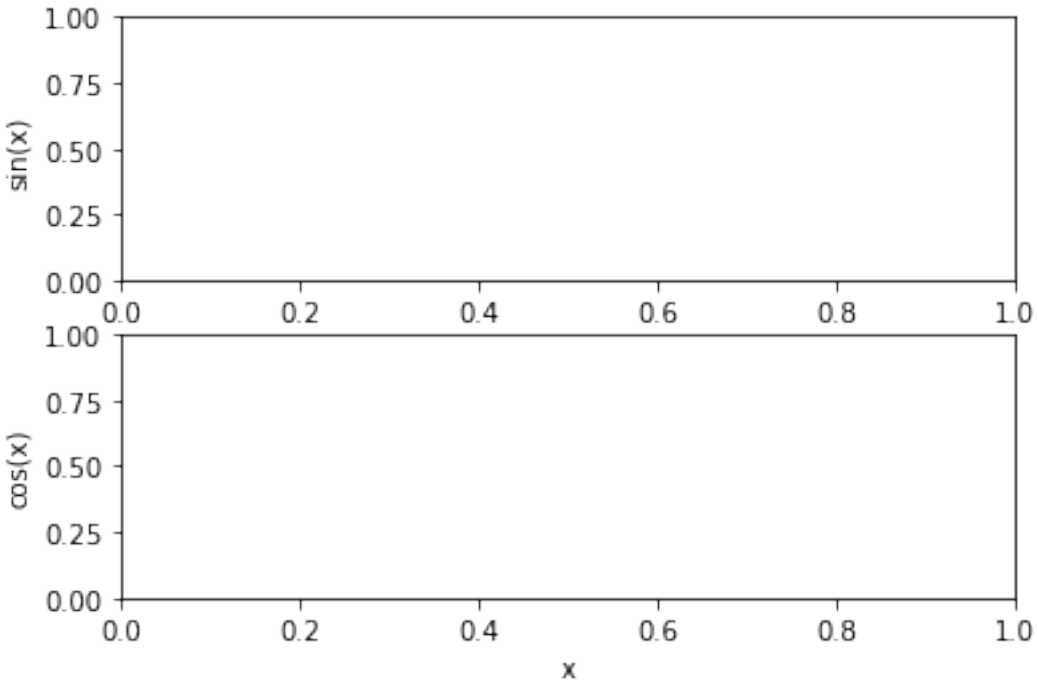

```
In [15]: sine_graph.savefig('my_graph.png')

In [16]: import IPython.display import Image # Get the notebook's own library for manipulating itself.
         Image(filename='my_graph.png')


           File "<ipython-input-16-1fc8b7d17d6b>", line 1
         import IPython.display import Image # Get the notebook's own library for manipulating itself.
                                    ^

     SyntaxError: invalid syntax


In [17]: double_graph = plt.figure()

<Figure size 432x288 with 0 Axes>


In [18]: sin_axes = double_graph.add_subplot(2, 1, 1) # 2 rows, 1 column, 1st subplot

In [19]: cos_axes = double_graph.add_subplot(2, 1, 2)

In [20]: double_graph

   Out[20]:
```




```
In [21]: sin_axes.plot([sin(pi*x/100.0) for x in range(100)])

Out[21]: [<matplotlib.lines.Line2D at 0x11150ff60>]

In [22]: sin_axes.set_ylabel("sin(x)")

Out[22]: Text(3.200000000000003, 0.5, 'sin(x)')

In [23]: cos_axes.plot([cos(pi*x/100.0) for x in range(100)])

Out[23]: [<matplotlib.lines.Line2D at 0x1115184a8>]

In [24]: cos_axes.set_ylabel("cos(x)")

Out[24]: Text(3.200000000000003, 0.5, 'cos(x)')

In [25]: cos_axes.set_xlabel("100 x")

Out[25]: Text(0.5, 3.200000000000003, '100 x')

In [26]: double_graph

   Out[26]:
```




```
In [27]: double_graph = plt.figure()
         sin_axes = double_graph.add_subplot(2, 1, 1)
         cos_axes = double_graph.add_subplot(2, 1, 2)
         cos_axes.set_ylabel("cos(x)")
         sin_axes.set_ylabel("sin(x)")
         cos_axes.set_xlabel("x")

Out[27]: Text(0.5, 0, 'x')
```

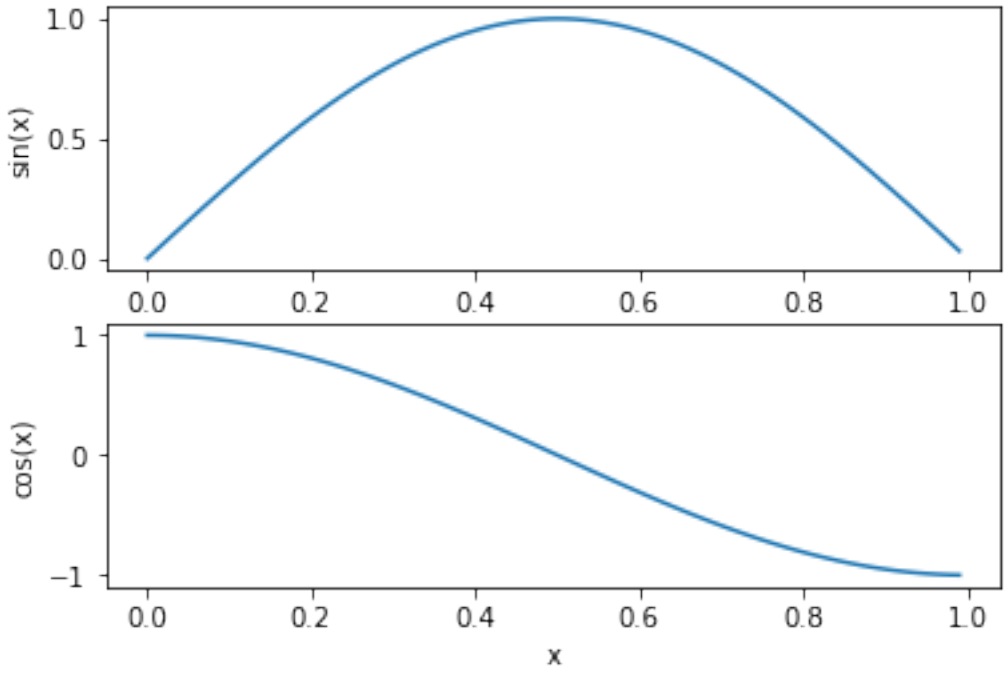



```
In [28]: sin_axes.plot([x/100.0 for x in range(100)], [sin(pi*x/100.0) for x in range(100)])
         cos_axes.plot([x/100.0 for x in range(100)], [cos(pi*x/100.0) for x in range(100)])
```

```
Out[28]: [<matplotlib.lines.Line2D at 0x1115cc588>]
```

```
In [29]: double_graph
```

Out[29]:



```
In [1]: x = [list(range(5)) for N in range(5)]

In [2]: x

Out[2]: [[0, 1, 2, 3, 4],
         [0, 1, 2, 3, 4],
         [0, 1, 2, 3, 4],
         [0, 1, 2, 3, 4],
         [0, 1, 2, 3, 4]]

In [3]: x[2][2]

Out[3]: 2

In [4]: x + 5


        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-4-9e8324a7b754> in <module>
    ----> 1 x + 5


        TypeError: can only concatenate list (not "int") to list


In [5]: [[elem + 5 for elem in row] for row in x]

Out[5]: [[5, 6, 7, 8, 9],
         [5, 6, 7, 8, 9],
         [5, 6, 7, 8, 9],
         [5, 6, 7, 8, 9],
         [5, 6, 7, 8, 9]]

In [6]: import numpy as np
        my_array = np.array(range(5))

In [7]: my_array

Out[7]: array([0, 1, 2, 3, 4])

In [8]: my_array[2]

Out[8]: 2

In [9]: for element in my_array:
            print("Hello" * element)


Hello
HelloHello
HelloHelloHello
HelloHelloHelloHello
```

```
In [10]: my_array.append(4)
```

```
---------------------------------------------------------------------------

AttributeError                            Traceback (most recent call last)

<ipython-input-10-b12177763178> in <module>
----> 1 my_array.append(4)


AttributeError: 'numpy.ndarray' object has no attribute 'append'
```

```
In [11]: my_array + 2

Out[11]: array([2, 3, 4, 5, 6])

In [12]: import numpy as np
         big_list = range(10000)
         big_array = np.arange(10000)

In [13]: %%timeit
         [x**2 for x in big_list]

3.14 ms ś 4.51 ţs per loop (mean ś std. dev. of 7 runs, 100 loops each)


In [14]: %%timeit
         big_array**2

5.46 ţs ś 50.8 ns per loop (mean ś std. dev. of 7 runs, 100000 loops each)


In [15]: x = np.arange(0, 10, 0.1)  # Start, stop, step size

In [16]: y = list(range(0, 10, 0.1))
```

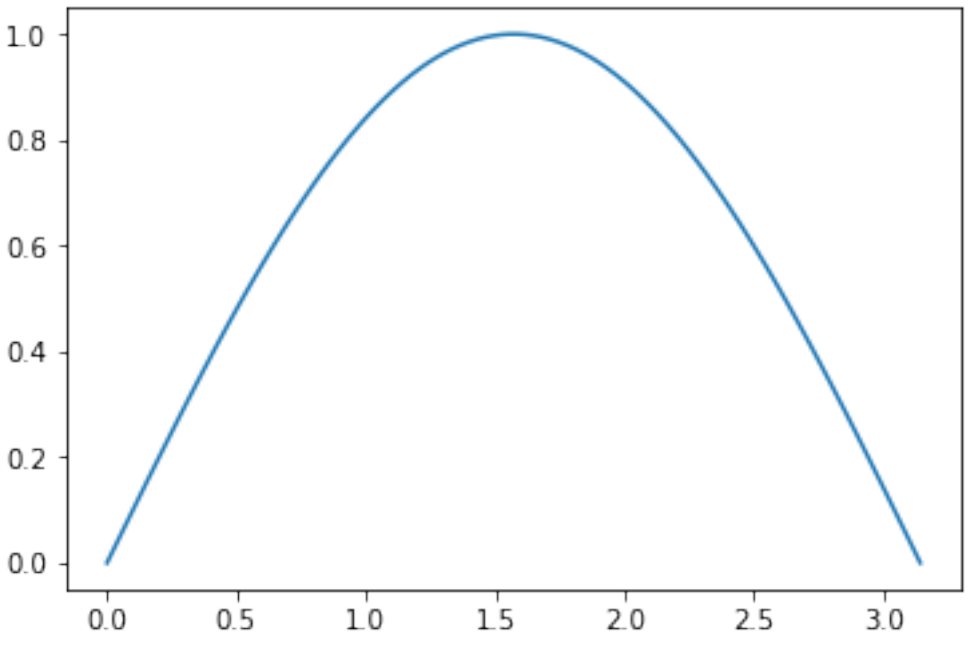
```
---------------------------------------------------------------------------

TypeError                                 Traceback (most recent call last)

<ipython-input-16-90c31a0aefc9> in <module>
----> 1 y = list(range(0, 10, 0.1))


TypeError: 'float' object cannot be interpreted as an integer
```

```
In [17]: import math
         values = np.linspace(0, math.pi, 100)  # Start, stop, number of steps

In [18]: values
```

```
Out[18]: array([0.        , 0.03173326, 0.06346652, 0.09519978, 0.12693304,
                0.1586663 , 0.19039955, 0.22213281, 0.25386607, 0.28559933,
                0.31733259, 0.34906585, 0.38079911, 0.41253237, 0.44426563,
                0.47599889, 0.50773215, 0.53946541, 0.57119866, 0.60293192,
                0.63466518, 0.66639844, 0.6981317 , 0.72986496, 0.76159822,
                0.79333148, 0.82506474, 0.856798  , 0.88853126, 0.92026451,
                0.95199777, 0.98373103, 1.01546429, 1.04719755, 1.07893081,
                1.11066407, 1.14239733, 1.17413059, 1.20586385, 1.23759711,
                1.26933037, 1.30106362, 1.33279688, 1.36453014, 1.3962634 ,
                1.42799666, 1.45972992, 1.49146318, 1.52319644, 1.5549297 ,
                1.58666296, 1.61839622, 1.65012947, 1.68186273, 1.71359599,
                1.74532925, 1.77706251, 1.80879577, 1.84052903, 1.87226229,
                1.90399555, 1.93572881, 1.96746207, 1.99919533, 2.03092858,
                2.06266184, 2.0943951 , 2.12612836, 2.15786162, 2.18959488,
                2.22132814, 2.2530614 , 2.28479466, 2.31652792, 2.34826118,
                2.37999443, 2.41172769, 2.44346095, 2.47519421, 2.50692747,
                2.53866073, 2.57039399, 2.60212725, 2.63386051, 2.66559377,
                2.69732703, 2.72906028, 2.76079354, 2.7925268 , 2.82426006,
                2.85599332, 2.88772658, 2.91945984, 2.9511931 , 2.98292636,
                3.01465962, 3.04639288, 3.07812614, 3.10985939, 3.14159265])
```

```
In [19]: %matplotlib inline

         from matplotlib import pyplot as plt
         plt.plot(values, np.sin(values))
```

```
Out[19]: [<matplotlib.lines.Line2D at 0x1190c3cc0>]
```




```
In [20]: np.zeros([3, 4, 2])  # 3 arrays with 4 rows and 2 columns each
```

```
Out[20]: array([[[0., 0.],
                 [0., 0.],
                 [0., 0.],
                 [0., 0.]],

                [[0., 0.],
                 [0., 0.],
                 [0., 0.],
                 [0., 0.]],

                [[0., 0.],
                 [0., 0.],
                 [0., 0.],
                 [0., 0.]]])

In [21]: x = np.array(range(40))
         x

Out[21]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
                17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
                34, 35, 36, 37, 38, 39])

In [22]: y = x.reshape([4, 5, 2])
         y

Out[22]: array([[[ 0,  1],
                 [ 2,  3],
                 [ 4,  5],
                 [ 6,  7],
                 [ 8,  9]],

                [[10, 11],
                 [12, 13],
                 [14, 15],
                 [16, 17],
                 [18, 19]],

                [[20, 21],
                 [22, 23],
                 [24, 25],
                 [26, 27],
                 [28, 29]],

                [[30, 31],
                 [32, 33],
                 [34, 35],
                 [36, 37],
                 [38, 39]]])

In [23]: y[3, 2, 1]

Out[23]: 35

In [24]: y[:, 2, 1]

Out[24]: array([ 5, 15, 25, 35])
```

```
In [25]: y[2:, :1, :]  # Last 2 axes, 1st row, all columns

Out[25]: array([[[20, 21]],

                [[30, 31]]])

In [26]: y.transpose()

Out[26]: array([[[ 0, 10, 20, 30],
                 [ 2, 12, 22, 32],
                 [ 4, 14, 24, 34],
                 [ 6, 16, 26, 36],
                 [ 8, 18, 28, 38]],

                [[ 1, 11, 21, 31],
                 [ 3, 13, 23, 33],
                 [ 5, 15, 25, 35],
                 [ 7, 17, 27, 37],
                 [ 9, 19, 29, 39]]])

In [27]: y.shape

Out[27]: (4, 5, 2)

In [28]: y.transpose().shape

Out[28]: (2, 5, 4)

In [29]: x = np.arange(12).reshape(4,3)
         x

Out[29]: array([[ 0,  1,  2],
                [ 3,  4,  5],
                [ 6,  7,  8],
                [ 9, 10, 11]])

In [30]: x.mean(1)  # Mean along the second axis, leaving the first.

Out[30]: array([ 1.,  4.,  7., 10.])

In [31]: x.mean(0)  # Mean along the first axis, leaving the second.

Out[31]: array([4.5, 5.5, 6.5])

In [32]: x.mean()  # mean of all axes

Out[32]: 5.5

In [33]: x = ['hello', 2, 3.4]

In [34]: type(x[2])

Out[34]: float

In [35]: type(x[1])

Out[35]: int

In [36]: np.array(x)
```

```
Out[36]: array(['hello', '2', '3.4'], dtype='<U5')

In [37]: y = np.array([2, 3.4])

In [38]: y

Out[38]: array([2. , 3.4])

In [39]: y.dtype

Out[39]: dtype('float64')

In [40]: type(y[0])

Out[40]: numpy.float64

In [41]: z = np.array([3, 4, 5])
         z

Out[41]: array([3, 4, 5])

In [42]: type(z[0])

Out[42]: numpy.int64

In [43]: x = [2, 3.4, 7.2, 0]

In [44]: int_array = np.array(x, dtype=int)

In [45]: float_array = np.array(x, dtype=float)

In [46]: int_array

Out[46]: array([2, 3, 7, 0])

In [47]: float_array

Out[47]: array([2. , 3.4, 7.2, 0. ])

In [48]: int_array.dtype

Out[48]: dtype('int64')

In [49]: float_array.dtype

Out[49]: dtype('float64')

In [50]: np.arange(5) * np.arange(5)

Out[50]: array([ 0,  1,  4,  9, 16])

In [51]: np.arange(5) * np.arange(6)

        ---------------------------------------------------------------------------

        ValueError                                Traceback (most recent call last)

        <ipython-input-51-d87da4b8a218> in <module>
    ----> 1 np.arange(5) * np.arange(6)


        ValueError: operands could not be broadcast together with shapes (5,) (6,)
```

```
In [52]: np.zeros([2,3]) * np.zeros([2,4])
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)

<ipython-input-52-b6b30bdbcb53> in <module>
----> 1 np.zeros([2,3]) * np.zeros([2,4])


ValueError: operands could not be broadcast together with shapes (2,3) (2,4)
```

```
In [53]: m1 = np.arange(100).reshape([10, 10])

In [54]: m2 = np.arange(100).reshape([10, 5, 2])

In [55]: m1 + m2
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)

<ipython-input-55-92db99ada483> in <module>
----> 1 m1 + m2


ValueError: operands could not be broadcast together with shapes (10,10) (10,5,2)
```

```
In [56]: np.ones([3, 3]) * np.ones([3, 3]) #ãNote elementwise multiply, *not* matrix multiply.

Out[56]: array([[1., 1., 1.],
                [1., 1., 1.],
                [1., 1., 1.]])

In [57]: col = np.arange(10).reshape([10, 1])
         col

Out[57]: array([[0],
                [1],
                [2],
                [3],
                [4],
                [5],
                [6],
                [7],
                [8],
                [9]])

In [58]: row = col.transpose()
         row

Out[58]: array([[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]])
```

```
In [59]: col.shape # "Column Vector"

Out[59]: (10, 1)

In [60]: row.shape # "Row Vector"

Out[60]: (1, 10)

In [61]: row + col

Out[61]: array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],
                [ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10],
                [ 2,  3,  4,  5,  6,  7,  8,  9, 10, 11],
                [ 3,  4,  5,  6,  7,  8,  9, 10, 11, 12],
                [ 4,  5,  6,  7,  8,  9, 10, 11, 12, 13],
                [ 5,  6,  7,  8,  9, 10, 11, 12, 13, 14],
                [ 6,  7,  8,  9, 10, 11, 12, 13, 14, 15],
                [ 7,  8,  9, 10, 11, 12, 13, 14, 15, 16],
                [ 8,  9, 10, 11, 12, 13, 14, 15, 16, 17],
                [ 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]])

In [62]: 10 * row + col

Out[62]: array([[ 0, 10, 20, 30, 40, 50, 60, 70, 80, 90],
                [ 1, 11, 21, 31, 41, 51, 61, 71, 81, 91],
                [ 2, 12, 22, 32, 42, 52, 62, 72, 82, 92],
                [ 3, 13, 23, 33, 43, 53, 63, 73, 83, 93],
                [ 4, 14, 24, 34, 44, 54, 64, 74, 84, 94],
                [ 5, 15, 25, 35, 45, 55, 65, 75, 85, 95],
                [ 6, 16, 26, 36, 46, 56, 66, 76, 86, 96],
                [ 7, 17, 27, 37, 47, 57, 67, 77, 87, 97],
                [ 8, 18, 28, 38, 48, 58, 68, 78, 88, 98],
                [ 9, 19, 29, 39, 49, 59, 69, 79, 89, 99]])

In [63]: import numpy as np
         x = np.arange(10).reshape(2, 5)
         y = np.arange(8).reshape(2, 2, 2)

In [64]: x

Out[64]: array([[0, 1, 2, 3, 4],
                [5, 6, 7, 8, 9]])

In [65]: y

Out[65]: array([[[0, 1],
                 [2, 3]],

                [[4, 5],
                 [6, 7]]])

In [66]: x[:, :, np.newaxis, np.newaxis].shape

Out[66]: (2, 5, 1, 1)

In [67]: y[:, np.newaxis, :, :].shape

Out[67]: (2, 1, 2, 2)
```

```
In [68]: res = x[:, :, np.newaxis, np.newaxis] * y[:, np.newaxis, :, :]

In [69]: res.shape

Out[69]: (2, 5, 2, 2)

In [70]: np.sum(res)

Out[70]: 830

In [71]: threebythree = np.arange(9).reshape(3, 3)
         threebythree

Out[71]: array([[0, 1, 2],
                [3, 4, 5],
                [6, 7, 8]])

In [72]: threebythree[:, np.newaxis, :]

Out[72]: array([[[0, 1, 2]],

                [[3, 4, 5]],

                [[6, 7, 8]]])

In [73]: a = np.arange(9).reshape(3, 3)
         a

Out[73]: array([[0, 1, 2],
                [3, 4, 5],
                [6, 7, 8]])

In [74]: b = np.arange(3, 12).reshape(3, 3)
         b

Out[74]: array([[ 3,  4,  5],
                [ 6,  7,  8],
                [ 9, 10, 11]])

In [75]: a * b

Out[75]: array([[ 0,  4, 10],
                [18, 28, 40],
                [54, 70, 88]])

In [76]: np.dot(a, b)

Out[76]: array([[ 24,  27,  30],
                [ 78,  90, 102],
                [132, 153, 174]])

In [77]: a[:, :, np.newaxis].shape

Out[77]: (3, 3, 1)

In [78]: b[np.newaxis, :, :].shape

Out[78]: (1, 3, 3)
```

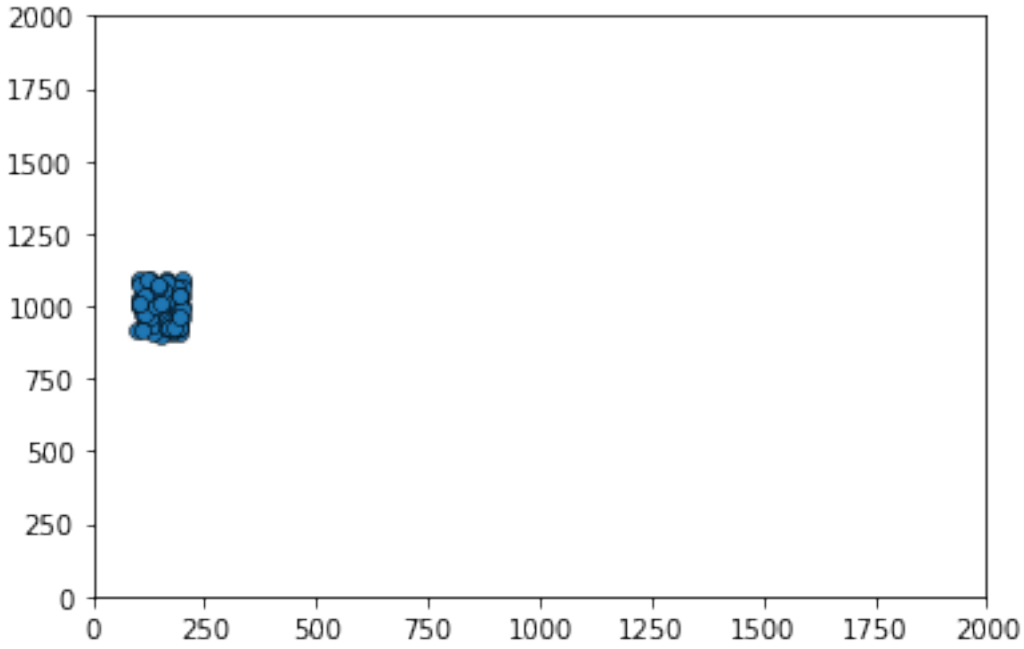
```
In [79]: a[:, :, np.newaxis] * b[np.newaxis, :, :]

Out[79]: array([[[ 0,  0,  0],
                 [ 6,  7,  8],
                 [18, 20, 22]],

                [[ 9, 12, 15],
                 [24, 28, 32],
                 [45, 50, 55]],

                [[18, 24, 30],
                 [42, 49, 56],
                 [72, 80, 88]]])

In [80]: (a[:, :, np.newaxis] * b[np.newaxis, :, :]).sum(1)

Out[80]: array([[ 24,  27,  30],
                [ 78,  90, 102],
                [132, 153, 174]])

In [81]: (a.reshape(3, 3, 1) * b.reshape(1, 3, 3)).sum(1)

Out[81]: array([[ 24,  27,  30],
                [ 78,  90, 102],
                [132, 153, 174]])

In [82]: a.reshape(3, 3, 1) * b.reshape(1, 3, 3)

Out[82]: array([[[ 0,  0,  0],
                 [ 6,  7,  8],
                 [18, 20, 22]],

                [[ 9, 12, 15],
                 [24, 28, 32],
                 [45, 50, 55]],

                [[18, 24, 30],
                 [42, 49, 56],
                 [72, 80, 88]]])

In [83]: x = np.arange(50).reshape([10, 5])

In [84]: record_x = x.view(dtype={'names': ["col1", "col2", "another", "more", "last"],
                                  'formats': [int]*5 })

In [85]: record_x

Out[85]: array([[( 0,  1,  2,  3,  4)],
                [( 5,  6,  7,  8,  9)],
                [(10, 11, 12, 13, 14)],
                [(15, 16, 17, 18, 19)],
                [(20, 21, 22, 23, 24)],
                [(25, 26, 27, 28, 29)],
                [(30, 31, 32, 33, 34)],
                [(35, 36, 37, 38, 39)],
                [(40, 41, 42, 43, 44)],
                [(45, 46, 47, 48, 49)]],
               dtype=[('col1', '<i8'), ('col2', '<i8'), ('another', '<i8'), ('more', '<i8'), ('last', '
```

```
In [86]: record_x['col1']

Out[86]: array([[ 0],
                [ 5],
                [10],
                [15],
                [20],
                [25],
                [30],
                [35],
                [40],
                [45]])

In [87]: x = np.zeros([3, 4])
         x

Out[87]: array([[0., 0., 0., 0.],
                [0., 0., 0., 0.],
                [0., 0., 0., 0.]])

In [88]: y = np.arange(-1, 2)[:, np.newaxis] * np.arange(-2, 2)[np.newaxis, :]
         y

Out[88]: array([[ 2,  1,  0, -1],
                [ 0,  0,  0,  0],
                [-2, -1,  0,  1]])

In [89]: iszero = x == y
         iszero

Out[89]: array([[False, False,  True, False],
                [ True,  True,  True,  True],
                [False, False,  True, False]])

In [90]: y[np.logical_not(iszero)]

Out[90]: array([ 2,  1, -1, -2, -1,  1])

In [91]: y[iszero] = 5

In [92]: y

Out[92]: array([[ 2,  1,  5, -1],
                [ 5,  5,  5,  5],
                [-2, -1,  5,  1]])

In [93]: x = np.arange(5)
         y = x[:]

In [94]: y[2] = 0
         x

Out[94]: array([0, 1, 0, 3, 4])

In [95]: x = list(range(5))
         y = x[:]

In [96]: y[2] = 0
         x
```

```
Out[96]: [0, 1, 2, 3, 4]

In [1]: import numpy as np

In [2]: boid_count = 10

In [3]: limits = np.array([2000, 2000])

In [4]: positions = np.random.rand(2, boid_count) * limits[:, np.newaxis]
        positions

Out[4]: array([[1767.69739642, 1125.42658472, 1224.80526146, 1054.36290103,
                1816.09198787, 1999.80475324,  633.43033947, 1561.32628068,
                1232.24448833,  217.95450273],
               [ 422.72470793,  158.97930444,  487.20449038, 1115.40329372,
                 712.36884967,  869.16727221, 1796.69464222, 1590.55888858,
                1935.41631713,  289.90022672]])

In [5]: positions.shape

Out[5]: (2, 10)

In [6]: limits[:, np.newaxis]

Out[6]: array([[2000],
               [2000]])

In [7]: limits[:, np.newaxis].shape

Out[7]: (2, 1)

In [8]: np.random.rand(2, boid_count).shape

Out[8]: (2, 10)

In [9]: def new_flock(count, lower_limits, upper_limits):
            width = upper_limits - lower_limits
            return (lower_limits[:, np.newaxis] + np.random.rand(2, count) * width[:, np.newaxis])

In [10]: velocities = new_flock(boid_count, np.array([0, -20]), np.array([10, 20]))
         velocities

Out[10]: array([[ 9.25706189,  8.54669982,  7.44079487,  1.2560114 ,  5.91591593,
                  9.49092133,  1.25720799,  9.3664265 ,  3.75519088,  7.51880438],
                [-4.54185971, -4.74857635,  9.05456151, 15.09208413, -7.07608772,
                 -0.25258491,  0.4395202 ,  4.06192627,  7.3295506 , -2.32494813]])

In [11]: positions += velocities

In [12]: from matplotlib import animation
         from matplotlib import pyplot as plt
         %matplotlib inline

In [13]: # create a simple plot
         # initial x position in [100, 200], initial y position in [900, 1100]
         # initial x velocity in [0, 10], initial y velocity in [-20, 20]
         positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
         velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
```

```
figure = plt.figure()
axes = plt.axes(xlim=(0, limits[0]), ylim=(0, limits[1]))
scatter = axes.scatter(positions[0, :], positions[1, :],
                       marker='o', edgecolor='k', lw=0.5)
scatter
```

Out[13]: <matplotlib.collections.PathCollection at 0x11b439d30>



In [14]:
```
def update_boids(positions, velocities):
    positions += velocities


def animate(frame):
    update_boids(positions, velocities)
    scatter.set_offsets(positions.transpose())
```

In [15]:
```
anim = animation.FuncAnimation(figure, animate,
                               frames=50, interval=50)
```

In [16]:
```
positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
anim.save('boids_1.mp4')
```

```
MovieWriter ffmpeg unavailable. Trying to use pillow instead.
```

```
    KeyError                                  Traceback (most recent call last)

    /usr/local/lib/python3.7/site-packages/PIL/Image.py in save(self, fp, format, **params)
   1973             try:
-> 1974                 format = EXTENSION[ext]
   1975             except KeyError:


    KeyError: '.mp4'


During handling of the above exception, another exception occurred:


    ValueError                                Traceback (most recent call last)

    <ipython-input-16-b7c0923dbe9c> in <module>
      1 positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
      2 velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
----> 3 anim.save('boids_1.mp4')


    /usr/local/lib/python3.7/site-packages/matplotlib/animation.py in save(self, filename, writer, i
   1172                         # TODO: See if turning off blit is really necessary
   1173                         anim._draw_next_frame(d, blit=False)
-> 1174                     writer.grab_frame(**savefig_kwargs)
   1175
   1176         # Reconnect signal for first draw if necessary


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/contextl
    117         if type is None:
    118             try:
--> 119                 next(self.gen)
    120             except StopIteration:
    121                 return False


    /usr/local/lib/python3.7/site-packages/matplotlib/animation.py in saving(self, fig, outfile, dpi
    230             yield self
    231         finally:
--> 232             self.finish()
    233
    234


    /usr/local/lib/python3.7/site-packages/matplotlib/animation.py in finish(self)
    581         self._frames[0].save(
    582             self._outfile, save_all=True, append_images=self._frames[1:],
--> 583             duration=int(1000 / self.fps))
    584
    585
```

```
/usr/local/lib/python3.7/site-packages/PIL/Image.py in save(self, fp, format, **params)
   1974                 format = EXTENSION[ext]
   1975             except KeyError:
-> 1976                 raise ValueError('unknown file extension: {}'.format(ext))
   1977
   1978         if format.upper() not in SAVE:


ValueError: unknown file extension: .mp4
```

```
In [17]: from IPython.display import HTML
         HTML(anim.to_jshtml())
```

```
Out[17]: <IPython.core.display.HTML object>
```

```
In [18]: positions = new_flock(4, np.array([100, 900]), np.array([200, 1100]))
         velocities = new_flock(4, np.array([0, -20]), np.array([10, 20]))
```

```
In [19]: positions
```

```
Out[19]: array([[ 143.0038255 ,  114.65877983,  171.82734626,  136.3336279 ],
                [ 946.20816007, 1003.31686891, 1070.92129149,  953.15017427]])
```

```
In [20]: velocities
```

```
Out[20]: array([[  6.12181951,   0.9793209 ,   3.44871544,   1.5335275 ],
                [  4.12085997, -18.72708909,   8.37964394,  14.99614472]])
```

```
In [21]: middle = np.mean(positions, 1)
         middle
```

```
Out[21]: array([141.45589487, 993.39912368])
```

```
In [22]: direction_to_middle = positions - middle[:, np.newaxis]
         direction_to_middle
```

```
Out[22]: array([[  1.54793063, -26.79711504,  30.37145139,  -5.12226698],
                [-47.19096362,   9.91774523,  77.5221678 , -40.24894941]])
```

```
In [23]: move_to_middle_strength = 0.01
         velocities = velocities - direction_to_middle * move_to_middle_strength
```

```
In [24]: def update_boids(positions, velocities):
             move_to_middle_strength = 0.01
             middle = np.mean(positions, 1)
             direction_to_middle = positions - middle[:, np.newaxis]
             velocities -= direction_to_middle * move_to_middle_strength
             positions += velocities
```

```
In [25]: def animate(frame):
             update_boids(positions, velocities)
             scatter.set_offsets(positions.transpose())
```

```
In [26]: anim = animation.FuncAnimation(figure, animate,
                                        frames=50, interval=50)
```

```
In [27]: positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
         velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
         HTML(anim.to_jshtml())

Out[27]: <IPython.core.display.HTML object>

In [28]: positions = new_flock(4, np.array([100, 900]), np.array([200, 1100]))
         velocities = new_flock(4, np.array([0, -20]), np.array([10, 20]))

In [29]: xpos = positions[0, :]

In [30]: xsep_matrix = xpos[:, np.newaxis] - xpos[np.newaxis, :]

In [31]: xsep_matrix.shape

Out[31]: (4, 4)

In [32]: xsep_matrix

Out[32]: array([[  0.        ,  16.19937413,  22.65004241, -17.98343261],
                [-16.19937413,   0.        ,   6.45066828, -34.18280673],
                [-22.65004241,  -6.45066828,   0.        , -40.63347502],
                [ 17.98343261,  34.18280673,  40.63347502,   0.        ]])

In [33]: separations = positions[:, np.newaxis, :] - positions[:, :, np.newaxis]

In [34]: separations.shape

Out[34]: (2, 4, 4)

In [35]: squared_displacements = separations * separations

In [36]: square_distances = np.sum(squared_displacements, 0)

In [37]: square_distances

Out[37]: array([[    0.        ,   262.62070779, 34894.64142882,   324.91858087],
                [  262.62070779,     0.        , 34257.17374299,  1171.28351522],
                [34894.64142882, 34257.17374299,     0.        , 36490.62730797],
                [  324.91858087,  1171.28351522, 36490.62730797,     0.        ]])

In [38]: alert_distance = 2000
         close_birds = square_distances < alert_distance
         close_birds

Out[38]: array([[ True,  True, False,  True],
                [ True,  True, False,  True],
                [False, False,  True, False],
                [ True,  True, False,  True]])

In [39]: separations_if_close = np.copy(separations)
         far_away = np.logical_not(close_birds)

In [40]: separations_if_close[0, :, :][far_away] = 0
         separations_if_close[1, :, :][far_away] = 0
         separations_if_close
```

```
Out[40]: array([[[  0.        , -16.19937413,   0.        ,  17.98343261],
                 [ 16.19937413,   0.        ,   0.        ,  34.18280673],
                 [  0.        ,   0.        ,   0.        ,   0.        ],
                 [-17.98343261, -34.18280673,   0.        ,   0.        ]],

                [[  0.        ,   0.44831425,   0.        ,  -1.23074471],
                 [ -0.44831425,   0.        ,   0.        ,  -1.67905896],
                 [  0.        ,   0.        ,   0.        ,   0.        ],
                 [  1.23074471,   1.67905896,   0.        ,   0.        ]]])

In [41]: np.sum(separations_if_close, 2)

Out[41]: array([[  1.78405848,  50.38218086,   0.        , -52.16623934],
                [ -0.78243046,  -2.12737321,   0.        ,   2.90980367]])

In [42]: velocities = velocities + np.sum(separations_if_close, 2)

In [43]: def update_boids(positions, velocities):
             move_to_middle_strength = 0.01
             middle = np.mean(positions, 1)
             direction_to_middle = positions - middle[:, np.newaxis]
             velocities -= direction_to_middle * move_to_middle_strength

             separations = positions[:, np.newaxis, :] - positions[:, :, np.newaxis]
             squared_displacements = separations * separations
             square_distances = np.sum(squared_displacements, 0)
             alert_distance = 100
             far_away = square_distances > alert_distance
             separations_if_close = np.copy(separations)
             separations_if_close[0, :, :][far_away] = 0
             separations_if_close[1, :, :][far_away] = 0
             velocities += np.sum(separations_if_close, 1)

             positions += velocities

In [44]: def animate(frame):
             update_boids(positions, velocities)
             scatter.set_offsets(positions.transpose())


         anim = animation.FuncAnimation(figure, animate,
                                        frames=50, interval=50)

         positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
         velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
         HTML(anim.to_jshtml())

Out[44]: <IPython.core.display.HTML object>

In [45]: def update_boids(positions, velocities):
             move_to_middle_strength = 0.01
             middle = np.mean(positions, 1)
             direction_to_middle = positions - middle[:, np.newaxis]
             velocities -= direction_to_middle * move_to_middle_strength

             separations = positions[:, np.newaxis, :] - positions[:, :, np.newaxis]
```

```
        squared_displacements = separations * separations
        square_distances = np.sum(squared_displacements, 0)
        alert_distance = 100
        far_away = square_distances > alert_distance
        separations_if_close = np.copy(separations)
        separations_if_close[0, :, :][far_away] = 0
        separations_if_close[1, :, :][far_away] = 0
        velocities += np.sum(separations_if_close, 1)

        velocity_differences = velocities[:, np.newaxis, :] - velocities[:, :, np.newaxis]
        formation_flying_distance = 10000
        formation_flying_strength = 0.125
        very_far = square_distances > formation_flying_distance
        velocity_differences_if_close = np.copy(velocity_differences)
        velocity_differences_if_close[0, :, :][very_far] = 0
        velocity_differences_if_close[1, :, :][very_far] = 0
        velocities -= np.mean(velocity_differences_if_close, 1) * formation_flying_strength

        positions += velocities
```

In [46]:
```
def animate(frame):
    update_boids(positions, velocities)
    scatter.set_offsets(positions.transpose())


anim = animation.FuncAnimation(figure, animate,
                               frames=200, interval=50)


positions = new_flock(100, np.array([100, 900]), np.array([200, 1100]))
velocities = new_flock(100, np.array([0, -20]), np.array([10, 20]))
HTML(anim.to_jshtml())
```

Out[46]: <IPython.core.display.HTML object>

In [1]:
```
%%bash
mkdir -p greengraph  # Create the folder for the module (on mac or linux)
```

In [2]:
```
%%writefile greengraph/graph.py
import numpy as np
import geopy
from .map import Map


class Greengraph(object):
    def __init__(self, start, end):
        self.start = start
        self.end = end
        self.geocoder = geopy.geocoders.Yandex(lang="en_US")

    def geolocate(self, place):
        return self.geocoder.geocode(place, exactly_one=False)[0][1]

    def location_sequence(self, start, end, steps):
        lats = np.linspace(start[0], end[0], steps)
```

```
            longs = np.linspace(start[1], end[1], steps)
            return np.vstack([lats, longs]).transpose()

        def green_between(self, steps):
            return [Map(*location).count_green()
                    for location in self.location_sequence(
                        self.geolocate(self.start),
                        self.geolocate(self.end),
                        steps)]
```

```
Overwriting greengraph/graph.py
```

In [3]:
```
%%writefile greengraph/map.py

import numpy as np
from io import BytesIO
import imageio as img
import requests

class Map(object):
    def __init__(self, lat, long, satellite=True, zoom=10,
                 size=(400, 400), sensor=False):
        base = "https://static-maps.yandex.ru/1.x/?"

        params = dict(
            z=zoom,
            size=str(size[0]) + "," + str(size[1]),
            ll=str(long) + "," + str(lat),
            l="sat" if satellite else "map",
            lang="en_US"
        )

        self.image = requests.get(
            base, params=params).content  # Fetch our PNG image data
        content = BytesIO(self.image)
        self.pixels = img.imread(content) # Parse our PNG image as a numpy array

    def green(self, threshold):
        # Use NumPy to build an element-by-element logical array
        greener_than_red = self.pixels[:, :, 1] > threshold * self.pixels[:, :, 0]
        greener_than_blue = self.pixels[:, :, 1] > threshold * self.pixels[:, :, 2]
        green = np.logical_and(greener_than_red, greener_than_blue)
        return green

    def count_green(self, threshold=1.1):
        return np.sum(self.green(threshold))

    def show_green(data, threshold=1.1):
        green = self.green(threshold)
        out = green[:, :, np.newaxis] * array([0, 1, 0])[np.newaxis, np.newaxis, :]
        buffer = BytesIO()
        result = img.imwrite(buffer, out, format='png')
        return buffer.getvalue()
```

```
Overwriting greengraph/map.py
```

```
In [4]: %%writefile greengraph/__init__.py
        from .graph import Greengraph
```

```
Overwriting greengraph/__init__.py
```

```
In [5]: from matplotlib import pyplot as plt
        from greengraph import Greengraph
        %matplotlib inline

        mygraph = Greengraph('New York', 'Chicago')
        data = mygraph.green_between(20)
```

```
        ---------------------------------------------------------------------------

        ModuleNotFoundError                       Traceback (most recent call last)

        <ipython-input-5-a69e6d6508d4> in <module>
          1 from matplotlib import pyplot as plt
    ----> 2 from greengraph import Greengraph
          3 get_ipython().run_line_magic('matplotlib', 'inline')
          4
          5 mygraph = Greengraph('New York', 'Chicago')


        ~/Projects/rsd-engineeringcourse/ch01data/greengraph/__init__.py in <module>
    ----> 1 from .graph import Greengraph


        ~/Projects/rsd-engineeringcourse/ch01data/greengraph/graph.py in <module>
          1 import numpy as np
          2 import geopy
    ----> 3 from .map import Map
          4
          5


        ~/Projects/rsd-engineeringcourse/ch01data/greengraph/map.py in <module>
          2 import numpy as np
          3 from io import BytesIO
    ----> 4 import imageio as img
          5 import requests
          6


        ModuleNotFoundError: No module named 'imageio'
```

```
In [6]: plt.plot(data)
```

```
        ---------------------------------------------------------------------------
```

```
        NameError                                 Traceback (most recent call last)

        <ipython-input-6-727d88478626> in <module>
    ----> 1 plt.plot(data)


        NameError: name 'data' is not defined
```

```
In [1]: %%bash
        echo some output
```

```
some output
```

```
In [2]: %%writefile somefile.md
        Some content here
```

```
Writing somefile.md
```

```
In [3]: %%bash
        rm -rf learning_git/git_example # Just in case it's left over from a previous class; you won't
        mkdir -p learning_git/git_example
        cd learning_git/git_example
```

```
In [4]: import os
        top_dir = os.getcwd()
        top_dir
```

```
Out[4]: '/Users/edaub/Projects/rsd-engineeringcourse/ch02git'
```

```
In [5]: git_dir = os.path.join(top_dir, 'learning_git')
        git_dir
```

```
Out[5]: '/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git'
```

```
In [6]: working_dir=os.path.join(git_dir, 'git_example')
```

```
In [7]: os.chdir(working_dir)
```

```
In [8]: %%bash
        git config --global user.name "Giovanni1085"
        git config --global user.email "gcolavizza@turing.ac.uk"
```

```
In [9]: %%bash
        pwd # Note where we are standing-- MAKE SURE YOU INITIALISE THE RIGHT FOLDER
        git init
```

```
/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/git_example
Initialized empty Git repository in /Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/gi
```

```
In [10]: %%bash
         ls
```

```
In [11]: %%bash
         git status
```

```
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
        working_dir
```

```
Out[1]: '/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/git_example'
```

```
In [2]: %%writefile index.md
        Mountains in the UK
        ===================
        England is not very mountainous.
        But has some tall hills, and maybe a mountain or two depending on your definition.
```

```
Writing index.md
```

```
In [3]: cat index.md
```

```
Mountains in the UK
===================
England is not very mountainous.
But has some tall hills, and maybe a mountain or two depending on your definition.
```

```
In [4]: %%bash
        git add index.md
```

```
In [5]: %%bash
        git commit -m "First commit of discourse on UK topography"
```

```
[master (root-commit) 7a758a4] First commit of discourse on UK topography
 1 file changed, 4 insertions(+)
 create mode 100644 index.md
```

```
In [6]: %%bash
        git config --global core.editor vim
```

```
In [7]: %%bash
        git config --get core.editor
```

```
vim
```

```
In [8]: %%bash
        git log
```

```
commit 7a758a46b3d0b25de036e02f28ae439afc1b619c
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:19:41 2019 +0000

    First commit of discourse on UK topography
```

```
In [9]: %%bash
        git status
```

```
On branch master
nothing to commit, working tree clean
```

```
In [10]: %%writefile index.md
         Mountains in the UK
         ====================
         England is not very mountainous.
         But has some tall hills, and maybe a mountain or two depending on your definition.

         Mount Fictional, in Barsetshire, U.K. is the tallest mountain in the world.
```

```
Overwriting index.md
```

```
In [11]: cat index.md
```

```
Mountains in the UK
====================
England is not very mountainous.
But has some tall hills, and maybe a mountain or two depending on your definition.

Mount Fictional, in Barsetshire, U.K. is the tallest mountain in the world.
```

```
In [12]: %%bash
         git status
```

```
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.md

no changes added to commit (use "git add" and/or "git commit -a")
```

```
In [13]: %%bash
         git diff
```

```
diff --git a/index.md b/index.md
index a1f85df..3a2f7b0 100644
--- a/index.md
+++ b/index.md
@@ -2,3 +2,5 @@ Mountains in the UK
```

```
 ==================
 England is not very mountainous.
 But has some tall hills, and maybe a mountain or two depending on your definition.
+
+Mount Fictional, in Barsetshire, U.K. is the tallest mountain in the world.
```

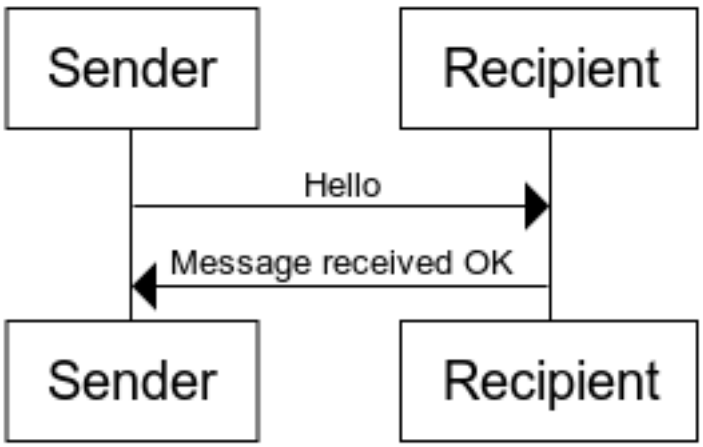
```
In [14]: %%bash
         git add --update
```

```
In [15]: %%writefile wsd.py
         import requests
         import re
         import IPython

         def wsd(code):
             response = requests.post("http://www.websequencediagrams.com/index.php", data={
                     'message': code,
                     'apiVersion': 1,
                 })
             expr = re.compile("(\?(img|pdf|png|svg)=[a-zA-Z0-9]+)")
             m = expr.search(response.text)
             if m == None:
                 print("Invalid response from server.")
                 return False

             image=requests.get("http://www.websequencediagrams.com/" + m.group(0))
             return IPython.core.display.Image(image.content)
```

```
Writing wsd.py
```

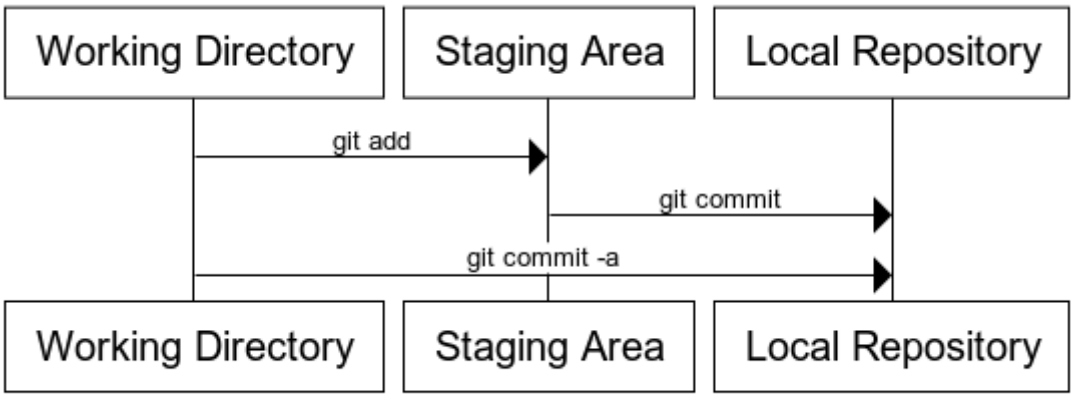
```
In [16]: from wsd import wsd
         %matplotlib inline
         wsd("Sender->Recipient: Hello\n Recipient->Sender: Message received OK")
```

Out[16]:



In [17]:
```
message="""
        Working Directory -> Staging Area : git add
        Staging Area -> Local Repository : git commit
        Working Directory -> Local Repository : git commit -a
        """
        wsd(message)
```

Out[17]:



In [18]:
```
%%bash
git status
```

```
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   index.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        __pycache__/
        wsd.py
```

In [19]:
```
%%bash
git commit -m "Add a lie about a mountain"
```

```
[master 4ad1c69] Add a lie about a mountain
 1 file changed, 2 insertions(+)
```

In [20]:
```
%%bash
git log
```

```
commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:19:43 2019 +0000

    Add a lie about a mountain

commit 7a758a46b3d0b25de036e02f28ae439afc1b619c
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:19:41 2019 +0000

    First commit of discourse on UK topography
```

```
In [21]: %%writefile index.md
         Mountains and Hills in the UK
         ==================
         England is not very mountainous.
         But has some tall hills, and maybe a mountain or two depending on your definition.

         Mount Fictional, in Barsetshire, U.K. is the tallest mountain in the world.
```

```
Overwriting index.md
```

```
In [22]: cat index.md
```

```
Mountains and Hills in the UK
==================
England is not very mountainous.
But has some tall hills, and maybe a mountain or two depending on your definition.

Mount Fictional, in Barsetshire, U.K. is the tallest mountain in the world.
```

```
In [23]: %%bash
         git commit -am "Change title"
```

```
[master 84d66d0] Change title
 1 file changed, 1 insertion(+), 1 deletion(-)
```

```
In [24]: %%bash
         git log | head
```

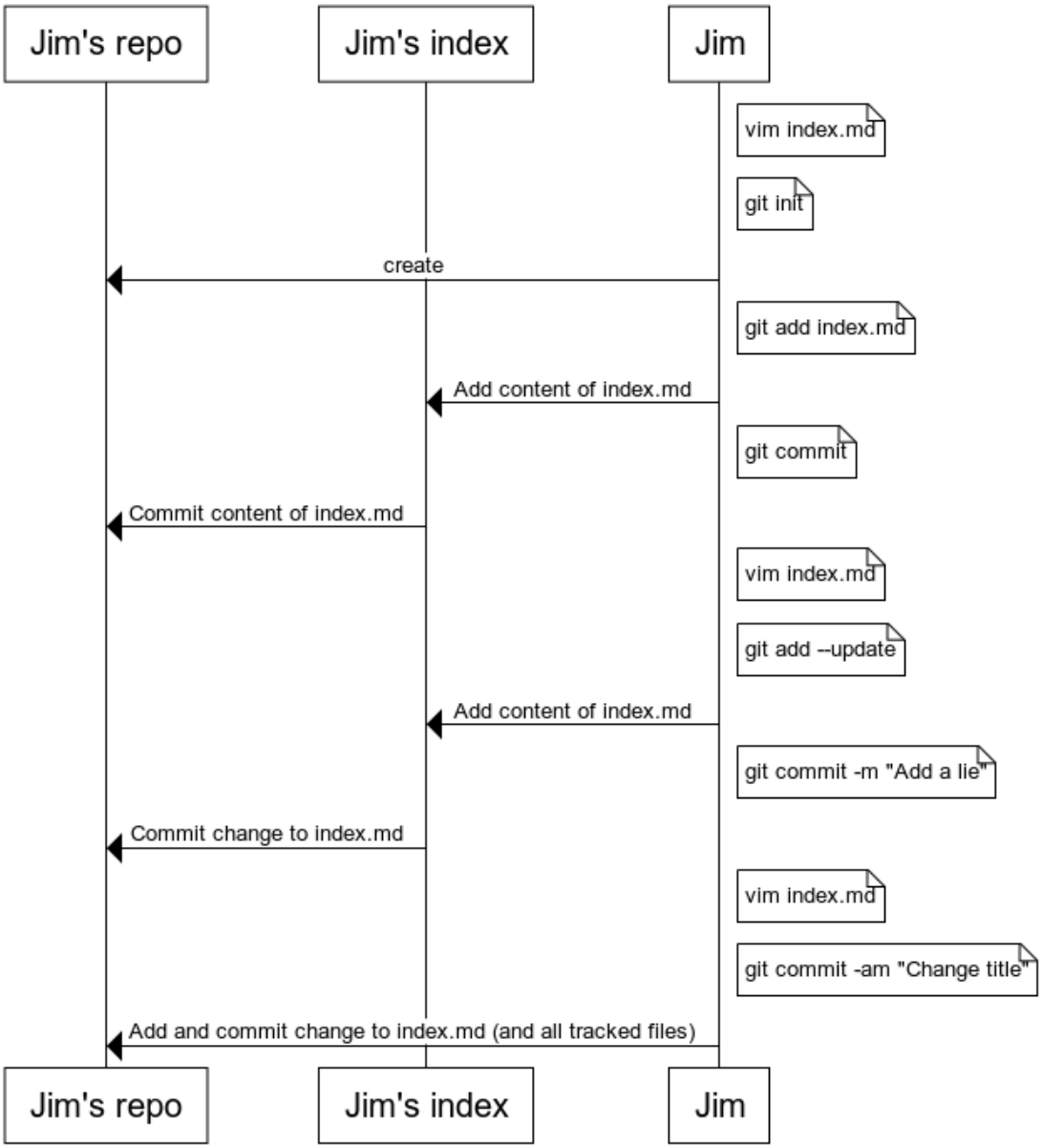
```
commit 84d66d0c34a8d697f5134b291943789f2b9ee32f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:19:43 2019 +0000

    Change title

commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:19:43 2019 +0000
```

```
In [25]: %%bash
         git log --oneline
```

```
84d66d0 Change title
4ad1c69 Add a lie about a mountain
7a758a4 First commit of discourse on UK topography
```

```
In [26]: message="""
         participant "Jim's repo" as R
         participant "Jim's index" as I
         participant Jim as J

         note right of J: vim index.md

         note right of J: git init
         J->R: create

         note right of J: git add index.md

         J->I: Add content of index.md

         note right of J: git commit
         I->R: Commit content of index.md

         note right of J:  vim index.md

         note right of J: git add --update
         J->I: Add content of index.md
         note right of J: git commit -m "Add a lie"
         I->R: Commit change to index.md

         note right of J:  vim index.md
         note right of J: git commit -am "Change title"
         J->R: Add and commit change to index.md (and all tracked files)
         """
         wsd(message)
```

```
Out[26]:
```




```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
        working_dir

Out[1]: '/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/git_example'

In [2]: %%bash
        git revert HEAD^
```

```
[master 230c644] Revert "Add a lie about a mountain"
 Date: Mon Nov 4 11:19:48 2019 +0000
 1 file changed, 2 deletions(-)
```

```
In [3]: %%bash
        git log --date=short
```

```
commit 230c64495a725c17d5d55f9bf7335cfb4945ad2f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Revert "Add a lie about a mountain"

    This reverts commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548.

commit 84d66d0c34a8d697f5134b291943789f2b9ee32f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Change title

commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Add a lie about a mountain

commit 7a758a46b3d0b25de036e02f28ae439afc1b619c
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    First commit of discourse on UK topography
```

```
In [4]: %%writefile index.md
        Mountains and Hills in the UK
        ==================
        Engerland is not very mountainous.
        But has some tall hills, and maybe a
        mountain or two depending on your definition.
```

```
Overwriting index.md
```

```
In [5]: %%bash
        cat index.md
```

```
Mountains and Hills in the UK
==================
Engerland is not very mountainous.
But has some tall hills, and maybe a
mountain or two depending on your definition.
```

```
In [6]: %%bash
        git diff
```

```
diff --git a/index.md b/index.md
index dd5cf9c..4801c98 100644
--- a/index.md
+++ b/index.md
@@ -1,4 +1,5 @@
 Mountains and Hills in the UK
 ==========
-England is not very mountainous.
-But has some tall hills, and maybe a mountain or two depending on your definition.
+Engerland is not very mountainous.
+But has some tall hills, and maybe a
+mountain or two depending on your definition.
```

```
In [7]: %%bash
        git commit -am "Add a silly spelling"
```

```
[master d77e0da] Add a silly spelling
 1 file changed, 3 insertions(+), 2 deletions(-)
```

```
In [8]: %%bash
        git log --date=short
```

```
commit d77e0dae522c10e48676b80520d372469db08d2f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Add a silly spelling

commit 230c64495a725c17d5d55f9bf7335cfb4945ad2f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Revert "Add a lie about a mountain"

    This reverts commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548.

commit 84d66d0c34a8d697f5134b291943789f2b9ee32f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Change title

commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Add a lie about a mountain

commit 7a758a46b3d0b25de036e02f28ae439afc1b619c
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    First commit of discourse on UK topography
```

```
In [9]: %%bash
        git reset HEAD^
```

```
Unstaged changes after reset:
M       index.md
```

```
In [10]: %%bash
         git log --date=short
```

```
commit 230c64495a725c17d5d55f9bf7335cfb4945ad2f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Revert "Add a lie about a mountain"

    This reverts commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548.

commit 84d66d0c34a8d697f5134b291943789f2b9ee32f
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Change title

commit 4ad1c692c02a1f39a09576c46f7cf0323b34e548
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    Add a lie about a mountain

commit 7a758a46b3d0b25de036e02f28ae439afc1b619c
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   2019-11-04

    First commit of discourse on UK topography
```

```
In [11]: %%bash
         cat index.md
```

```
Mountains and Hills in the UK
==================
Engerland is not very mountainous.
But has some tall hills, and maybe a
mountain or two depending on your definition.
```

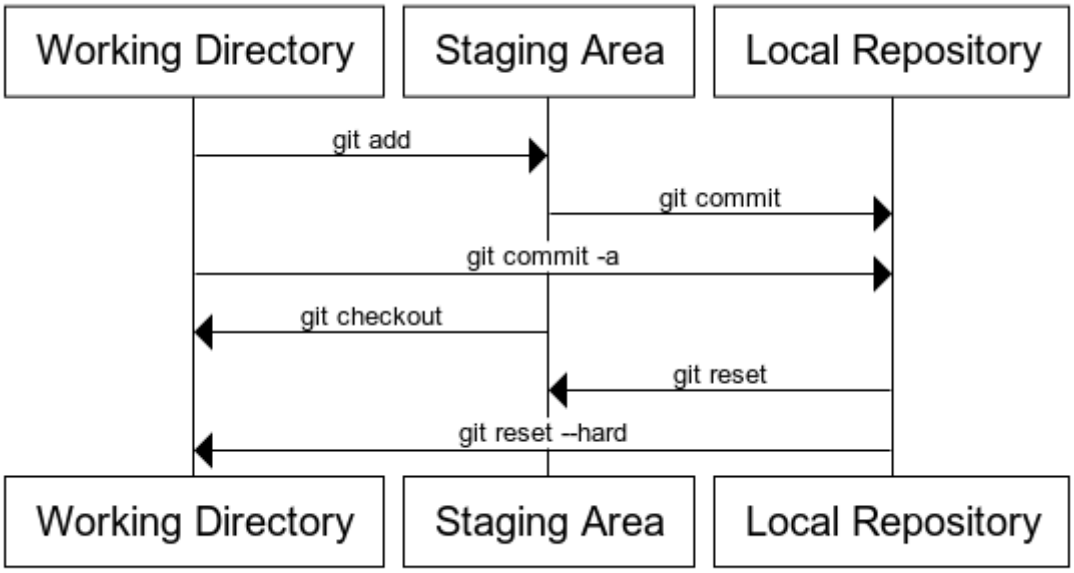
```
In [12]: %%bash
         git checkout index.md
```

```
In [13]: %%bash
         cat index.md
```

```
Mountains and Hills in the UK
==================
England is not very mountainous.
But has some tall hills, and maybe a mountain or two depending on your definition.
```

In [14]:
```
message="""
Working Directory -> Staging Area : git add
Staging Area -> Local Repository : git commit
Working Directory -> Local Repository : git commit -a
Staging Area -> Working Directory : git checkout
Local Repository -> Staging Area : git reset
Local Repository -> Working Directory: git reset --hard
"""
from wsd import wsd
%matplotlib inline
wsd(message)
```

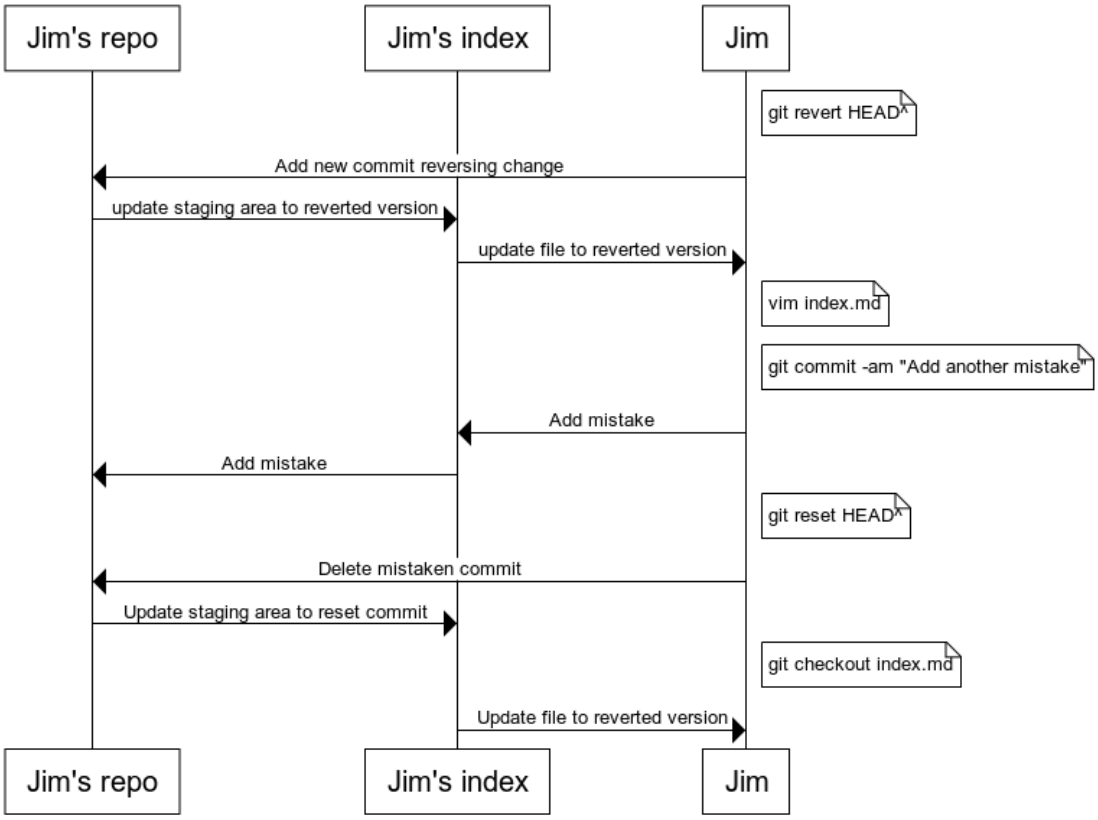
Out[14]:



In [15]:
```
message="""
participant "Jim's repo" as R
participant "Jim's index" as I
participant Jim as J

note right of J: git revert HEAD^

J->R: Add new commit reversing change
R->I: update staging area to reverted version
I->J: update file to reverted version



note right of J: vim index.md
note right of J: git commit -am "Add another mistake"
J->I: Add mistake
I->R: Add mistake
```

```
        note right of J: git reset HEAD^

        J->R: Delete mistaken commit
        R->I: Update staging area to reset commit

        note right of J: git checkout index.md

        I->J: Update file to reverted version


        """
        wsd(message)
```

Out[15]:



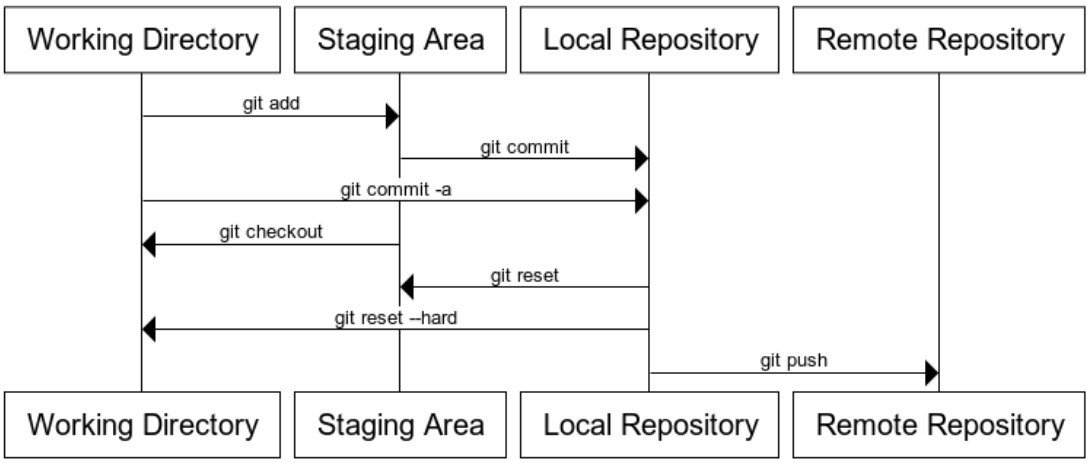

```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
        working_dir
```

```
Out[1]: '/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/git_example'
```

```
In [2]: %%bash
        git remote add origin https://${GITHUB_TOKEN}@github.com/alan-turing-institute/github-example.g
In [3]: %%bash
        git push -uf origin master # I have an extra `f` switch here.
              #You should copy the instructions from YOUR repository.
fatal: could not read Password for 'https://github.com': Device not configured


        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-3-48dc481361e8> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push -uf origin master # I have an extra `f` s


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git push -uf origin master # I have an extra `f` switch here.\n
```

```
In [4]: message="""
        Working Directory -> Staging Area : git add
        Staging Area -> Local Repository : git commit
        Working Directory -> Local Repository : git commit -a
        Staging Area -> Working Directory : git checkout
        Local Repository -> Staging Area : git reset
        Local Repository -> Working Directory: git reset --hard
        Local Repository -> Remote Repository : git push
        """
        from wsd import wsd
        %matplotlib inline
        wsd(message)
```

Out[4]:




```
In [5]: %%writefile lakeland.md
        Lakeland
        ========

        Cumbria has some pretty hills, and lakes too.

Writing lakeland.md
```

```
In [6]: cat lakeland.md

Lakeland
========

Cumbria has some pretty hills, and lakes too.
```

```
In [7]: %%bash
        git commit -am "Try to add Lakeland"
```

```
On branch master
Untracked files:
        __pycache__/
        lakeland.md
        wsd.py

nothing added to commit but untracked files present



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-7-b38098616040> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git commit -am "Try to add Lakeland"\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git commit -am "Try to add Lakeland"\n'' returned non-zero exit s
```

```
In [8]: %%bash
        git add lakeland.md
        git commit -am "Add lakeland"

[master 902cd48] Add lakeland
 1 file changed, 4 insertions(+)
 create mode 100644 lakeland.md


In [9]: %%bash
        git push

fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-9-38bca6a3a98d> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
243             sys.stderr.flush()
244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
246
247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [10]: %%writefile lakeland.md
         Lakeland
         ========

         Cumbria has some pretty hills, and lakes too

         Mountains:
         * Helvellyn
```

```
Overwriting lakeland.md
```

```
In [11]: %%writefile index.md
         Mountains and Lakes in the UK
         ==================
         Engerland is not very mountainous.
         But has some tall hills, and maybe a
         mountain or two depending on your definition.
```

```
Overwriting index.md
```

```
In [12]: %%bash
         git status
```

```
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.md
        modified:   lakeland.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        __pycache__/
        wsd.py

no changes added to commit (use "git add" and/or "git commit -a")
```

```
In [13]: %%bash
         git add index.md
         git commit -m "Include lakes in the scope"
```

```
[master 5bc2d14] Include lakes in the scope
 1 file changed, 4 insertions(+), 3 deletions(-)
```

```
In [14]: %%bash
         git commit -am "Add Helvellyn"
```

```
[master ddd5665] Add Helvellyn
 1 file changed, 4 insertions(+), 1 deletion(-)
```

```
In [15]: %%bash
         git log --oneline
```

```
ddd5665 Add Helvellyn
5bc2d14 Include lakes in the scope
902cd48 Add lakeland
230c644 Revert "Add a lie about a mountain"
84d66d0 Change title
4ad1c69 Add a lie about a mountain
7a758a4 First commit of discourse on UK topography
```

```
In [16]: %%bash
         git push
```

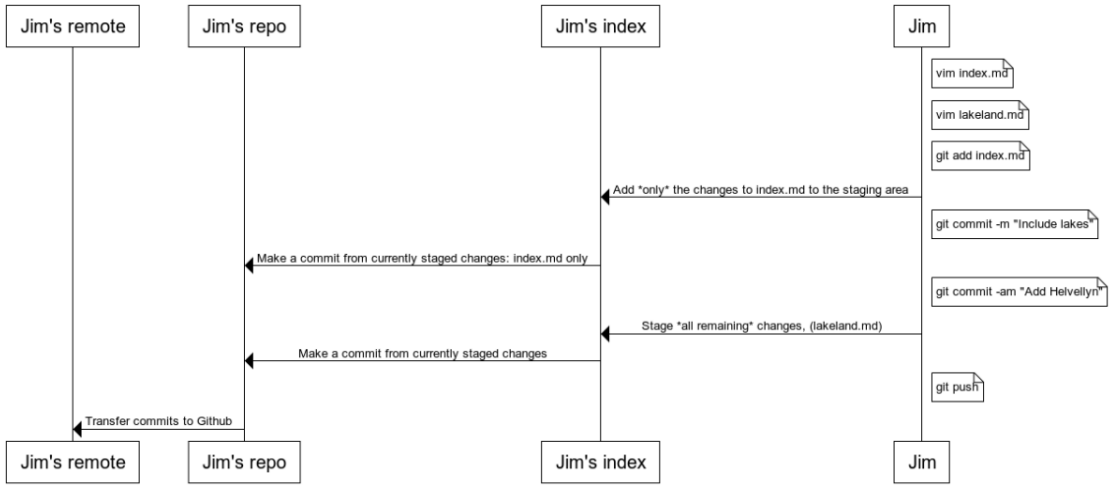
```
fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-16-38bca6a3a98d> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:
```

```
<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
185     # but it's overkill for just that one bit of state.
186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
188
189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
243             sys.stderr.flush()
244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
246
247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [17]: message="""
         participant "Jim's remote" as M
         participant "Jim's repo" as R
         participant "Jim's index" as I
         participant Jim as J

         note right of J: vim index.md
         note right of J: vim lakeland.md

         note right of J: git add index.md
         J->I: Add *only* the changes to index.md to the staging area

         note right of J: git commit -m "Include lakes"
         I->R: Make a commit from currently staged changes: index.md only

         note right of J: git commit -am "Add Helvellyn"
         J->I: Stage *all remaining* changes, (lakeland.md)
         I->R: Make a commit from currently staged changes

         note right of J: git push
         R->M: Transfer commits to Github
         """
         wsd(message)
```

Out[17]:




```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(git_dir)
```

```
In [2]: %%bash
        pwd
        rm -rf github-example # cleanup after previous example
        rm -rf partner_dir # cleanup after previous example
```

```
/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git
```

```
In [3]: %%bash
        pwd
        git clone https://${GITHUB_TOKEN}@github.com/alan-turing-institute/github-example.git
        mv github-example partner_dir
```

```
/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git


Cloning into 'github-example'...
```

```
In [4]: partner_dir=os.path.join(git_dir, 'partner_dir')
        os.chdir(partner_dir)
```

```
In [5]: %%bash
        pwd
        ls
```

```
/Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/partner_dir
Makefile
Scotland.md
Wales.md
index.md
lakeland.md
```

```
In [6]: %%bash
        cat lakeland.md
```

```
Lakeland
========

Cumbria has some pretty hills, and lakes too

Mountains:
* Helvellyn
```

```
In [7]: os.chdir(working_dir)
```

```
In [8]: %%writefile Wales.md
        Mountains In Wales
        ==================

        * Tryfan
        * Yr Wyddfa
```

```
Writing Wales.md
```

```
In [9]: %%bash
        ls
```

```
Wales.md
__pycache__
index.md
lakeland.md
wsd.py
```

```
In [10]: %%bash
         git add Wales.md
         git commit -m "Add wales"
```

```
[master a3a4d7b] Add wales
 1 file changed, 5 insertions(+)
 create mode 100644 Wales.md
```

```
In [11]: os.chdir(partner_dir)
```

```
In [12]: %%writefile Scotland.md
         Mountains In Scotland
         =====================

         * Ben Eighe
         * Cairngorm
```

```
Overwriting Scotland.md
```

```
In [13]: %%bash
         ls
```

```
Makefile
Scotland.md
Wales.md
index.md
lakeland.md
```

```
In [14]: %%bash
         git add Scotland.md
         git commit -m "Add Scotland"
```

```
[master 0ccadc9] Add Scotland
 1 file changed, 1 deletion(-)
```

```
In [15]: %%bash
         git push
```

```
fatal: could not read Password for 'https://github.com': Device not configured
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-15-38bca6a3a98d> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):
```

```
        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [16]: os.chdir(working_dir)
```

```
In [17]: %%bash
         git pull
```

```
warning: no common commits
From https://github.com/alan-turing-institute/github-example
 * [new branch]      gh-pages   -> origin/gh-pages
 * [new branch]      master     -> origin/master
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-17-de7feceb5348> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git pull\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
```

```
    143
    144         # write a basic docstring:


    <decorator-gen-109> in shebang(self, line, cell)


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


    CalledProcessError: Command 'b'git pull\n'' returned non-zero exit status 1.
```

```
In [18]: %%bash
         git push
```

```
fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master
```

```
    ---------------------------------------------------------------------------

    CalledProcessError                        Traceback (most recent call last)

    <ipython-input-18-38bca6a3a98d> in <module>
----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
```

```
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:


<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
185     # but it's overkill for just that one bit of state.
186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
188
189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
243             sys.stderr.flush()
244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
246
247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [19]: os.chdir(partner_dir)
```

```
In [20]: %%bash
         git pull
```

```
Already up to date.
```

```
In [21]: %%bash
         ls
```

```
Makefile
Scotland.md
Wales.md
index.md
lakeland.md
```

```
In [22]: %%writefile Wales.md
         Mountains In Wales
         ==================

         * Tryfan
         * Snowdon
```

```
Overwriting Wales.md
```

```
In [23]: %%bash
         git diff
```

```
diff --git a/Wales.md b/Wales.md
index d8c8384..90f23ec 100644
--- a/Wales.md
+++ b/Wales.md
@@ -1,9 +1,5 @@
 Mountains In Wales
 ==================

-* Pen y Fan
 * Tryfan
 * Snowdon
-* Glyder Fawr
-* Fan y Big
-* Cadair Idris
```

```
In [24]: %%bash
         git commit -am "Translating from the Welsh"
```

```
[master 70fd00e] Translating from the Welsh
 1 file changed, 4 deletions(-)
```

```
In [25]: %%bash
         git log --oneline
```

```
70fd00e Translating from the Welsh
0ccadc9 Add Scotland
ddfdd62 Add a makefile and ignore generated files
9620f2b Merge branch 'experiment'
096d737 Commit Aonach onto master branch
0883623 Add Cadair Idris
bde3b0e Merge branch 'master' of https://github.com/alan-turing-institute/github-example
a002154 Add Glyder
078bbe7 Add another Beacon
fcf2adb Merge branch 'master' of https://github.com/alan-turing-institute/github-example
4dc7c07 Translating from the Welsh
d6cf181 Add a beacon
6d52853 Merge branch 'master' of https://github.com/alan-turing-institute/github-example
0a1c03b Add wales
8ba1b82 Add Scotland
c30a222 Add Helvellyn
d6ed1e1 Include lakes in the scope
de99563 Add lakeland
2703b0a Revert "Add a lie about a mountain"
956ba4a Change title
fc27d65 Add a lie about a mountain
bfc06cd First commit of discourse on UK topography
```

```
In [26]: os.chdir(working_dir)
```

```
In [27]: %%writefile Wales.md
         Mountains In Wales
```

```
         ==================

         * Pen y Fan
         * Tryfan
         * Snowdon

Overwriting Wales.md


In [28]: %%bash
         git commit -am "Add a beacon"

[master b40d7cd] Add a beacon
 1 file changed, 2 insertions(+), 1 deletion(-)


In [29]: %%bash
         git log --oneline

b40d7cd Add a beacon
a3a4d7b Add wales
ddd5665 Add Helvellyn
5bc2d14 Include lakes in the scope
902cd48 Add lakeland
230c644 Revert "Add a lie about a mountain"
84d66d0 Change title
4ad1c69 Add a lie about a mountain
7a758a4 First commit of discourse on UK topography


In [30]: %%bash
         git push

fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master




         ---------------------------------------------------------------------------

         CalledProcessError                        Traceback (most recent call last)

         <ipython-input-30-38bca6a3a98d> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


         /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325
```

```
        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [31]: os.chdir(partner_dir)
```

```
In [32]: %%bash
         git pull
```

```
Already up to date.
```

```
In [33]: %%bash
         git push
```

```
fatal: could not read Password for 'https://github.com': Device not configured
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-33-38bca6a3a98d> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
```

```
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:
```

```
<decorator-gen-109> in shebang(self, line, cell)
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):
```

```
CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [34]: %%bash
         git log --oneline --graph
```

```
* 70fd00e Translating from the Welsh
* 0ccadc9 Add Scotland
* ddfdd62 Add a makefile and ignore generated files
*   9620f2b Merge branch 'experiment'
|\
| * 0883623 Add Cadair Idris
* | 096d737 Commit Aonach onto master branch
|/
*   bde3b0e Merge branch 'master' of https://github.com/alan-turing-institute/github-example
|\
| * 078bbe7 Add another Beacon
* | a002154 Add Glyder
|/
*   fcf2adb Merge branch 'master' of https://github.com/alan-turing-institute/github-example
|\
```

```
| * d6cf181 Add a beacon
* | 4dc7c07 Translating from the Welsh
|/
*   6d52853 Merge branch 'master' of https://github.com/alan-turing-institute/github-example
|\
| * 8ba1b82 Add Scotland
* | 0a1c03b Add wales
|/
* c30a222 Add Helvellyn
* d6ed1e1 Include lakes in the scope
* de99563 Add lakeland
* 2703b0a Revert "Add a lie about a mountain"
* 956ba4a Change title
* fc27d65 Add a lie about a mountain
* bfc06cd First commit of discourse on UK topography
```

```
In [35]: os.chdir(working_dir)
```

```
In [36]: %%bash
         git pull
```

```
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-36-de7feceb5348> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git pull\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
```

```
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git pull\n'' returned non-zero exit status 1.
```

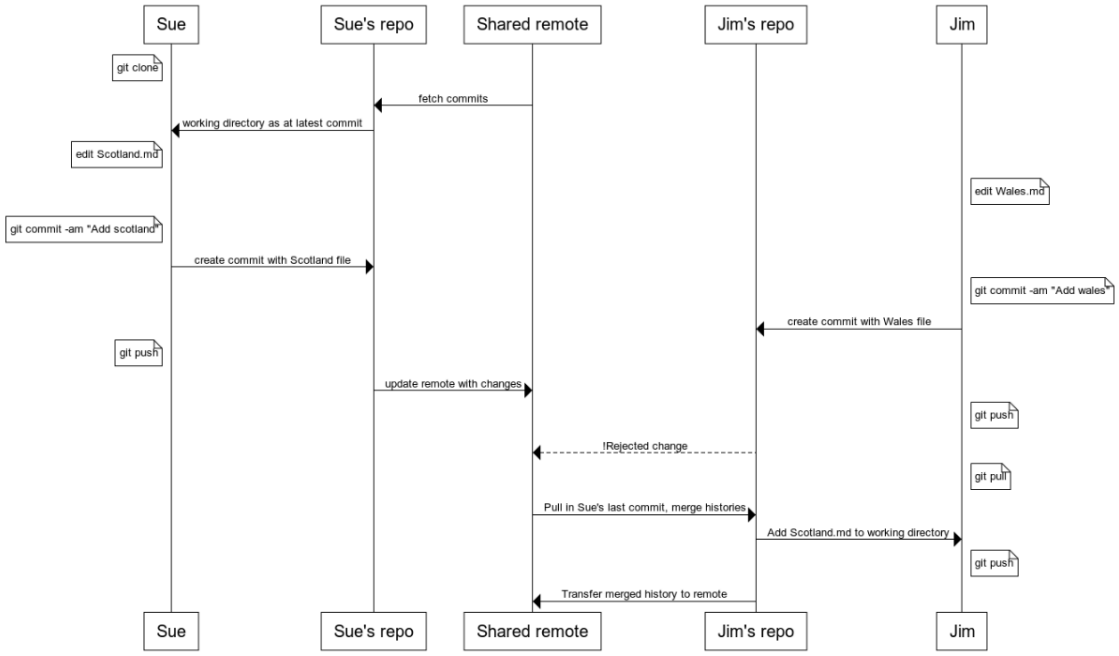
```
In [37]: %%bash
         git log --graph --oneline
```

```
* b40d7cd Add a beacon
* a3a4d7b Add wales
* ddd5665 Add Helvellyn
* 5bc2d14 Include lakes in the scope
* 902cd48 Add lakeland
* 230c644 Revert "Add a lie about a mountain"
* 84d66d0 Change title
* 4ad1c69 Add a lie about a mountain
* 7a758a4 First commit of discourse on UK topography
```

```
In [38]: message="""
         participant Sue as S
         participant "Sue's repo" as SR
         participant "Shared remote" as M
         participant "Jim's repo" as JR
         participant Jim as J

         note left of S: git clone
         M->SR: fetch commits
         SR->S: working directory as at latest commit

         note left of S: edit Scotland.md
         note right of J: edit Wales.md

         note left of S: git commit -am "Add scotland"
```

```
S->SR: create commit with Scotland file

note right of J: git commit -am "Add wales"
J->JR: create commit with Wales file

note left of S: git push
SR->M: update remote with changes

note right of J: git push
JR-->M: !Rejected change

note right of J: git pull
M->JR: Pull in Sue's last commit, merge histories
JR->J: Add Scotland.md to working directory

note right of J: git push
JR->M: Transfer merged history to remote

"""
from wsd import wsd
%matplotlib inline
wsd(message)
```

Out[38]:



In [39]:
```
%%writefile Wales.md
Mountains In Wales
==================

* Pen y Fan
```

```
        * Tryfan
        * Snowdon
        * Fan y Big

Overwriting Wales.md


In [40]: %%bash
         git commit -am "Add another Beacon"
         git push

[master 335ff22] Add another Beacon
 1 file changed, 1 insertion(+)


fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master




        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-40-04be21b514c2> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git commit -am "Add another Beacon"\ngit push\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
```

```
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git commit -am "Add another Beacon"\ngit push\n'' returned non-z
```

```
In [41]: os.chdir(partner_dir)
```

```
In [42]: %%writefile Wales.md
         Mountains In Wales
         ==================

         * Pen y Fan
         * Tryfan
         * Snowdon
         * Glyder Fawr

Overwriting Wales.md
```

```
In [43]: %%bash
         git commit -am "Add Glyder"

[master 0d27c53] Add Glyder
 1 file changed, 2 insertions(+)
```

```
In [44]: %%bash
         git pull

Already up to date.
```

```
In [45]: %%bash
         cat Wales.md

Mountains In Wales
==================

* Pen y Fan
* Tryfan
* Snowdon
* Glyder Fawr
```

```
In [46]: %%writefile Wales.md
         Mountains In Wales
         ==================
```

```
        * Pen y Fan
        * Tryfan
        * Snowdon
        * Glyder Fawr
        * Fan y Big

Overwriting Wales.md
```

```
In [47]: %%bash
         git commit -a --no-edit # I added a No-edit for this non-interactive session. You can edit the

Aborting commit due to empty commit message.
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-47-585cefcfcdcf> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git commit -a --no-edit # I added a No-edit for th


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
```

```
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


    CalledProcessError: Command 'b'git commit -a --no-edit # I added a No-edit for this non-interac
```

```
In [48]: %%bash
         git push
```

```
fatal: could not read Password for 'https://github.com': Device not configured
```

```
    ---------------------------------------------------------------------------

    CalledProcessError                        Traceback (most recent call last)

    <ipython-input-48-38bca6a3a98d> in <module>
----> 1 get_ipython().run_cell_magic('bash', '', 'git push\n')


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:


    <decorator-gen-109> in shebang(self, line, cell)


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
```

```
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git push\n'' returned non-zero exit status 128.
```

```
In [49]: os.chdir(working_dir)
```

```
In [50]: %%bash
         git pull
```

```
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-50-de7feceb5348> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git pull\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
```

```
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git pull\n'' returned non-zero exit status 1.
```

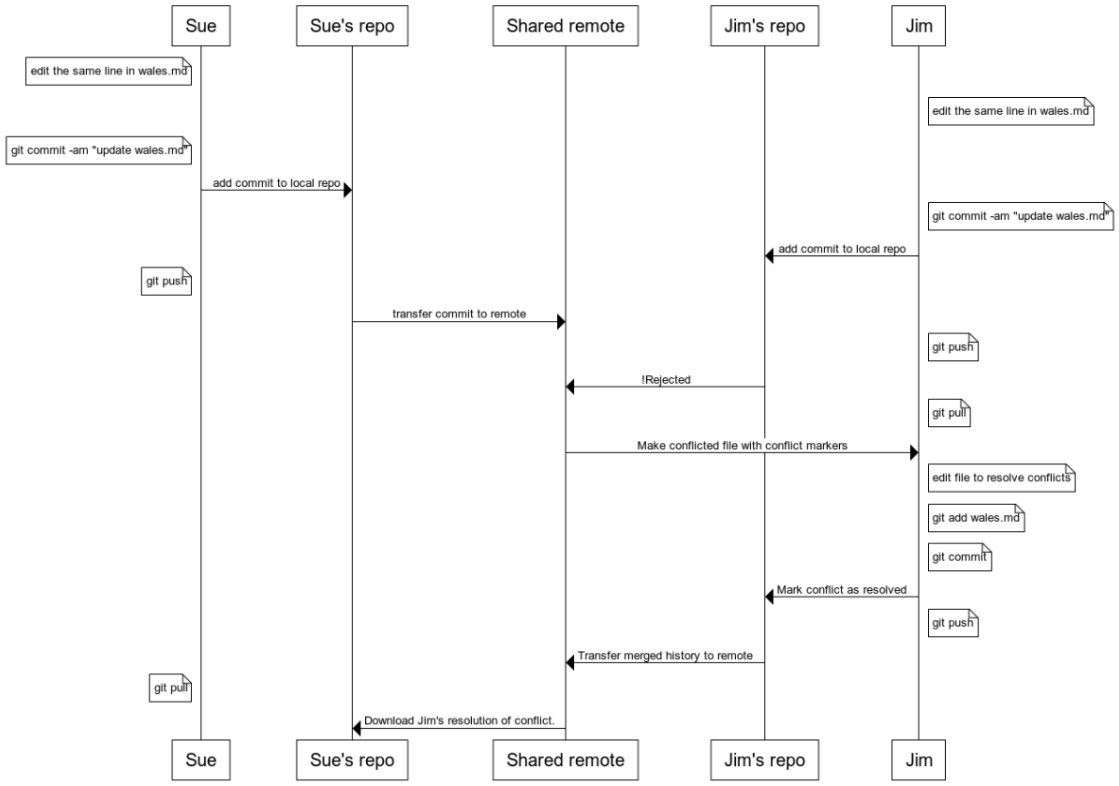
```
In [51]: %%bash
         cat Wales.md

Mountains In Wales
==================

* Pen y Fan
* Tryfan
* Snowdon
* Fan y Big
```

```
In [52]: %%bash
         git log --oneline --graph

* 335ff22 Add another Beacon
* b40d7cd Add a beacon
* a3a4d7b Add wales
* ddd5665 Add Helvellyn
* 5bc2d14 Include lakes in the scope
* 902cd48 Add lakeland
* 230c644 Revert "Add a lie about a mountain"
* 84d66d0 Change title
* 4ad1c69 Add a lie about a mountain
* 7a758a4 First commit of discourse on UK topography
```

```
In [53]: message="""
         participant Sue as S
         participant "Sue's repo" as SR
         participant "Shared remote" as M
         participant "Jim's repo" as JR
         participant Jim as J

         note left of S: edit the same line in wales.md
         note right of J: edit the same line in wales.md

         note left of S: git commit -am "update wales.md"
         S->SR: add commit to local repo
```

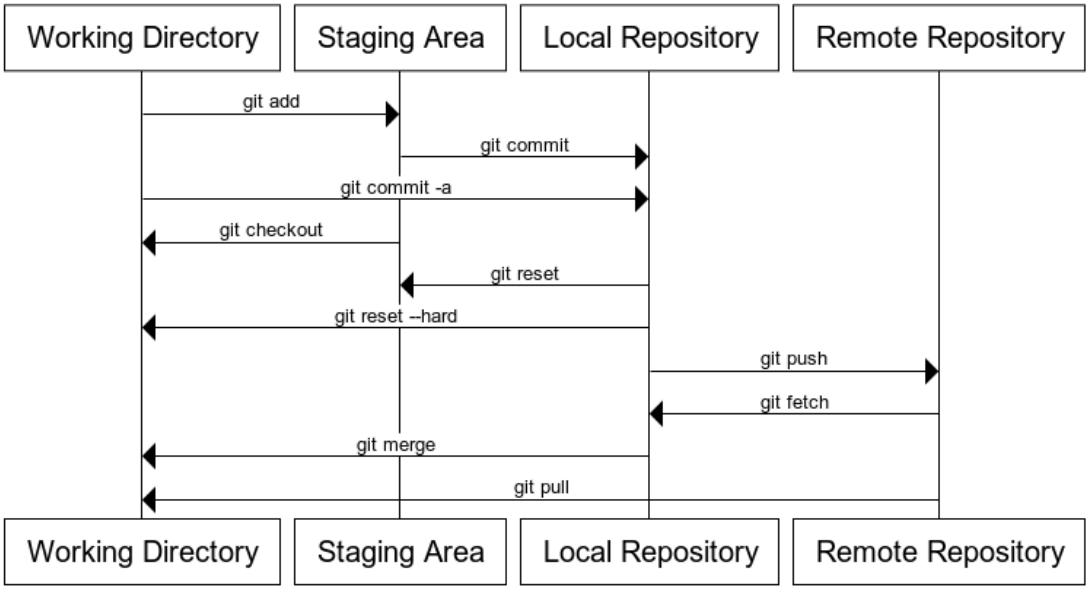
```
        note right of J: git commit -am "update wales.md"
        J->JR: add commit to local repo

        note left of S: git push
        SR->M: transfer commit to remote

        note right of J: git push
        JR->M: !Rejected

        note right of J: git pull
        M->J: Make conflicted file with conflict markers

        note right of J: edit file to resolve conflicts
        note right of J: git add wales.md
        note right of J: git commit
        J->JR: Mark conflict as resolved

        note right of J: git push
        JR->M: Transfer merged history to remote

        note left of S: git pull
        M->SR: Download Jim's resolution of conflict.

        """

        wsd(message)
```

Out[53]:



```
In [54]: message="""
         Working Directory -> Staging Area : git add
         Staging Area -> Local Repository : git commit
         Working Directory -> Local Repository : git commit -a
         Staging Area -> Working Directory : git checkout
         Local Repository -> Staging Area : git reset
         Local Repository -> Working Directory: git reset --hard
         Local Repository -> Remote Repository : git push
         Remote Repository -> Local Repository : git fetch
         Local Repository -> Working Directory : git merge
         Remote Repository -> Working Directory: git pull
         """

         wsd(message)
```

Out[54]:




```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir=os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)

In [2]: %%bash
        git log --graph --oneline

* 335ff22 Add another Beacon
* b40d7cd Add a beacon
* a3a4d7b Add wales
```

```
* ddd5665 Add Helvellyn
* 5bc2d14 Include lakes in the scope
* 902cd48 Add lakeland
* 230c644 Revert "Add a lie about a mountain"
* 84d66d0 Change title
* 4ad1c69 Add a lie about a mountain
* 7a758a4 First commit of discourse on UK topography
```

```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
```

```
In [2]: %%bash
        git branch # Tell me what branches exist
```

```
* master
```

```
In [3]: %%bash
        git checkout -b experiment # Make a new branch
```

```
Switched to a new branch 'experiment'
```

```
In [4]: %%bash
        git branch
```

```
* experiment
  master
```

```
In [5]: %%writefile Wales.md
        Mountains In Wales
        ==================

        * Pen y Fan
        * Tryfan
        * Snowdon
        * Glyder Fawr
        * Fan y Big
        * Cadair Idris
```

```
Overwriting Wales.md
```

```
In [6]: %%bash
        git commit -am "Add Cadair Idris"
```

```
[experiment 53cbc78] Add Cadair Idris
 1 file changed, 2 insertions(+)
```

```
In [7]: %%bash
        git checkout master # Switch to an existing branch
```

```
Switched to branch 'master'
```

```
In [8]: %%bash
        cat Wales.md
```

```
Mountains In Wales
==================

* Pen y Fan
* Tryfan
* Snowdon
* Fan y Big
```

```
In [9]: %%bash
        git checkout experiment
```

```
Switched to branch 'experiment'
```

```
In [10]: cat Wales.md
```

```
Mountains In Wales
==================

* Pen y Fan
* Tryfan
* Snowdon
* Glyder Fawr
* Fan y Big
* Cadair Idris
```

```
In [11]: %%bash
         git push -u origin experiment
```

```
fatal: could not read Password for 'https://github.com': Device not configured



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-11-3df2618a7a35> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push -u origin experiment\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325
```

```
        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git push -u origin experiment\n'' returned non-zero exit status 1
```

In [12]:
```
%%bash
git branch -r
```

```
  origin/gh-pages
  origin/master
```

In [13]:
```
%%bash
git branch -vv
```

```
* experiment 53cbc78 Add Cadair Idris
  master     335ff22 Add another Beacon
```

In [14]:
```
%%bash
git log master..experiment
```

```
commit 53cbc78c4ef5b5fbd7d241cacf3ef5d7b000a08d
Author: Giovanni1085 <gcolavizza@turing.ac.uk>
Date:   Mon Nov 4 11:20:22 2019 +0000

    Add Cadair Idris
```

In [15]:
```
%%bash
git log --graph --oneline HEAD~9..HEAD~5
```

```
* 5bc2d14 Include lakes in the scope
* 902cd48 Add lakeland
* 230c644 Revert "Add a lie about a mountain"
* 84d66d0 Change title
```

```
In [16]: %%bash
         git log --graph --oneline HEAD~5..HEAD
```

```
* 53cbc78 Add Cadair Idris
* 335ff22 Add another Beacon
* b40d7cd Add a beacon
* a3a4d7b Add wales
* ddd5665 Add Helvellyn
```

```
In [17]: %%bash
         git checkout master
```

```
Switched to branch 'master'
```

```
In [18]: %%writefile Scotland.md
         Mountains In Scotland
         ==================

         * Ben Eighe
         * Cairngorm
         * Aonach Eagach
```

```
Writing Scotland.md
```

```
In [19]: %%bash
         git diff Scotland.md
```

```
In [20]: %%bash
         git commit -am "Commit Aonach onto master branch"
```

```
On branch master
Untracked files:
        Scotland.md
        __pycache__/
        wsd.py

nothing added to commit but untracked files present



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-20-6982798a43de> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git commit -am "Commit Aonach onto master branch"
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:


<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git commit -am "Commit Aonach onto master branch"\n'' returned no
```

```
In [21]: %%bash
         git log --left-right --oneline master...experiment
```

```
> 53cbc78 Add Cadair Idris
```

```
In [22]: %%bash
         git branch
         git merge experiment
```

```
  experiment
* master
Updating 335ff22..53cbc78
Fast-forward
 Wales.md | 2 ++
 1 file changed, 2 insertions(+)
```

```
In [23]: %%bash
         git log --graph --oneline HEAD~3..HEAD
```

```
* 53cbc78 Add Cadair Idris
* 335ff22 Add another Beacon
* b40d7cd Add a beacon
```

```
In [24]: %%bash
         git branch
```

```
  experiment
* master
```

```
In [25]: %%bash
         git branch -d experiment
```

```
Deleted branch experiment (was 53cbc78).
```

```
In [26]: %%bash
         git branch
```

```
* master
```

```
In [27]: %%bash
         git branch --remote
```

```
  origin/gh-pages
  origin/master
```

```
In [28]: %%bash
         git push --delete origin experiment
         # Remove remote branch
         # - also can use github interface
```

```
fatal: could not read Password for 'https://github.com': Device not configured
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-28-0d7439fb72e4> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git push --delete origin experiment \n# Remove re

        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
140             else:
141                 line = script
--> 142             return self.shebang(line, cell)
143
144         # write a basic docstring:


<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
185     # but it's overkill for just that one bit of state.
186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
188
189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
243             sys.stderr.flush()
244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
246
247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git push --delete origin experiment \n# Remove remote branch \n#
```

```
In [29]: %%bash
         git branch --remote
```

```
  origin/gh-pages
  origin/master
```

```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir = os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
```

```
In [2]: %%writefile Wales.md
        Mountains In Wales
        ==================

        * Pen y Fan
        * Tryfan
        * Snowdon
        * Glyder Fawr
        * Fan y Big
        * Cadair Idris
```

```
Overwriting Wales.md
```

```
In [3]: %%bash
        git pull
```

```
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

    git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

    git branch --set-upstream-to=origin/<branch> master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-3-de7feceb5348> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git pull\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
```

```
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


    CalledProcessError: Command 'b'git pull\n'' returned non-zero exit status 1.
```

```
In [4]: %%bash
        git stash apply
```

```
No stash entries found.
```

```
    ---------------------------------------------------------------------------

    CalledProcessError                        Traceback (most recent call last)

    <ipython-input-4-c47375944a4e> in <module>
----> 1 get_ipython().run_cell_magic('bash', '', 'git stash apply\n')


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:


    <decorator-gen-109> in shebang(self, line, cell)


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
```

```
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git stash apply\n'' returned non-zero exit status 1.
```

```
In [5]: %%writefile Makefile

        MDS=$(wildcard *.md)
        PDFS=$(MDS:.md=.pdf)

        default: $(PDFS)

        %.pdf: %.md
                pandoc $< -o $@
```

```
Writing Makefile
```

```
In [6]: %%bash
        make
```

```
pandoc Scotland.md -o Scotland.pdf
pandoc Wales.md -o Wales.pdf
pandoc index.md -o index.pdf
pandoc lakeland.md -o lakeland.pdf
```

```
In [7]: %%bash
        git status
```

```
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        Makefile
        Scotland.md
        Scotland.pdf
        Wales.pdf
        __pycache__/
        index.pdf
        lakeland.pdf
        wsd.py

nothing added to commit but untracked files present (use "git add" to track)
```

```
In [8]: %%writefile .gitignore
        *.pdf
```

```
Writing .gitignore
```

```
In [9]: %%bash
        git status
```

```
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        .gitignore
        Makefile
        Scotland.md
        __pycache__/
        wsd.py

nothing added to commit but untracked files present (use "git add" to track)
```

```
In [10]: %%bash
         git add Makefile
         git add .gitignore
         git commit -am "Add a makefile and ignore generated files"
         git push
```

```
[master 13941ea] Add a makefile and ignore generated files
 2 files changed, 9 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 Makefile


fatal: The current branch master has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin master
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-10-df7f373b34b5> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git add Makefile\ngit add .gitignore\ngit commit -


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:
```

```
        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'git add Makefile\ngit add .gitignore\ngit commit -am "Add a makef
```

```
In [11]: %%bash
         git clean -fX
```

```
Removing Scotland.pdf
Removing Wales.pdf
Removing index.pdf
Removing lakeland.pdf
```

```
In [12]: %%bash
         ls
```

```
Makefile
Scotland.md
Wales.md
__pycache__
index.md
lakeland.md
wsd.py
```

```
In [13]: %%writefile index.md
         ---
         title: Github Pages Example
         ---
         Mountains and Lakes in the UK
         ==================

         Engerland is not very mountainous.
         But has some tall hills, and maybe a mountain or two depending on your definition.
```

```
Overwriting index.md
```

```
In [14]: %%bash
         git commit -am "Add github pages YAML frontmatter"

[master 7da2ba7] Add github pages YAML frontmatter
 1 file changed, 7 insertions(+), 4 deletions(-)


In [15]: os.chdir(working_dir)

In [16]: %%bash

         git checkout -b gh-pages
         git push -uf origin gh-pages

Switched to a new branch 'gh-pages'
fatal: could not read Password for 'https://github.com': Device not configured



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-16-9fdaecd874b1> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', '\ngit checkout -b gh-pages\ngit push -uf origin gh


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
```

```
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'\ngit checkout -b gh-pages\ngit push -uf origin gh-pages\n'' retu
```

```
In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        working_dir=os.path.join(git_dir, 'git_example')
        os.chdir(working_dir)
```

```
In [2]: %%bash
        git checkout master
        git remote add jamespjh https://${GITHUB_TOKEN}@github.com/Giovanni1085/github-example.git
```

```
Switched to branch 'master'
```

```
In [3]: %%writefile Pennines.md

        Mountains In the Pennines
        =========================

        * Cross Fell
        * Whernside
```

```
Writing Pennines.md
```

```
In [4]: %%bash
        git commit -am "Add Whernside"
```

```
On branch master
Untracked files:
        Pennines.md
        Scotland.md
        __pycache__/
        wsd.py

nothing added to commit but untracked files present
```

```
        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-4-892653cd7911> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'git commit -am "Add Whernside"\n')
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:


<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git commit -am "Add Whernside"\n'' returned non-zero exit status
```

In [5]:
```
%%bash
git push -uf jamespjh master
```

```
fatal: could not read Password for 'https://github.com': Device not configured



---------------------------------------------------------------------------

CalledProcessError                        Traceback (most recent call last)

<ipython-input-5-fc6717d06ab2> in <module>
----> 1 get_ipython().run_cell_magic('bash', '', 'git push -uf jamespjh master\n')


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
   2321             magic_arg_s = self.var_expand(line, stack_depth)
```

```
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
140             else:
141                 line = script
--> 142             return self.shebang(line, cell)
143
144         # write a basic docstring:


<decorator-gen-109> in shebang(self, line, cell)


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
185     # but it's overkill for just that one bit of state.
186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
188
189         if callable(arg):


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
243             sys.stderr.flush()
244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
246
247     def _run_script(self, p, cell, to_close):


CalledProcessError: Command 'b'git push -uf jamespjh master\n'' returned non-zero exit status 1
```

In [6]:
```
%%bash
git fetch
git log --oneline --left-right jamespjh/master...origin/master
```

```
fatal: ambiguous argument 'jamespjh/master...origin/master': unknown revision or path not in the workin
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
```

```
---------------------------------------------------------------------------

CalledProcessError                        Traceback (most recent call last)

<ipython-input-6-4b087a872018> in <module>
----> 1 get_ipython().run_cell_magic('bash', '', 'git fetch\ngit log --oneline --left-right jamespj


/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
```

```
   2321             magic_arg_s = self.var_expand(line, stack_depth)
   2322             with self.builtin_trap:
-> 2323                 result = fn(magic_arg_s, cell)
   2324             return result
   2325
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    140             else:
    141                 line = script
--> 142             return self.shebang(line, cell)
    143
    144         # write a basic docstring:
```

```
<decorator-gen-109> in shebang(self, line, cell)
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):
```

```
/usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):
```

```
CalledProcessError: Command 'b'git fetch\ngit log --oneline --left-right jamespjh/master...origi
```

```
In [7]: %%bash
        git diff --name-only origin/master
```

```
Scotland.md
index.md
```

```
In [8]: %%bash
        git branch -vv
```

```
  gh-pages 7da2ba7 Add github pages YAML frontmatter
* master   7da2ba7 Add github pages YAML frontmatter
```

```
In [9]: bare_dir=os.path.join(git_dir, 'bare_repo')
        os.chdir(git_dir)
```

```
In [10]: %%bash
         mkdir -p bare_repo
```

```
         cd bare_repo
         git init --bare

Initialized empty Git repository in /Users/edaub/Projects/rsd-engineeringcourse/ch02git/learning_git/ba


In [11]: os.chdir(working_dir)

In [12]: %%bash
         git remote add local_bare ../bare_repo
         git push -u local_bare master

Branch 'master' set up to track remote branch 'master' from 'local_bare'.


To ../bare_repo
 * [new branch]      master -> master


In [1]: import os
        top_dir = os.getcwd()
        git_dir = os.path.join(top_dir, 'learning_git')
        os.chdir(git_dir)

In [2]: %%bash
        rm -rf bisectdemo
        git clone https://github.com/shawnsi/bisectdemo.git

Cloning into 'bisectdemo'...


In [3]: bisect_dir=os.path.join(git_dir,'bisectdemo')
        os.chdir(bisect_dir)

In [4]: %%bash
        python squares.py 2 # 4

4


In [5]: %%bash
        ./breakme.sh > break_output

error: branch 'buggy' not found.
Switched to a new branch 'buggy'


In [6]: python squares.py 2 #ăError message


          File "<ipython-input-6-8e2377cd54bf>", line 1
        python squares.py 2 #ăError message
                            ^
    SyntaxError: invalid syntax
```

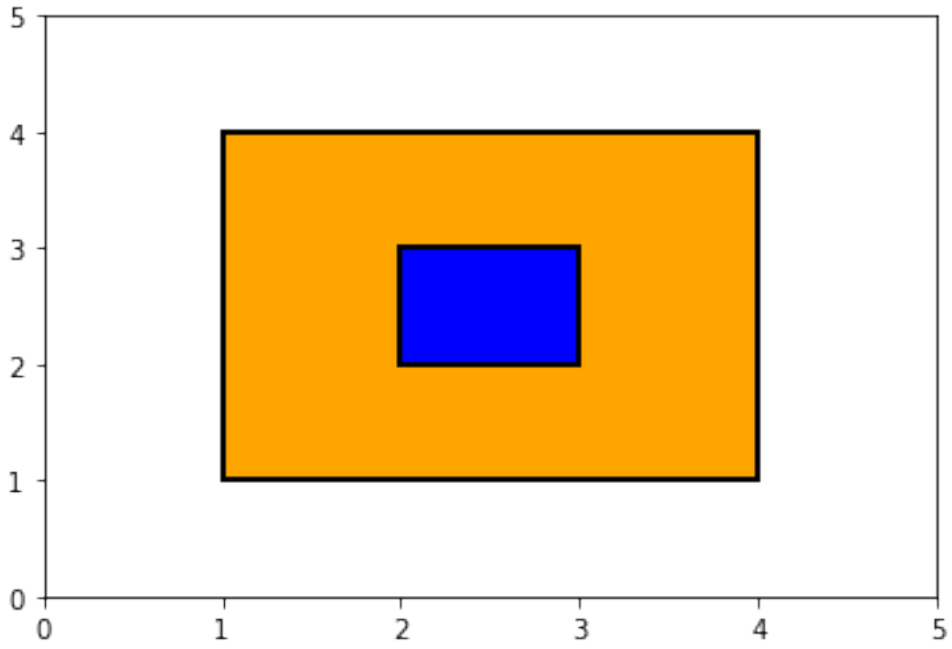
```
In [7]: %%bash
        git bisect start
        git bisect bad # We know the current state is broken
        git checkout master
        git bisect good # We know the master branch state is OK

Your branch is up to date with 'origin/master'.
Bisecting: 500 revisions left to test after this (roughly 9 steps)
[3e4632fd08b2316a590d477747f18d64aa2b7307] Comment 499


Switched to branch 'master'


In [8]: %%bash
        git bisect start
        git bisect bad HEAD # We know the current state is broken
        git bisect good master #ãWe know master is good
        git bisect run python squares.py 2

Bisecting: 500 revisions left to test after this (roughly 9 steps)
[3e4632fd08b2316a590d477747f18d64aa2b7307] Comment 499
running python squares.py 2
Bisecting: 249 revisions left to test after this (roughly 8 steps)
[d33e44bc1ac39ed6a5a4b253a17fe80d0ab444ee] Comment 249
running python squares.py 2
Bisecting: 124 revisions left to test after this (roughly 7 steps)
[c2776908adc968c9b8ba6dbe266de84c31fc910e] Comment 125
running python squares.py 2
4
Bisecting: 62 revisions left to test after this (roughly 6 steps)
[5c20adfb87a4fa912607476e496b3a97035ff1c8] Comment 186
running python squares.py 2
Bisecting: 30 revisions left to test after this (roughly 5 steps)
[2f09592995242119db05da1665db8cec45c638e8] Comment 156
running python squares.py 2
4
Bisecting: 15 revisions left to test after this (roughly 4 steps)
[499bcbbd7b5a4b6a281b82cb61545092851bdb10] Comment 171
running python squares.py 2
4
Bisecting: 7 revisions left to test after this (roughly 3 steps)
[455c5abb21ff7d01b5a2291bbf9e651d5e976679] Comment 179
running python squares.py 2
4
Bisecting: 3 revisions left to test after this (roughly 2 steps)
[9c1ef02bb78daabf95aa3299b513c2e6abea76fd] Comment 183
running python squares.py 2
4
Bisecting: 1 revision left to test after this (roughly 1 step)
[cc43fea2395ae2ee5cc3579e35d2d634e4bc65c0] Breaking argument type
running python squares.py 2
Bisecting: 0 revisions left to test after this (roughly 0 steps)
[5c8efcf8446bc51af7819d447d2553e5d0f884fd] Comment 184
running python squares.py 2
```

```
4
cc43fea2395ae2ee5cc3579e35d2d634e4bc65c0 is the first bad commit
commit cc43fea2395ae2ee5cc3579e35d2d634e4bc65c0
Author: Shawn Siefkas <shawn.siefkas@meredith.com>
Date:   Thu Nov 14 09:23:55 2013 -0600

    Breaking argument type

:100644 100644 2f69830539c0fb9d486e2c79529a7030f7ac451b d589251e34a300e356b5737a23dce93e6d1cc380 M
bisect run success


Previous HEAD position was 3e4632f Comment 499
Switched to branch 'buggy'
Traceback (most recent call last):
  File "squares.py", line 9, in <module>
    print(integer**2)
TypeError: unsupported operand type(s) for ** or pow(): 'str' and 'int'
Traceback (most recent call last):
  File "squares.py", line 9, in <module>
    print(integer**2)
TypeError: unsupported operand type(s) for ** or pow(): 'str' and 'int'
Traceback (most recent call last):
  File "squares.py", line 9, in <module>
    print(integer**2)
TypeError: unsupported operand type(s) for ** or pow(): 'str' and 'int'
Traceback (most recent call last):
  File "squares.py", line 9, in <module>
    print(integer**2)
TypeError: unsupported operand type(s) for ** or pow(): 'str' and 'int'
```

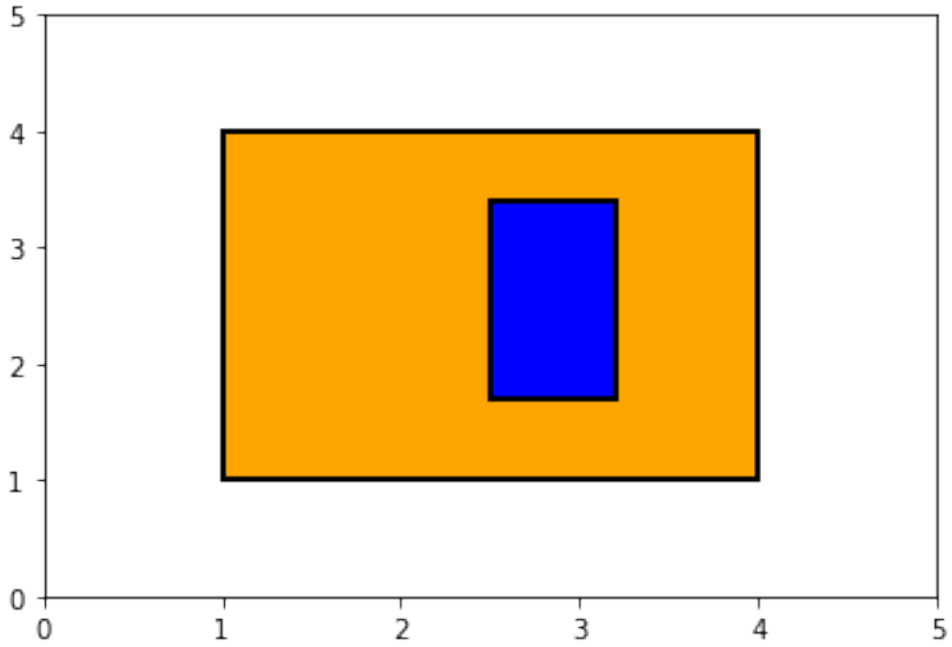
```
In [1]: import matplotlib.pyplot as plt
        from matplotlib.path import Path
        import matplotlib.patches as patches
        %matplotlib inline
```

```
In [2]: def show_fields(field1, field2):
            def vertices(left, bottom, right, top):
                verts = [(left, bottom),
                         (left, top),
                         (right, top),
                         (right, bottom),
                         (left, bottom)]
                return verts

            codes = [Path.MOVETO,
                     Path.LINETO,
                     Path.LINETO,
                     Path.LINETO,
                     Path.CLOSEPOLY]
            path1 = Path(vertices(*field1), codes)
            path2 = Path(vertices(*field2), codes)
            fig = plt.figure()
            ax = fig.add_subplot(111)
```

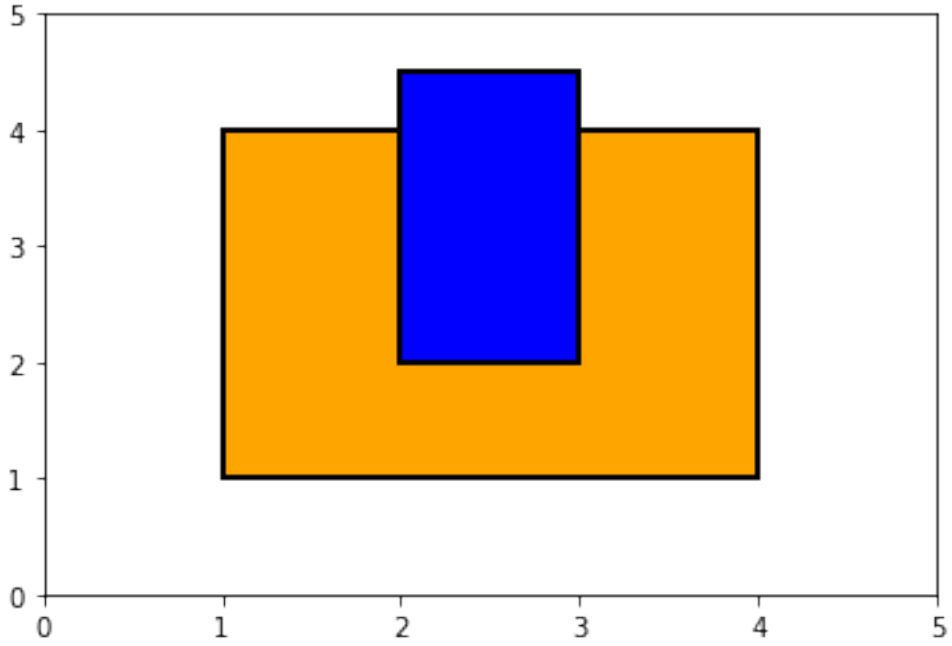
```
    patch1 = patches.PathPatch(path1, facecolor='orange', lw=2)
    patch2 = patches.PathPatch(path2, facecolor='blue', lw=2)
    ax.add_patch(patch1)
    ax.add_patch(patch2)
    ax.set_xlim(0,5)
    ax.set_ylim(0,5)

show_fields((1.,1.,4.,4.),(2.,2.,3.,3.))
```




```
In [3]: show_fields((1.,1.,4.,4.),(2.5,1.7,3.2,3.4))
```



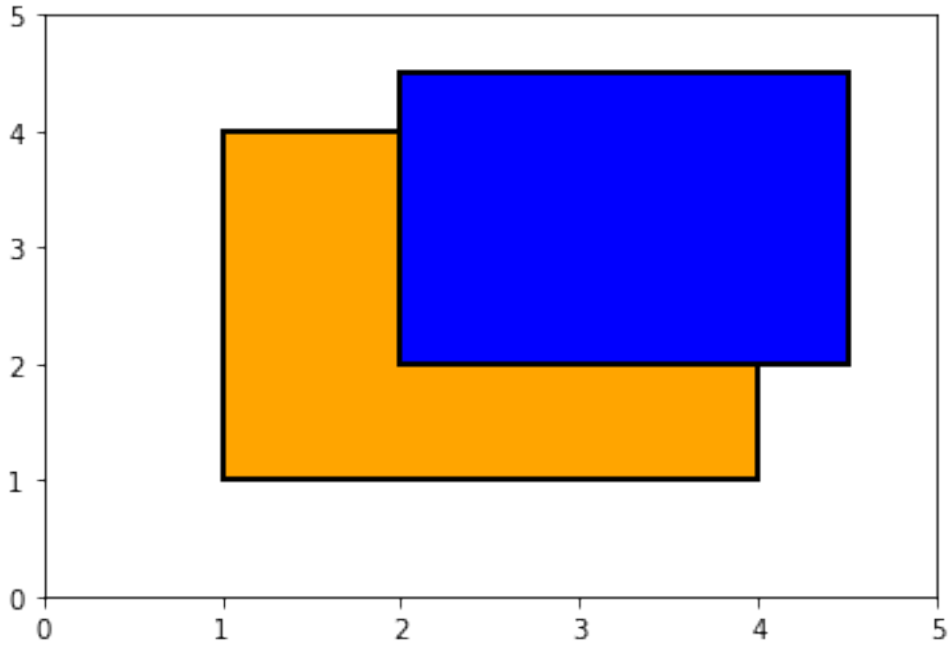

```
In [4]: show_fields((1.,1.,4.,4.),(2.,2.,3.,4.5))
```



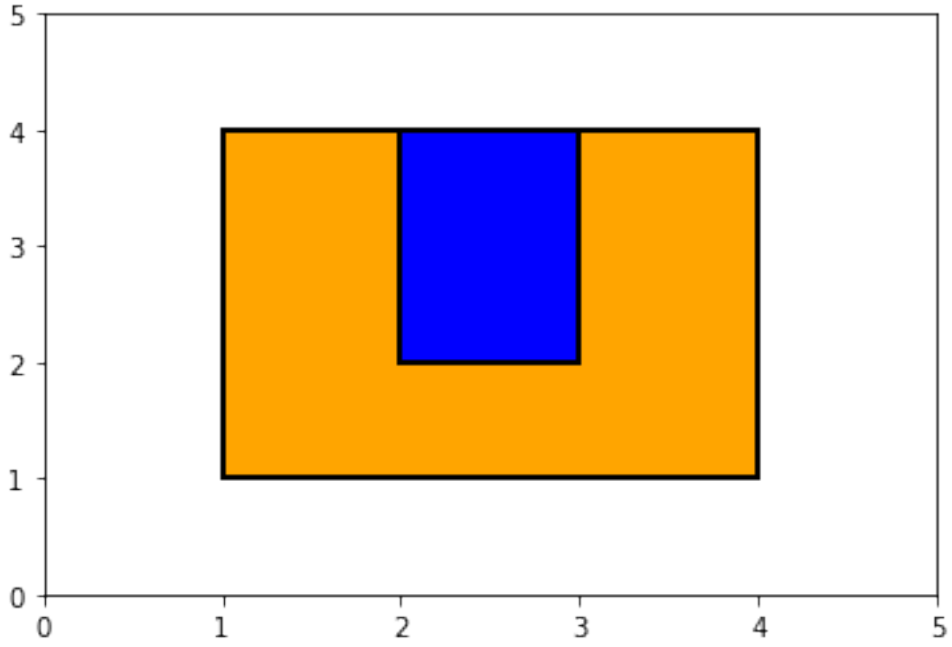

```
In [5]: for _ in range(50):
            print("Spoiler space")
```

```
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
Spoiler space
```

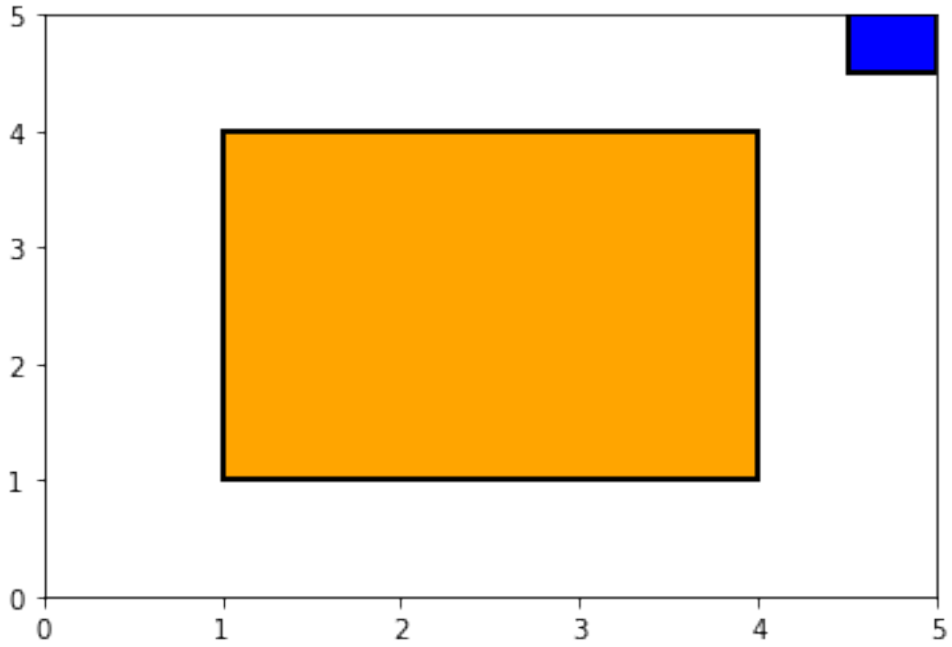
```
In [6]: show_fields((1.,1.,4.,4.),(2,2,4.5,4.5)) # Overlap corner
```



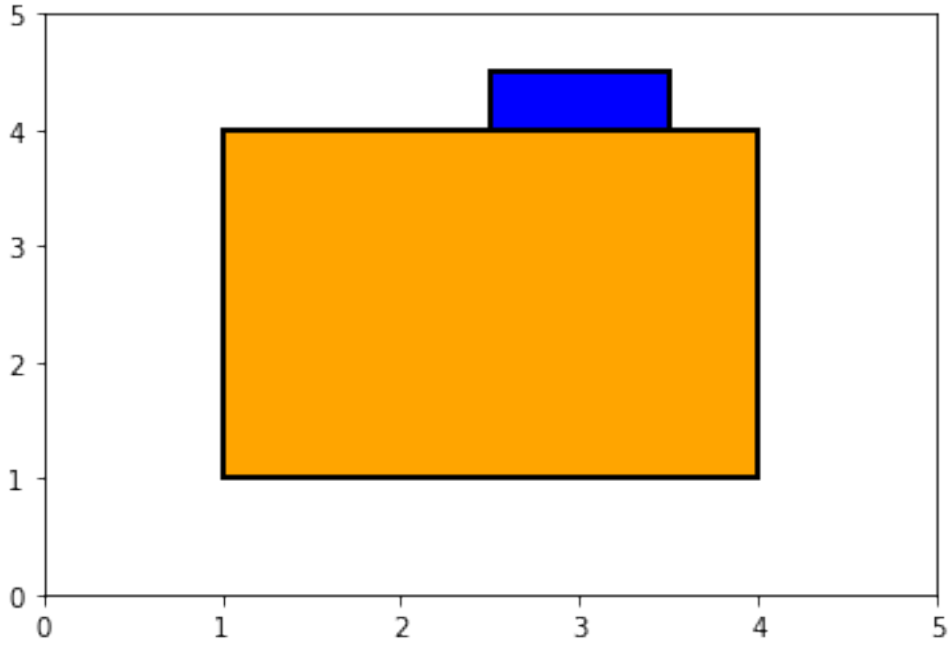

```
In [7]: show_fields((1.,1.,4.,4.),(2.,2.,3.,4.)) # Just touching
```



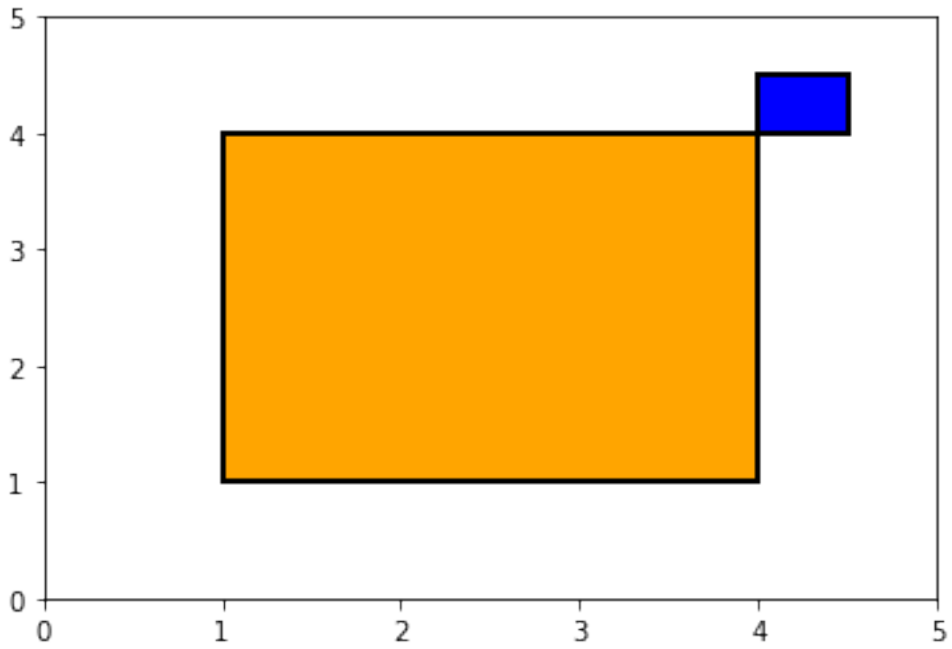

```
In [8]: show_fields((1.,1.,4.,4.),(4.5,4.5,5,5)) # No overlap
```




```
In [9]: show_fields((1.,1.,4.,4.),(2.5,4,3.5,4.5)) # Just touching from outside
```




```
In [10]: show_fields((1.,1.,4.,4.),(4,4,4.5,4.5)) # Touching corner
```



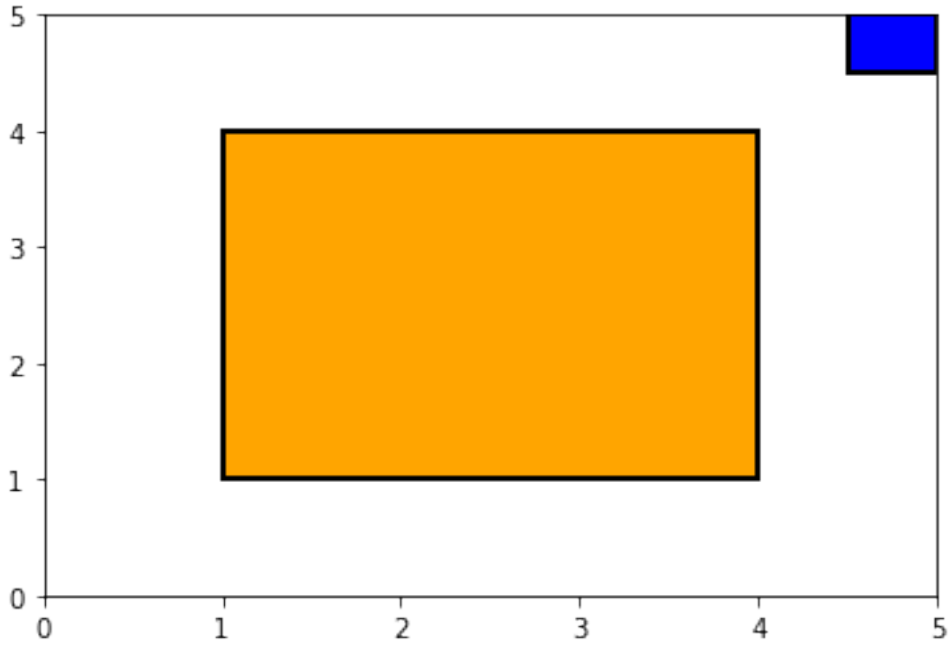

```
In [11]: def overlap(field1, field2):
             left1, bottom1, top1, right1 = field1
             left2, bottom2, top2, right2 = field2
             overlap_left=max(left1, left2)
             overlap_bottom=max(bottom1, bottom2)
             overlap_right=min(right1, right2)
             overlap_top=min(top1, top2)
             overlap_height=(overlap_top-overlap_bottom)
             overlap_width=(overlap_right-overlap_left)
             return overlap_height*overlap_width

In [12]: overlap((1.,1.,4.,4.),(2.,2.,3.,3.))

Out[12]: 1.0

In [13]: assert overlap((1.,1.,4.,4.),(2.,2.,3.,3.)) == 1.0

In [14]: assert overlap((1.,1.,4.,4.),(2.,2.,3.,4.5)) == 2.0

In [15]: assert overlap((1.,1.,4.,4.),(2.,2.,4.5,4.5)) == 4.0

In [16]: assert overlap((1.,1.,4.,4.),(4.5,4.5,5,5)) == 0.0

         ---------------------------------------------------------------------------

         AssertionError                            Traceback (most recent call last)

         <ipython-input-16-21bafdf6842e> in <module>
         ----> 1 assert overlap((1.,1.,4.,4.),(4.5,4.5,5,5)) == 0.0
```

```
        AssertionError:
```

```
In [17]: print(overlap((1.,1.,4.,4.),(4.5,4.5,5,5)))
```

```
0.25
```

```
In [18]: show_fields((1.,1.,4.,4.),(4.5,4.5,5,5))
```




```
In [19]: overlap_left=4.5
         overlap_right=4
         overlap_width=-0.5
         overlap_height=-0.5
```

```
In [20]: def overlap(field1, field2):
             left1, bottom1, top1, right1 = field1
             left2, bottom2, top2, right2 = field2

             overlap_left=max(left1, left2)
             overlap_bottom=max(bottom1, bottom2)
             overlap_right=min(right1, right2)
             overlap_top=min(top1, top2)

             overlap_height=max(0, (overlap_top-overlap_bottom))
             overlap_width=max(0, (overlap_right-overlap_left))

             return overlap_height*overlap_width
```

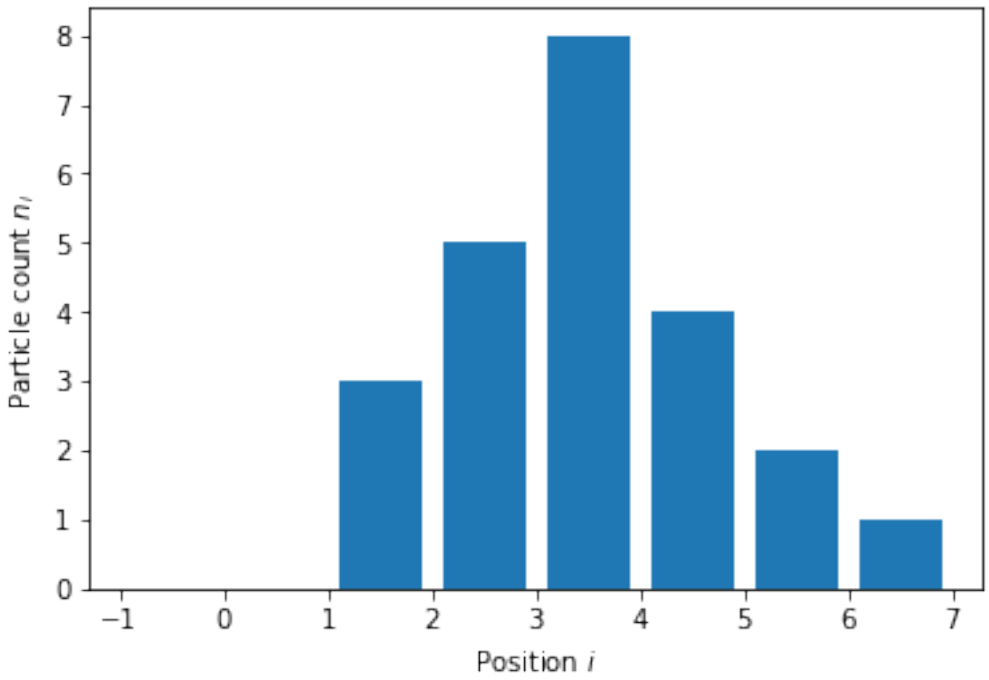
```
In [21]: assert overlap((1,1,4,4),(2,2,3,3)) == 1.0
         assert overlap((1,1,4,4),(2,2,3,4.5)) == 2.0
         assert overlap((1,1,4,4),(2,2,4.5,4.5)) == 4.0
         assert overlap((1,1,4,4),(4.5,4.5,5,5)) == 0.0
         assert overlap((1,1,4,4),(2.5,4,3.5,4.5)) == 0.0
         assert overlap((1,1,4,4),(4,4,4.5,4.5)) == 0.0

In [22]: def I_only_accept_positive_numbers(number):
             # Check input
             if number < 0:
                 raise ValueError("Input "+ str(number)+" is negative")

             # Do something

In [23]: I_only_accept_positive_numbers(5)

In [24]: I_only_accept_positive_numbers(-5)


         ---------------------------------------------------------------------------

         ValueError                                Traceback (most recent call last)

         <ipython-input-24-ac3b0fd3c476> in <module>
     ----> 1 I_only_accept_positive_numbers(-5)


         <ipython-input-22-2403a45f688e> in I_only_accept_positive_numbers(number)
           2     # Check input
           3     if number < 0:
     ----> 4         raise ValueError("Input "+ str(number)+" is negative")
           5
           6     # Do something


         ValueError: Input -5 is negative


In [25]: assert I_only_accept_positive_numbers(-5) == # Gives a value error


           File "<ipython-input-25-55b8782568ca>", line 1
         assert I_only_accept_positive_numbers(-5) == # Gives a value error
                                                                            ^
     SyntaxError: invalid syntax



In [1]: def I_only_accept_positive_numbers(number):
            # Check input
            if number < 0:
                raise ValueError("Input "+ str(number)+" is negative")

            # Do something

In [2]: from pytest import raises
```

```
In [3]: with raises(ValueError):
            I_only_accept_positive_numbers(-5)

In [4]: %%bash
        mkdir -p saskatchewan
        touch saskatchewan/__init__.py

In [5]: %%writefile saskatchewan/overlap.py
        def overlap(field1, field2):
            left1, bottom1, top1, right1 = field1
            left2, bottom2, top2, right2 = field2

            overlap_left=max(left1, left2)
            overlap_bottom=max(bottom1, bottom2)
            overlap_right=min(right1, right2)
            overlap_top=min(top1, top2)
            # Here's our wrong code again
            overlap_height=(overlap_top-overlap_bottom)
            overlap_width=(overlap_right-overlap_left)

            return overlap_height*overlap_width

Overwriting saskatchewan/overlap.py


In [6]: %%writefile saskatchewan/test_overlap.py
        from .overlap import overlap

        def test_full_overlap():
            assert overlap((1.,1.,4.,4.),(2.,2.,3.,3.)) == 1.0

        def test_partial_overlap():
            assert overlap((1,1,4,4),(2,2,3,4.5)) == 2.0

        def test_no_overlap():
            assert overlap((1,1,4,4),(4.5,4.5,5,5)) == 0.0

Overwriting saskatchewan/test_overlap.py


In [7]: %%bash
        cd saskatchewan
        py.test

============================= test session starts ==============================
platform darwin -- Python 3.7.2, pytest-4.1.1, py-1.7.0, pluggy-0.8.0
rootdir: /Users/edaub/Projects/rsd-engineeringcourse/ch03tests/saskatchewan, inifile:
collected 3 items

test_overlap.py ..F                                                      [100%]

=================================== FAILURES ===================================
_______________________________ test_no_overlap ________________________________

    def test_no_overlap():
>       assert overlap((1,1,4,4),(4.5,4.5,5,5)) == 0.0
```

```
E       assert 0.25 == 0.0
E        +  where 0.25 = overlap((1, 1, 4, 4), (4.5, 4.5, 5, 5))

test_overlap.py:10: AssertionError
====================== 1 failed, 2 passed in 0.08 seconds ======================



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-7-40dce32fa8c8> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'cd saskatchewan\npy.test\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'cd saskatchewan\npy.test\n'' returned non-zero exit status 1.
```

In [8]: `1000.0 * 1.0 - 1000.0 * 0.9999999999999998`

```
Out[8]: 2.2737367544323206e-13

In [9]: 1000.0 * (1.0 - 0.9999999999999998)

Out[9]: 2.220446049250313e-13

In [10]: 1000.0 * 1e5 - 1000.0 * 0.9999999999999998e5

Out[10]: 1.4901161193847656e-08

In [11]: from pytest import approx
         assert  0.7 == approx(0.7 + 1e-7)

In [12]: magnitude = 0.7
         assert 0.7 == approx(0.701 , rel=0.1, abs=0.1)

In [13]: from numpy import array, pi

         vector_of_reals = array([0.1, 0.2, 0.3, 0.4]) * pi

In [14]: from numpy import array, pi
         from numpy.testing import assert_allclose
         expected = array([0.1, 0.2, 0.3, 0.4, 1e-12]) * pi
         actual = array([0.1, 0.2, 0.3, 0.4, 2e-12]) * pi
         actual[:-1] += 1e-6

         assert_allclose(actual, expected, rtol=1e-5, atol=1e-8)

In [1]: import numpy as np
        from matplotlib import pyplot as plt
        %matplotlib inline

        density =  np.array([0, 0, 3, 5, 8, 4, 2, 1])
        fig, ax = plt.subplots()
        ax.bar(np.arange(len(density))-0.5, density)
        ax.xrange=[-0.5, len(density)-0.5]
        ax.set_ylabel("Particle count $n_i$")
        ax.set_xlabel("Position $i$")

Out[1]: Text(0.5, 0, 'Position $i$')
```




```
In [2]: %%bash
        mkdir -p diffusion
        touch diffusion/__init__.py
```

```
In [3]: %%writefile diffusion/model.py
        def energy(density, coeff=1.0):
          """ Energy associated with the diffusion model

              Parameters
              ----------

              density: array of positive integers
                  Number of particles at each position i in the array
              coeff: float
                  Diffusion coefficient.
          """
          # implementation goes here
```

```
Overwriting diffusion/model.py
```

```
In [4]: %%writefile diffusion/test_model.py
        from .model import energy
        def test_energy():
          """ Optional description for nose reporting """
          # Test something
```

```
Overwriting diffusion/test_model.py
```

```
In [5]: %%bash
        cd diffusion
        py.test
```

```
============================= test session starts ==============================
platform darwin -- Python 3.7.2, pytest-4.1.1, py-1.7.0, pluggy-0.8.0
rootdir: /Users/edaub/Projects/rsd-engineeringcourse/ch03tests/diffusion, inifile:
collected 1 item

test_model.py .                                                          [100%]

=========================== 1 passed in 0.05 seconds ===========================
```

```
In [6]: %%writefile diffusion/model.py
        """  Simplistic 1-dimensional diffusion model """

        def energy(density):
          """ Energy associated with the diffusion model
              :Parameters:
                density: array of positive integers
                   Number of particles at each position i in the array/geometry
          """
          from numpy import array, any, sum

          # Make sure input is an numpy array
          density = array(density)

          # ...of the right kind (integer). Unless it is zero length,
          #    in which case type does not matter.

          if density.dtype.kind != 'i' and len(density) > 0:
            raise TypeError("Density should be a array of *integers*.")
          # and the right values (positive or null)
          if any(density < 0):
            raise ValueError("Density should be an array of *positive* integers.")
          if density.ndim != 1:
            raise ValueError("Density should be an a *1-dimensional*"+
                             "array of positive integers.")

          return sum(density * (density - 1))
```

```
Overwriting diffusion/model.py
```

```
In [7]: %%writefile diffusion/test_model.py
        """ Unit tests for a diffusion model """

        from pytest import raises
        from .model import energy

        def test_energy_fails_on_non_integer_density():
            with raises(TypeError) as exception:
               energy([1.0, 2, 3])
```

```
        def test_energy_fails_on_negative_density():
            with raises(ValueError) as exception: energy(
                    [-1, 2, 3])

        def test_energy_fails_ndimensional_density():
            with raises(ValueError) as exception: energy(
                    [[1, 2, 3], [3, 4, 5]])

        def test_zero_energy_cases():
          # Zero energy at zero density
          densities = [ [], [0], [0, 0, 0] ]
          for density in densities:
            assert energy(density) == 0

        def test_derivative():
          from numpy.random import randint

          # Loop over vectors of different sizes (but not empty)
          for vector_size in randint(1, 1000, size=30):

            # Create random density of size N
            density = randint(50, size=vector_size)

            # will do derivative at this index
            element_index = randint(vector_size)

            # modified densities
            density_plus_one = density.copy()
            density_plus_one[element_index] += 1

            # Compute and check result
            # d(n^2-1)/dn = 2n
            expected = (2.0*density[element_index]
                        if density[element_index] > 0
                        else 0 )
            actual = energy(density_plus_one) - energy(density)
            assert expected == actual

        def test_derivative_no_self_energy():
          """ If particle is alone, then its participation to energy is zero """
          from numpy import array

          density = array([1, 0, 1, 10, 15, 0])
          density_plus_one = density.copy()
          density[1] += 1

          expected = 0
          actual = energy(density_plus_one) - energy(density)
          assert expected == actual
```

```
Overwriting diffusion/test_model.py
```

In [8]:
```
%%bash
cd diffusion
```

```
        py.test

============================= test session starts ==============================
platform darwin -- Python 3.7.2, pytest-4.1.1, py-1.7.0, pluggy-0.8.0
rootdir: /Users/edaub/Projects/rsd-engineeringcourse/ch03tests/diffusion, inifile:
collected 6 items

test_model.py ...                                                        [100%]

========================== 6 passed in 0.20 seconds ===========================


In [9]: %%bash
        cd diffusion
        py.test --cov

usage: py.test [options] [file_or_dir] [file_or_dir] [...]
py.test: error: unrecognized arguments: --cov
  inifile: None
  rootdir: /Users/edaub/Projects/rsd-engineeringcourse/ch03tests/diffusion



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-9-51e5cbfcce6f> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'cd diffusion\npy.test --cov\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshe
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
    --> 187         call = lambda f, *a, **k: f(*a, **k)
        188
```

```
        189         if callable(arg):


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        243             sys.stderr.flush()
        244         if args.raise_error and p.returncode!=0:
    --> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
        246
        247     def _run_script(self, p, cell, to_close):


        CalledProcessError: Command 'b'cd diffusion\npy.test --cov\n'' returned non-zero exit status 4.
```

```
In [10]: %%bash
         cd diffusion
         py.test --cov --cov-report html
```

```
usage: py.test [options] [file_or_dir] [file_or_dir] [...]
py.test: error: unrecognized arguments: --cov --cov-report
  inifile: None
  rootdir: /Users/edaub/Projects/rsd-engineeringcourse/ch03tests/diffusion



        ---------------------------------------------------------------------------

        CalledProcessError                        Traceback (most recent call last)

        <ipython-input-10-553699036005> in <module>
    ----> 1 get_ipython().run_cell_magic('bash', '', 'cd diffusion\npy.test --cov --cov-report html\n')


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/interactiveshel
       2321             magic_arg_s = self.var_expand(line, stack_depth)
       2322             with self.builtin_trap:
    -> 2323                 result = fn(magic_arg_s, cell)
       2324             return result
       2325


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
        140             else:
        141                 line = script
    --> 142             return self.shebang(line, cell)
        143
        144         # write a basic docstring:


        <decorator-gen-109> in shebang(self, line, cell)


        /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magic.py in <la
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
```

```
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /usr/local/Cellar/ipython/7.2.0/libexec/lib/python3.7/site-packages/IPython/core/magics/script.p
    243             sys.stderr.flush()
    244         if args.raise_error and p.returncode!=0:
--> 245             raise CalledProcessError(p.returncode, cell, output=out, stderr=err)
    246
    247     def _run_script(self, p, cell, to_close):


    CalledProcessError: Command 'b'cd diffusion\npy.test --cov --cov-report html\n'' returned non-z
```

```
In [1]: from unittest.mock import Mock
        function = Mock(name="myroutine", return_value=2)

In [2]: function(1)

Out[2]: 2

In [3]: function(5, "hello", a=True)

Out[3]: 2

In [4]: function.mock_calls

Out[4]: [call(1), call(5, 'hello', a=True)]

In [5]: name, args, kwargs = function.mock_calls[1]
        args, kwargs

Out[5]: ((5, 'hello'), {'a': True})

In [6]: function = Mock(name="myroutine", side_effect=[2, "xyz"])

In [7]: function(1)

Out[7]: 2

In [8]: function(1, "hello", {'a': True})

Out[8]: 'xyz'

In [9]: function()
```

```
    ---------------------------------------------------------------------------

    StopIteration                             Traceback (most recent call last)

    <ipython-input-9-30ca0b4348da> in <module>
----> 1 function()


    /usr/local/Cellar/python/3.7.2_1/Frameworks/Python.framework/Versions/3.7/lib/python3.7/unittes
```