Data science fundamentals
Lesson 8

Hay Kranen

Wednesday October 17th, 2018

Today's programme

Recap

Temperatures notebook

Solving bugs

Break

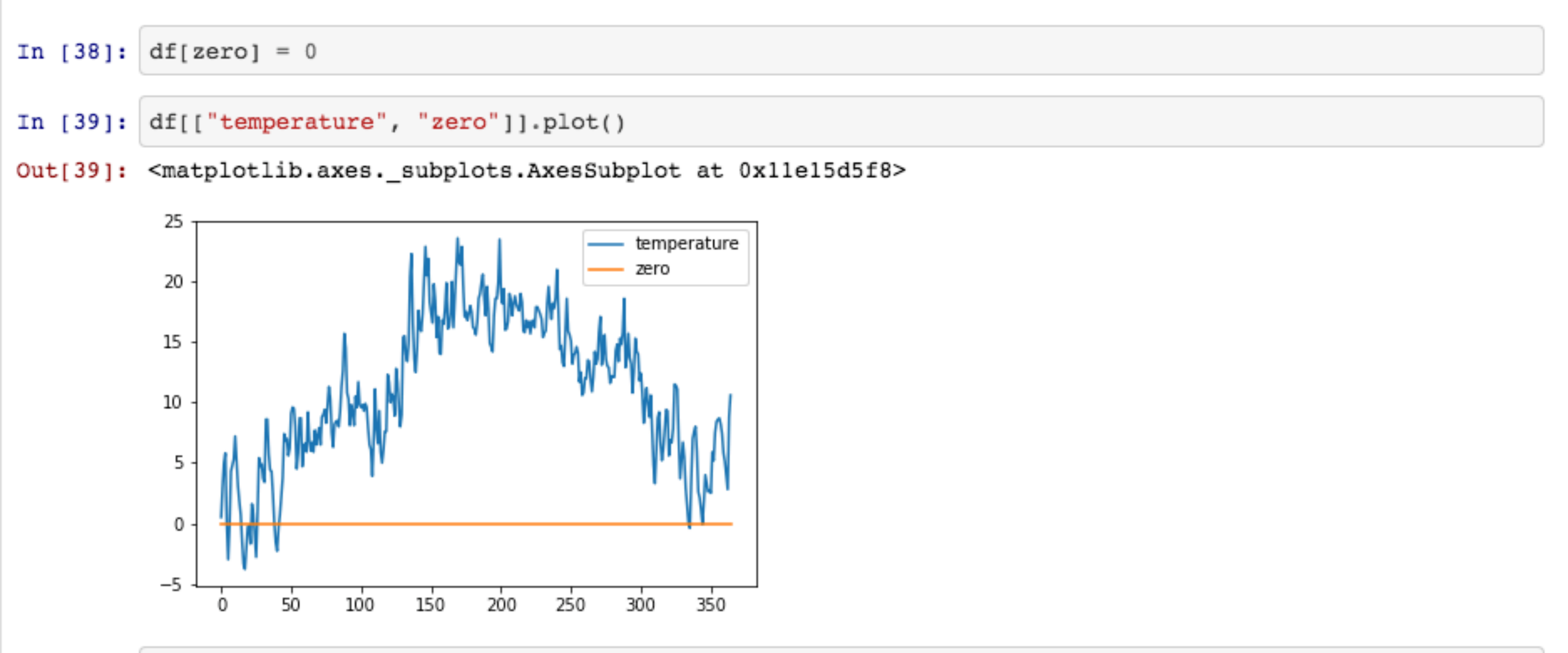
Reddit API

Plotting with Pandas

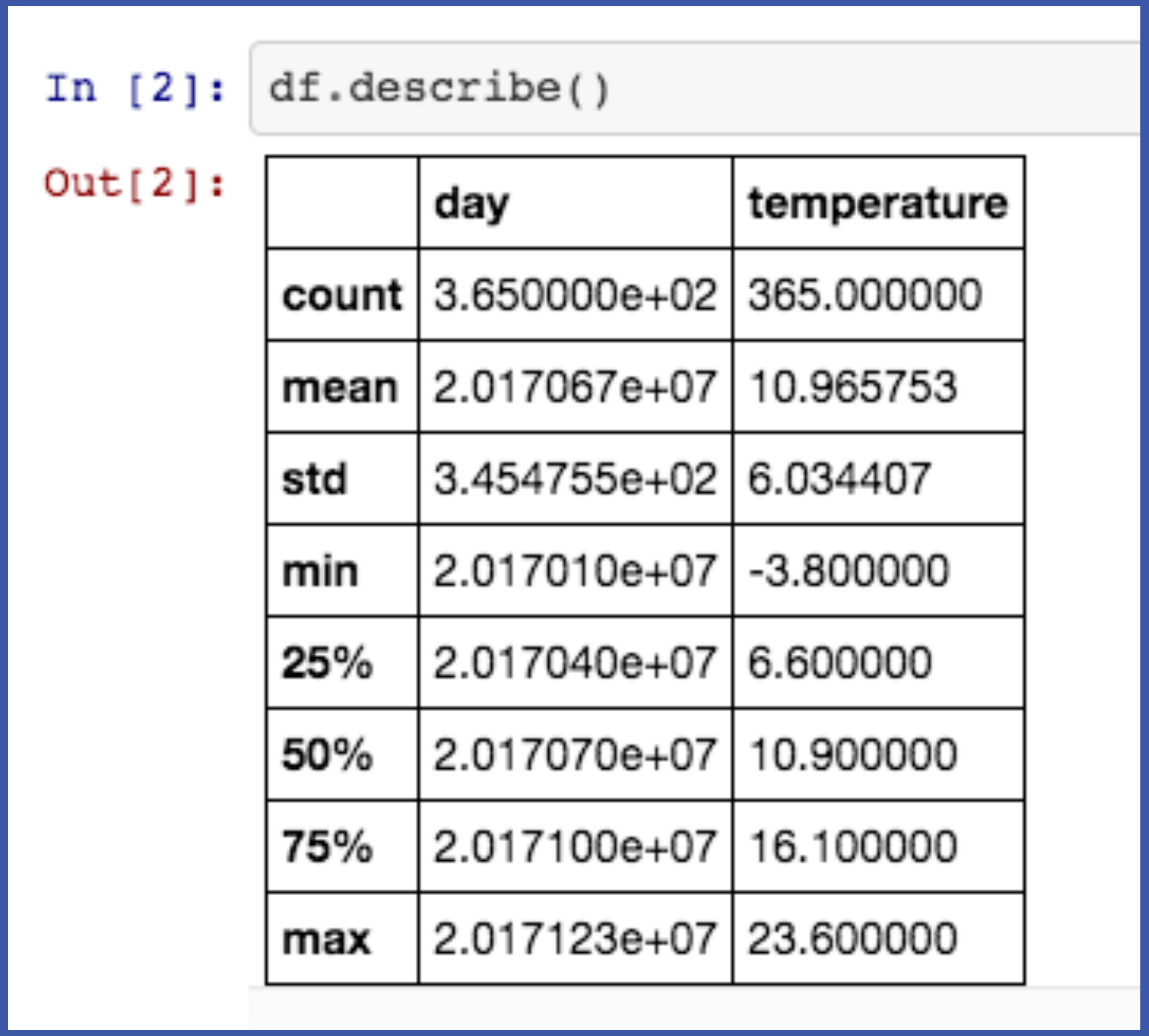
Lunch break

Refactor

RECAP



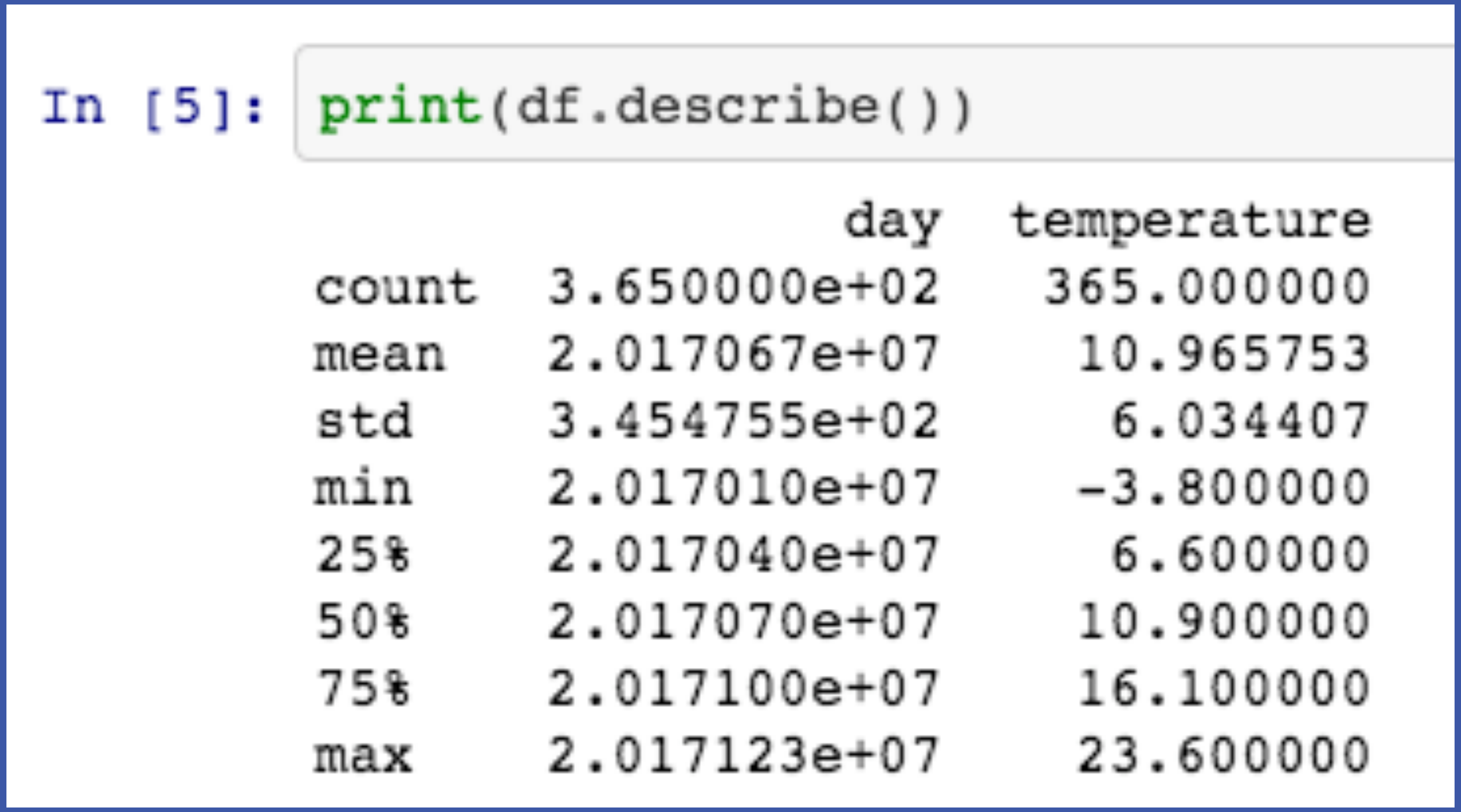
```
In [38]: df[zero] = 0

In [39]: df[["temperature", "zero"]].plot()

Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x11e15d5f8>
```



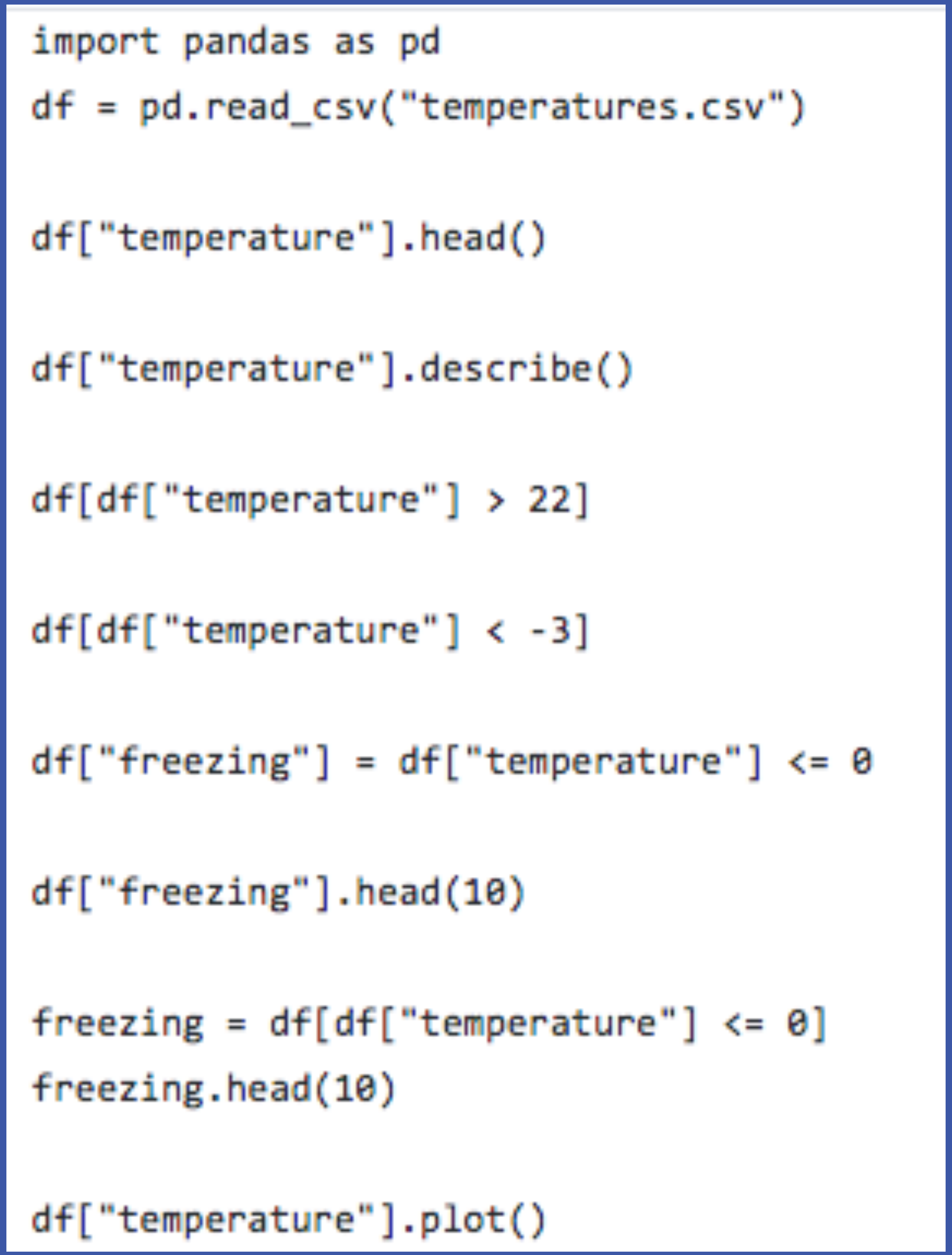
```
In [2]: df.describe()
```

Out[2]:

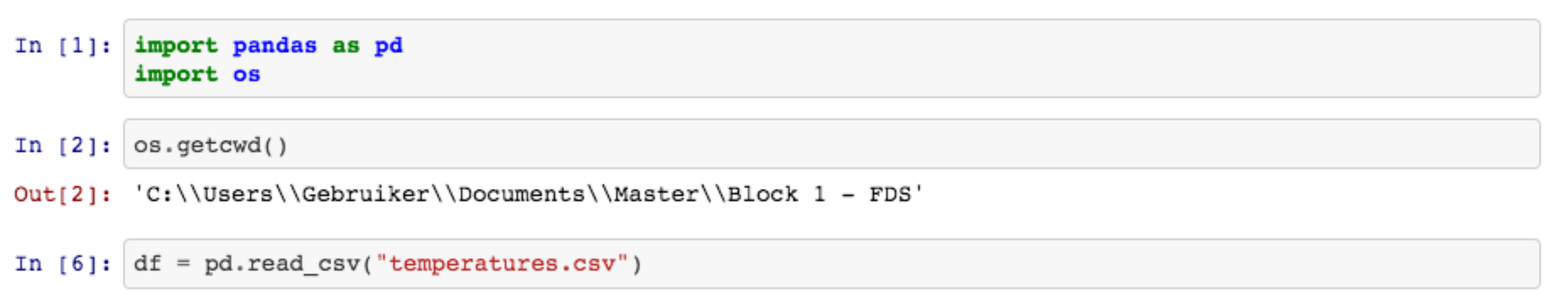
	day	temperature
count	3.650000e+02	365.000000
mean	2.017067e+07	10.965753
std	3.454755e+02	6.034407
min	2.017010e+07	-3.800000
25%	2.017040e+07	6.600000
50%	2.017070e+07	10.900000
75%	2.017100e+07	16.100000
max	2.017123e+07	23.600000



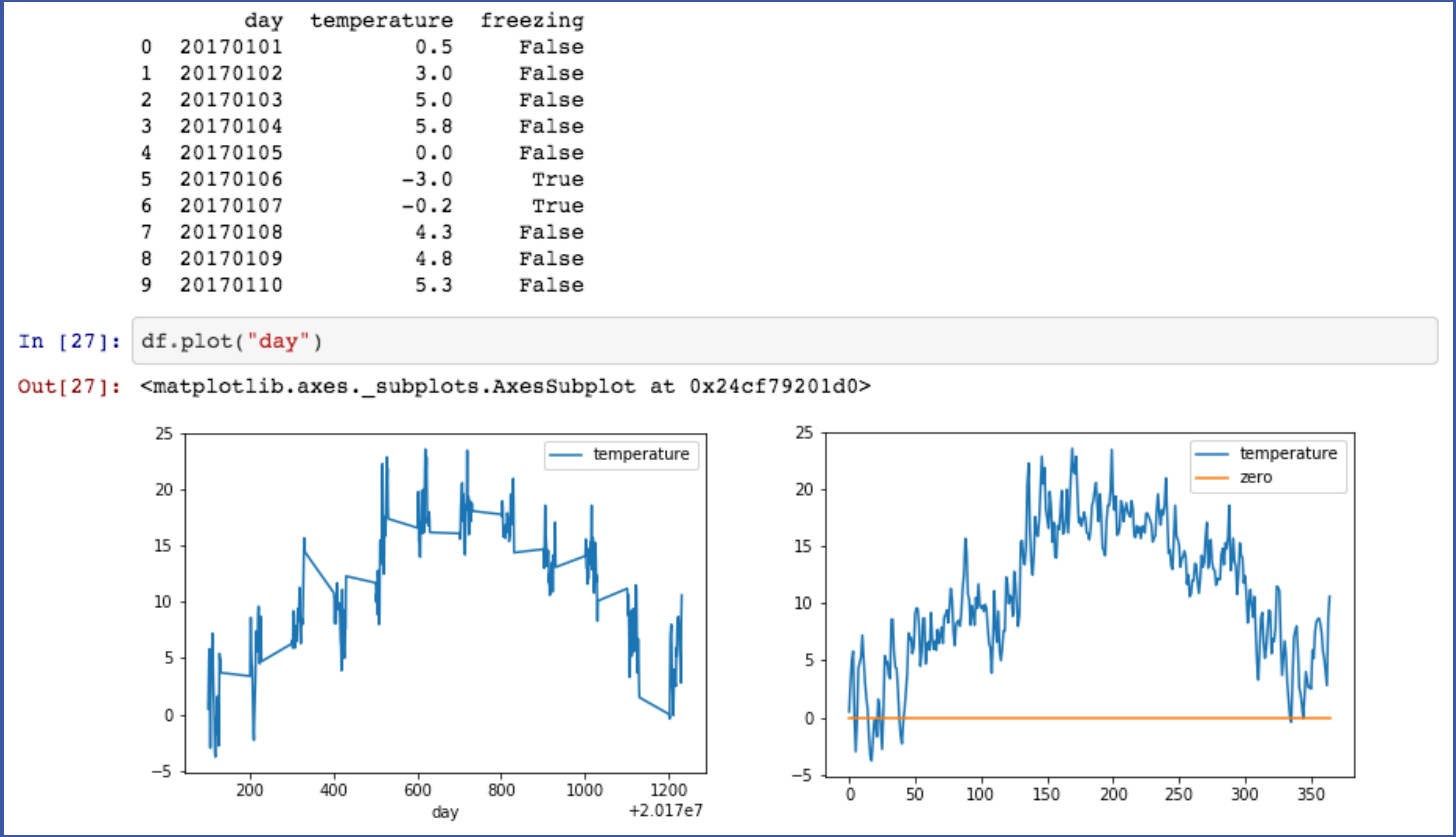
```
In [5]: print(df.describe())

                day  temperature
count  3.650000e+02   365.000000
mean   2.017067e+07    10.965753
std    3.454755e+02     6.034407
min    2.017010e+07    -3.800000
25%    2.017040e+07     6.600000
50%    2.017070e+07    10.900000
75%    2.017100e+07    16.100000
max    2.017123e+07    23.600000
```



```
import pandas as pd
df = pd.read_csv("temperatures.csv")

df["temperature"].head()

df["temperature"].describe()

df[df["temperature"] > 22]

df[df["temperature"] < -3]

df["freezing"] = df["temperature"] <= 0

df["freezing"].head(10)

freezing = df[df["temperature"] <= 0]
freezing.head(10)

df["temperature"].plot()
```



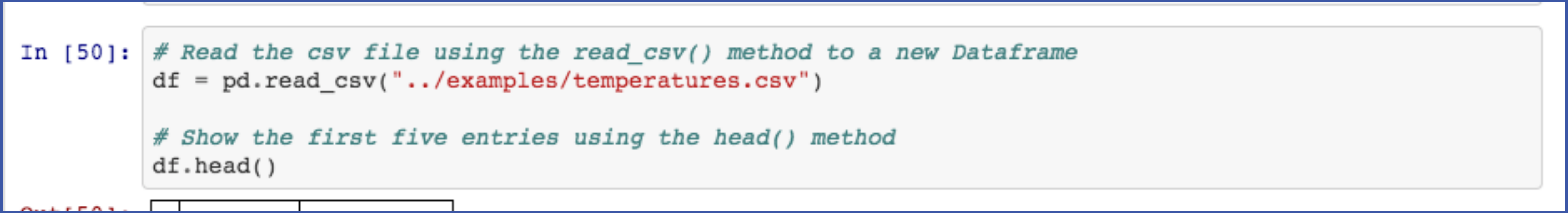
```
In [1]: import pandas as pd
        import os

In [2]: os.getcwd()

Out[2]: 'C:\\Users\\Gebruiker\\Documents\\Master\\Block 1 - FDS'

In [6]: df = pd.read_csv("temperatures.csv")
```



```
        day  temperature  freezing
0  20170101          0.5     False
1  20170102          3.0     False
2  20170103          5.0     False
3  20170104          5.8     False
4  20170105          0.0     False
5  20170106         -3.0      True
6  20170107         -0.2      True
7  20170108          4.3     False
8  20170109          4.8     False
9  20170110          5.3     False
```

```
In [27]: df.plot("day")
```

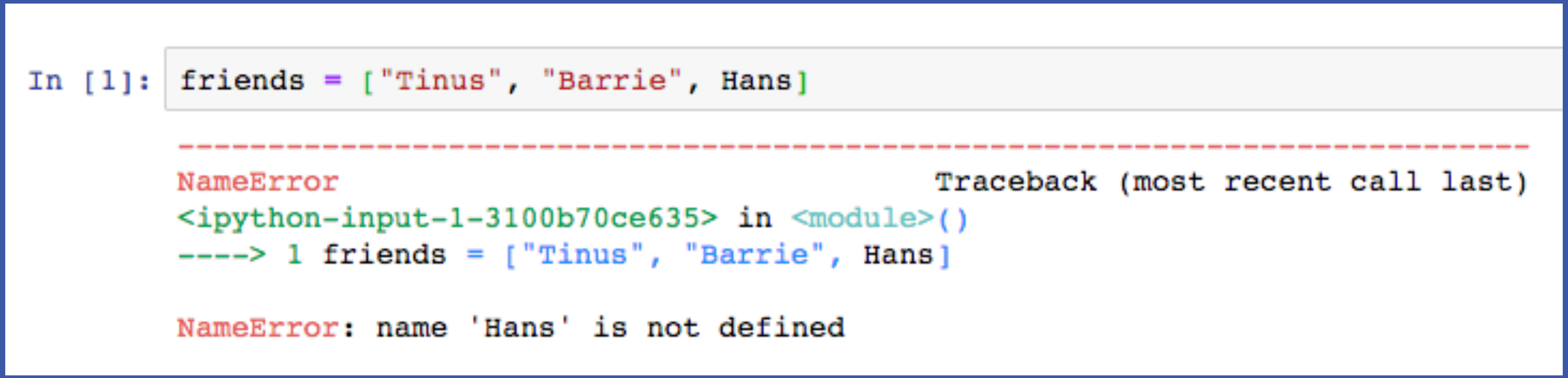
```
Out[27]: <matplotlib.axes._subplots.AxesSubplot at 0x24cf79201d0>
```



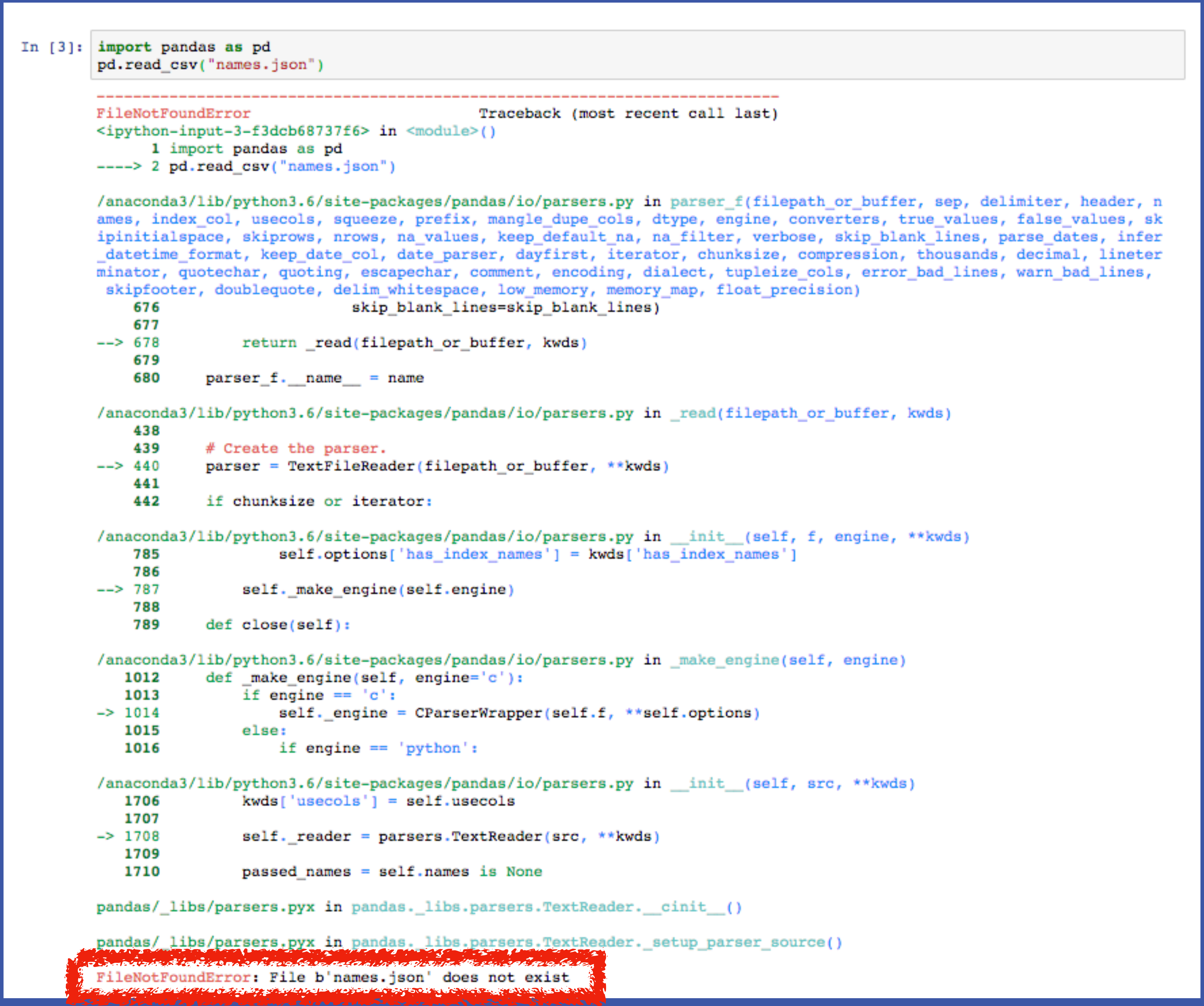
```
In [50]: # Read the csv file using the read_csv() method to a new Dataframe
         df = pd.read_csv("../examples/temperatures.csv")

         # Show the first five entries using the head() method
         df.head()
```

Solving Bugs

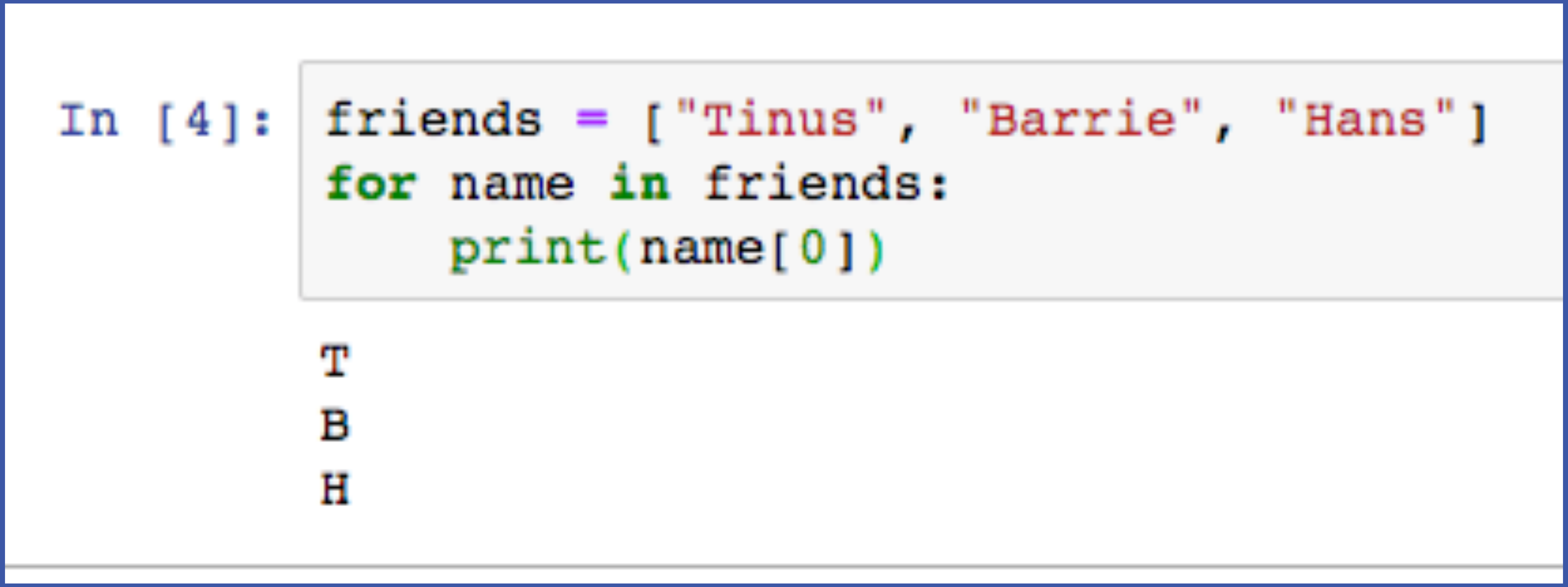
1. Is it an error or unintended behaviour?



```
In [1]: friends = ["Tinus", "Barrie", Hans]

        ---------------------------------------------------------------------------
        NameError                                 Traceback (most recent call last)
        <ipython-input-1-3100b70ce635> in <module>()
        ----> 1 friends = ["Tinus", "Barrie", Hans]

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



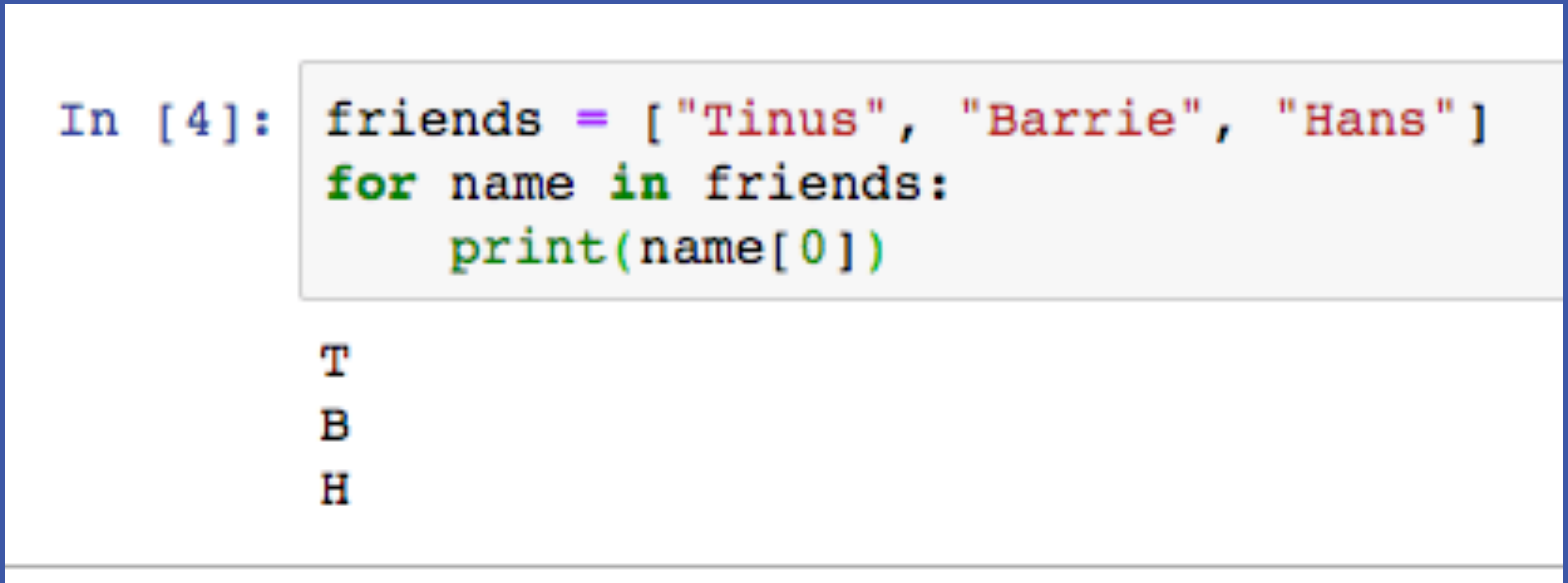
```
In [3]: import pandas as pd
        pd.read_csv("names.json")

        ---------------------------------------------------------------------------
        FileNotFoundError                         Traceback (most recent call last)
        <ipython-input-3-f3dcb68737f6> in <module>()
              1 import pandas as pd
        ----> 2 pd.read_csv("names.json")

        /anaconda3/lib/python3.6/site-packages/pandas/io/parsers.py in parser_f(filepath_or_buffer, sep, delimiter, header, n
        ames, index_col, usecols, squeeze, prefix, mangle_dupe_cols, dtype, engine, converters, true_values, false_values, sk
        ipinitialspace, skiprows, nrows, na_values, keep_default_na, na_filter, verbose, skip_blank_lines, parse_dates, infer
        _datetime_format, keep_date_col, date_parser, dayfirst, iterator, chunksize, compression, thousands, decimal, lineter
        minator, quotechar, quoting, escapechar, comment, encoding, dialect, tupleize_cols, error_bad_lines, warn_bad_lines,
         skipfooter, doublequote, delim_whitespace, low_memory, memory_map, float_precision)
            676                     skip_blank_lines=skip_blank_lines)
            677 
        --> 678         return _read(filepath_or_buffer, kwds)
            679 
            680     parser_f.__name__ = name

        /anaconda3/lib/python3.6/site-packages/pandas/io/parsers.py in _read(filepath_or_buffer, kwds)
            438 
            439     # Create the parser.
        --> 440     parser = TextFileReader(filepath_or_buffer, **kwds)
            441 
            442     if chunksize or iterator:

        /anaconda3/lib/python3.6/site-packages/pandas/io/parsers.py in __init__(self, f, engine, **kwds)
            785             self.options['has_index_names'] = kwds['has_index_names']
            786 
        --> 787         self._make_engine(self.engine)
            788 
            789     def close(self):

        /anaconda3/lib/python3.6/site-packages/pandas/io/parsers.py in _make_engine(self, engine)
           1012     def _make_engine(self, engine='c'):
           1013         if engine == 'c':
        -> 1014             self._engine = CParserWrapper(self.f, **self.options)
           1015         else:
           1016             if engine == 'python':

        /anaconda3/lib/python3.6/site-packages/pandas/io/parsers.py in __init__(self, src, **kwds)
           1706         kwds['usecols'] = self.usecols
           1707 
        -> 1708         self._reader = parsers.TextReader(src, **kwds)
           1709 
           1710         passed_names = self.names is None

        pandas/_libs/parsers.pyx in pandas._libs.parsers.TextReader.__cinit__()

        pandas/_libs/parsers.pyx in pandas._libs.parsers.TextReader._setup_parser_source()

        FileNotFoundError: File b'names.json' does not exist
```



```python
friends = ["Tinus", "Barrie", "Hans"]
for name in friends:
    print(name[0])
```

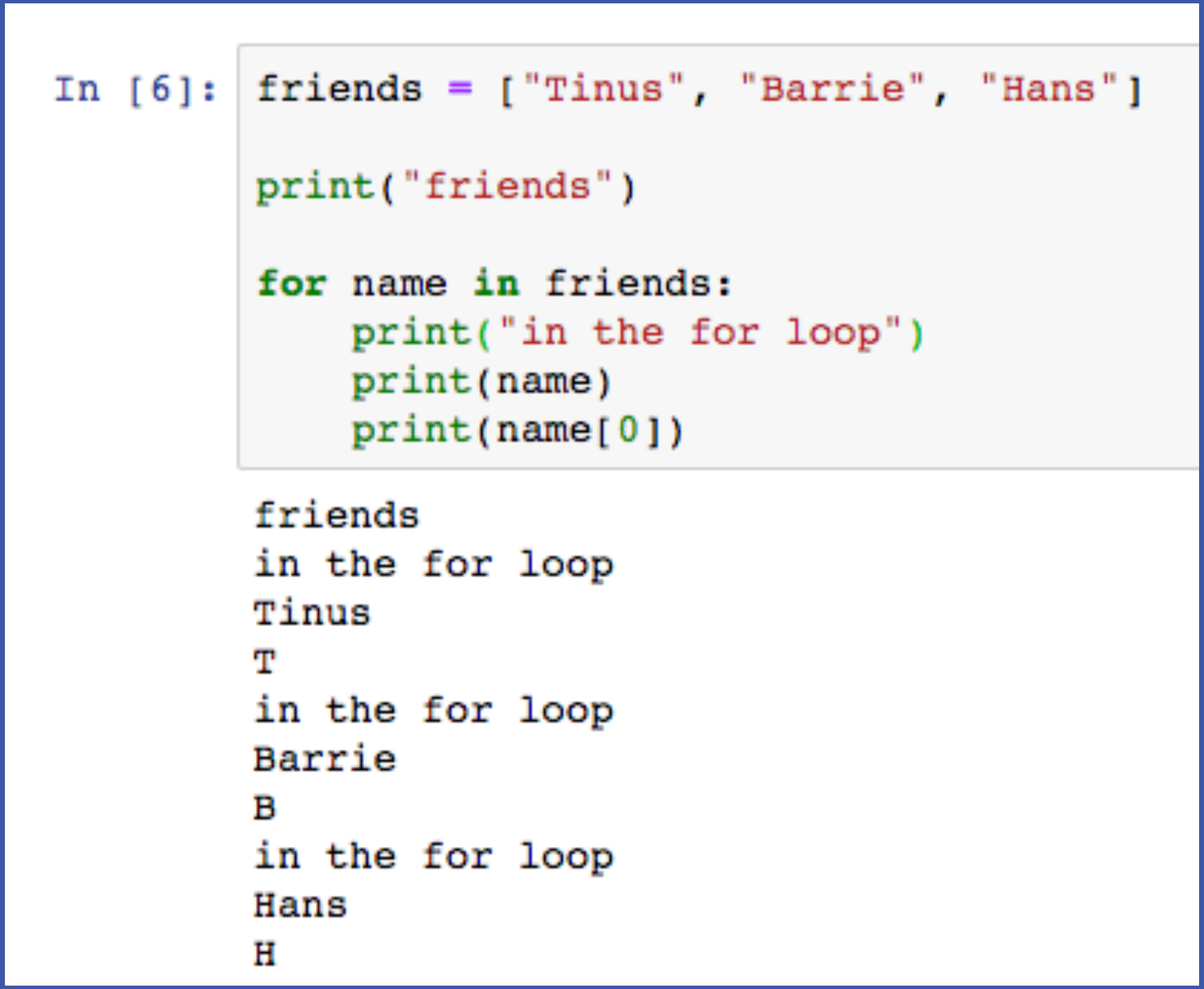
```
T
B
H
```

2. Use print() to debug



```python
friends = ["Tinus", "Barrie", "Hans"]
for name in friends:
    print(name[0])
```

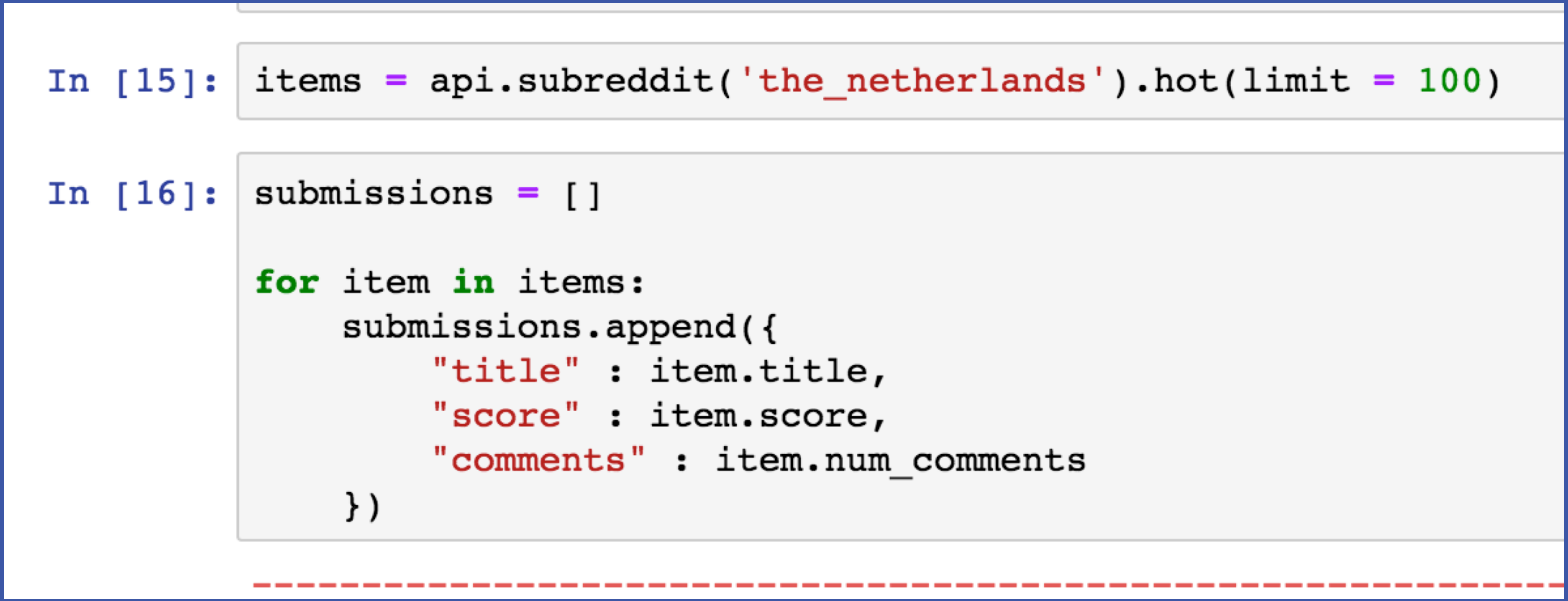
```
T
B
H
```



```python
friends = ["Tinus", "Barrie", "Hans"]

print("friends")

for name in friends:
    print("in the for loop")
    print(name)
    print(name[0])
```

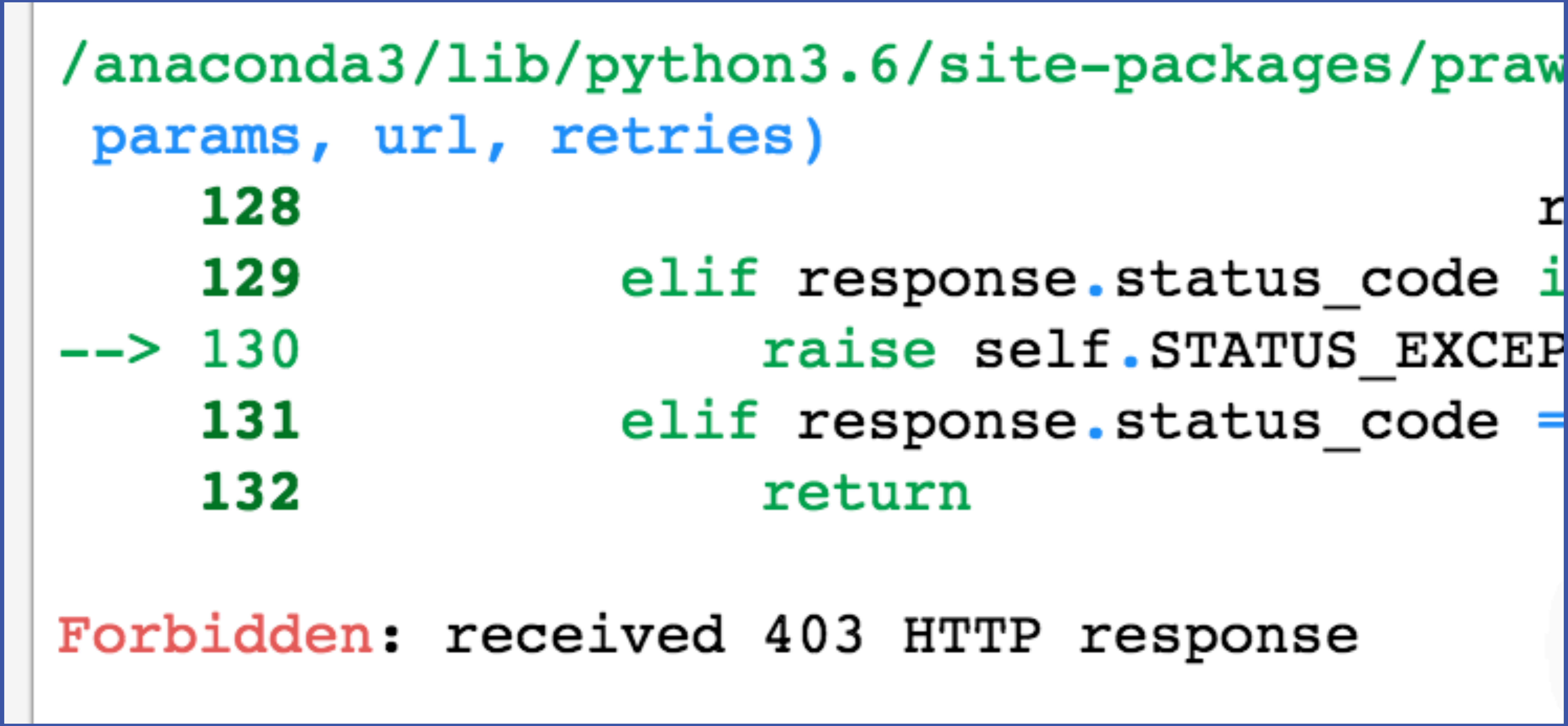
```
friends
in the for loop
Tinus
T
in the for loop
Barrie
B
in the for loop
Hans
H
```

3. Simplify the problem

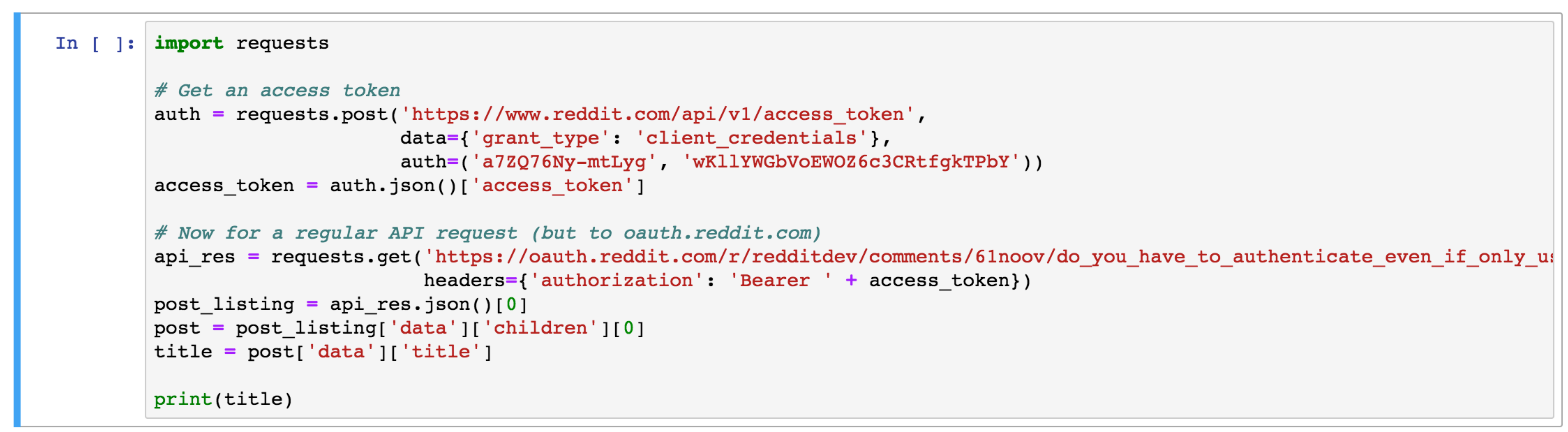


```
In [15]: items = api.subreddit('the_netherlands').hot(limit = 100)
```

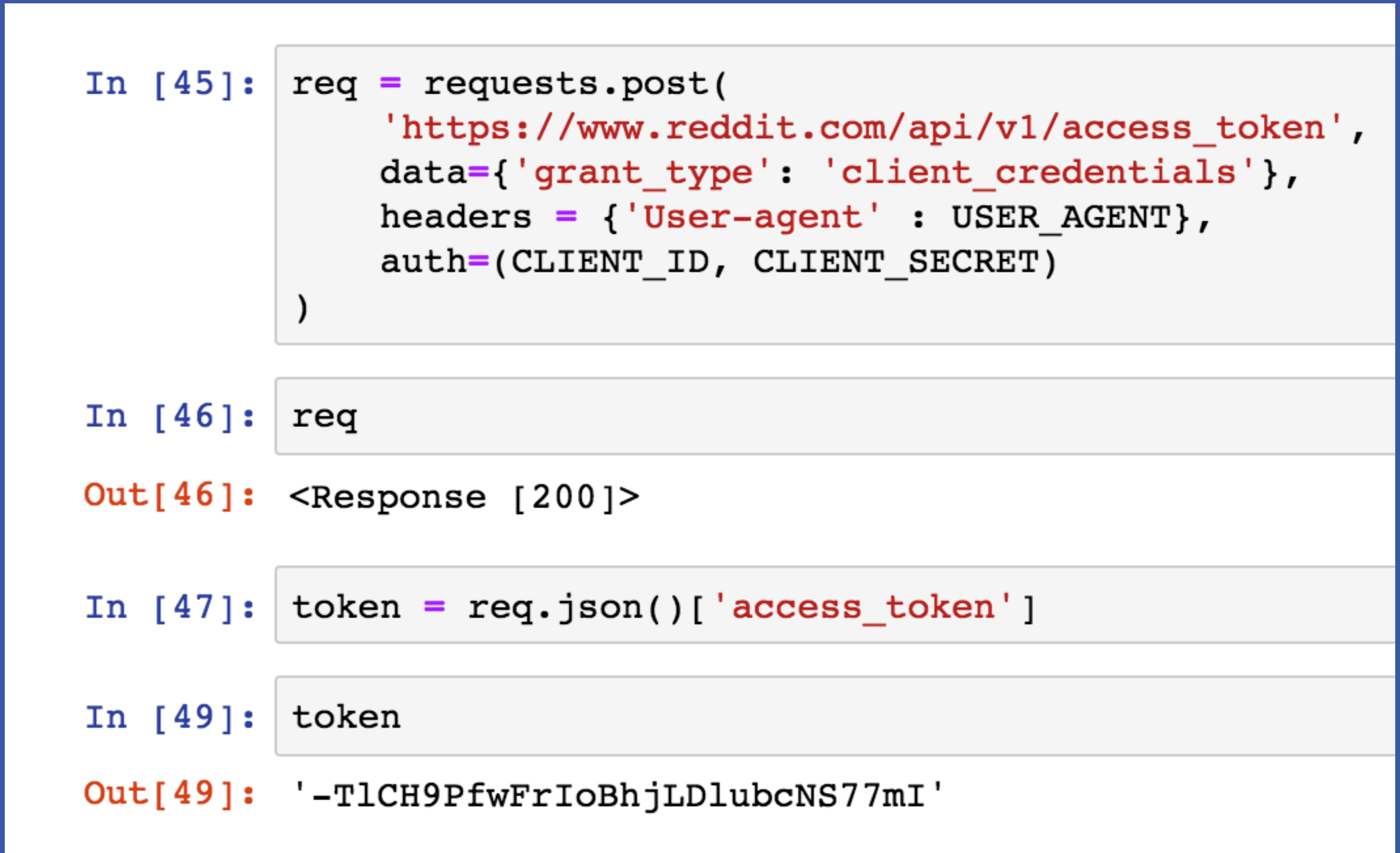
```
In [16]: submissions = []

         for item in items:
             submissions.append({
                 "title" : item.title,
                 "score" : item.score,
                 "comments" : item.num_comments
             })
```



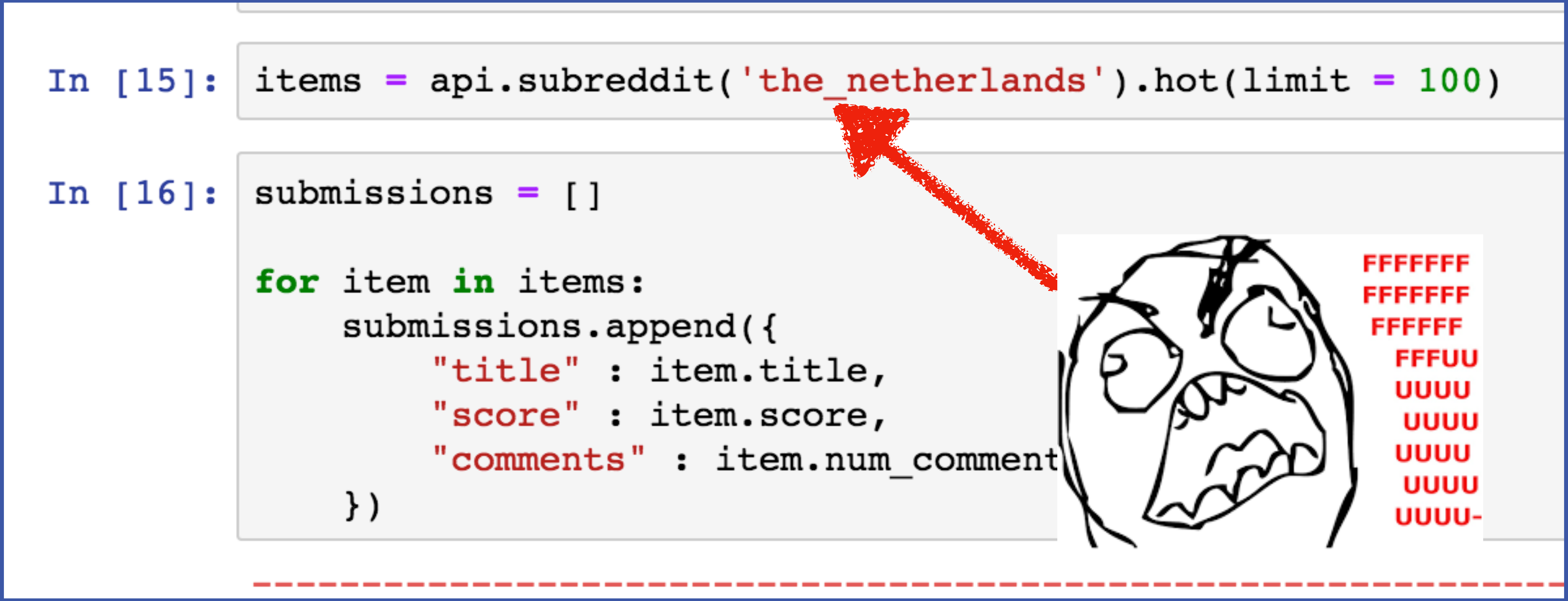
```
/anaconda3/lib/python3.6/site-packages/praw
 params, url, retries)
    128                                       r
    129             elif response.status_code i
--> 130                 raise self.STATUS_EXCEP
    131             elif response.status_code =
    132                 return

Forbidden: received 403 HTTP response
```



```
In [ ]: import requests

        # Get an access token
        auth = requests.post('https://www.reddit.com/api/v1/access_token',
                             data={'grant_type': 'client_credentials'},
                             auth=('a7ZQ76Ny-mtLyg', 'wKllYWGbVoEWOZ6c3CRtfgkTPbY'))
        access_token = auth.json()['access_token']

        # Now for a regular API request (but to oauth.reddit.com)
        api_res = requests.get('https://oauth.reddit.com/r/redditdev/comments/61noov/do_you_have_to_authenticate_even_if_only_us
                               headers={'authorization': 'Bearer ' + access_token})
        post_listing = api_res.json()[0]
        post = post_listing['data']['children'][0]
        title = post['data']['title']

        print(title)
```

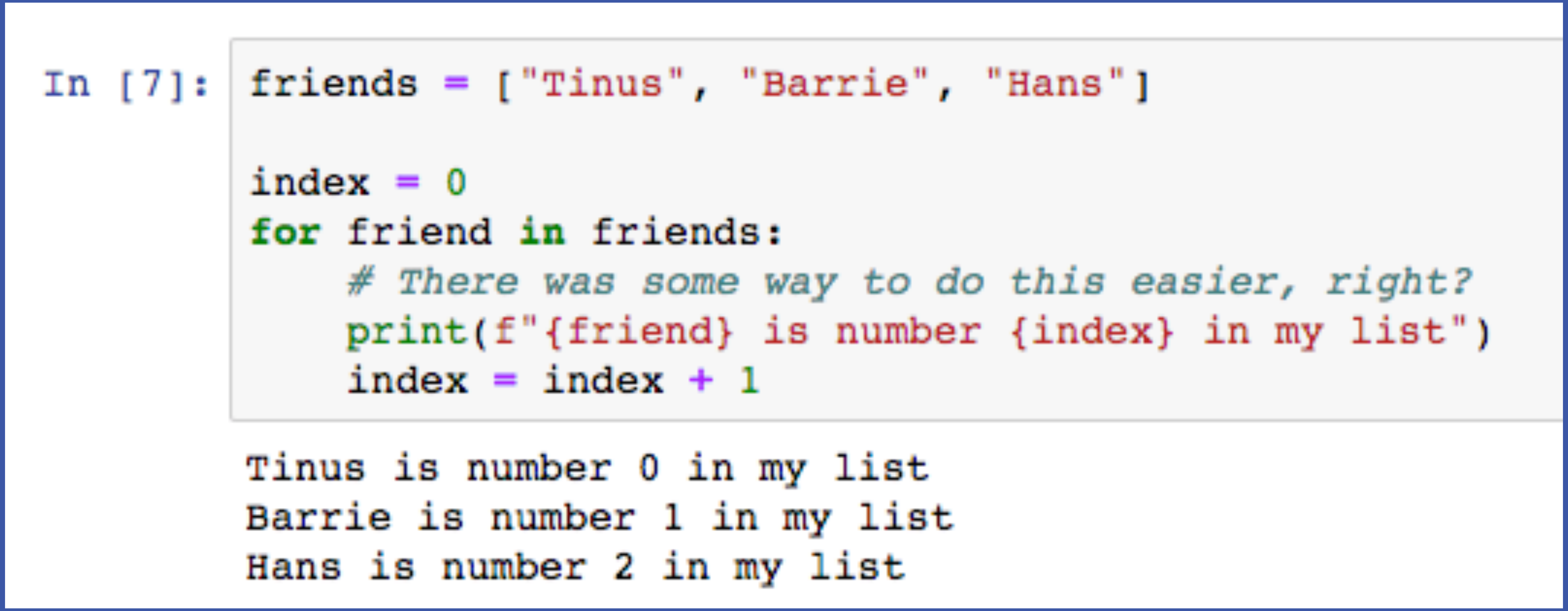


```
In [45]: req = requests.post(
             'https://www.reddit.com/api/v1/access_token',
             data={'grant_type': 'client_credentials'},
             headers = {'User-agent' : USER_AGENT},
             auth=(CLIENT_ID, CLIENT_SECRET)
         )

In [46]: req

Out[46]: <Response [200]>

In [47]: token = req.json()['access_token']

In [49]: token

Out[49]: '-TlCH9PfwFrIoBhjLDlubcNS77mI'
```



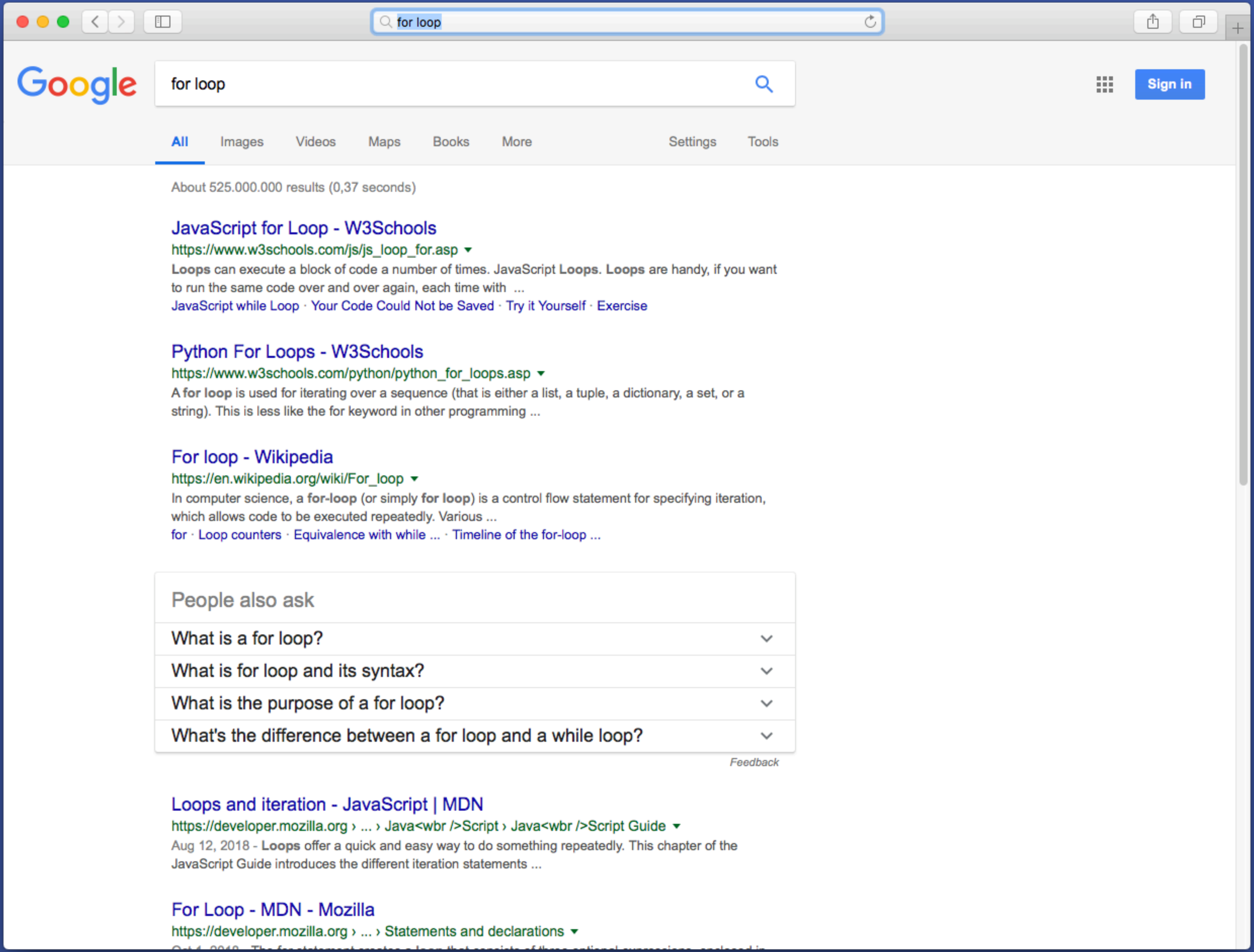
```
In [15]: items = api.subreddit('the_netherlands').hot(limit = 100)

In [16]: submissions = []

         for item in items:
             submissions.append({
                 "title" : item.title,
                 "score" : item.score,
                 "comments" : item.num_comment
             })
```

4. Use Google / Stack Overflow



```python
friends = ["Tinus", "Barrie", "Hans"]

index = 0
for friend in friends:
    # There was some way to do this easier, right?
    print(f"{friend} is number {index} in my list")
    index = index + 1
```

```
Tinus is number 0 in my list
Barrie is number 1 in my list
Hans is number 2 in my list
```



About 525.000.000 results (0,37 seconds)

JavaScript for Loop - W3Schools
https://www.w3schools.com/js/js_loop_for.asp
Loops can execute a block of code a number of times. JavaScript Loops. Loops are handy, if you want to run the same code over and over again, each time with ...
JavaScript while Loop · Your Code Could Not be Saved · Try it Yourself · Exercise

Python For Loops - W3Schools
https://www.w3schools.com/python/python_for_loops.asp
A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). This is less like the for keyword in other programming ...

For loop - Wikipedia
https://en.wikipedia.org/wiki/For_loop
In computer science, a for-loop (or simply for loop) is a control flow statement for specifying iteration, which allows code to be executed repeatedly. Various ...
for · Loop counters · Equivalence with while ... · Timeline of the for-loop ...

People also ask

What is a for loop?

What is for loop and its syntax?

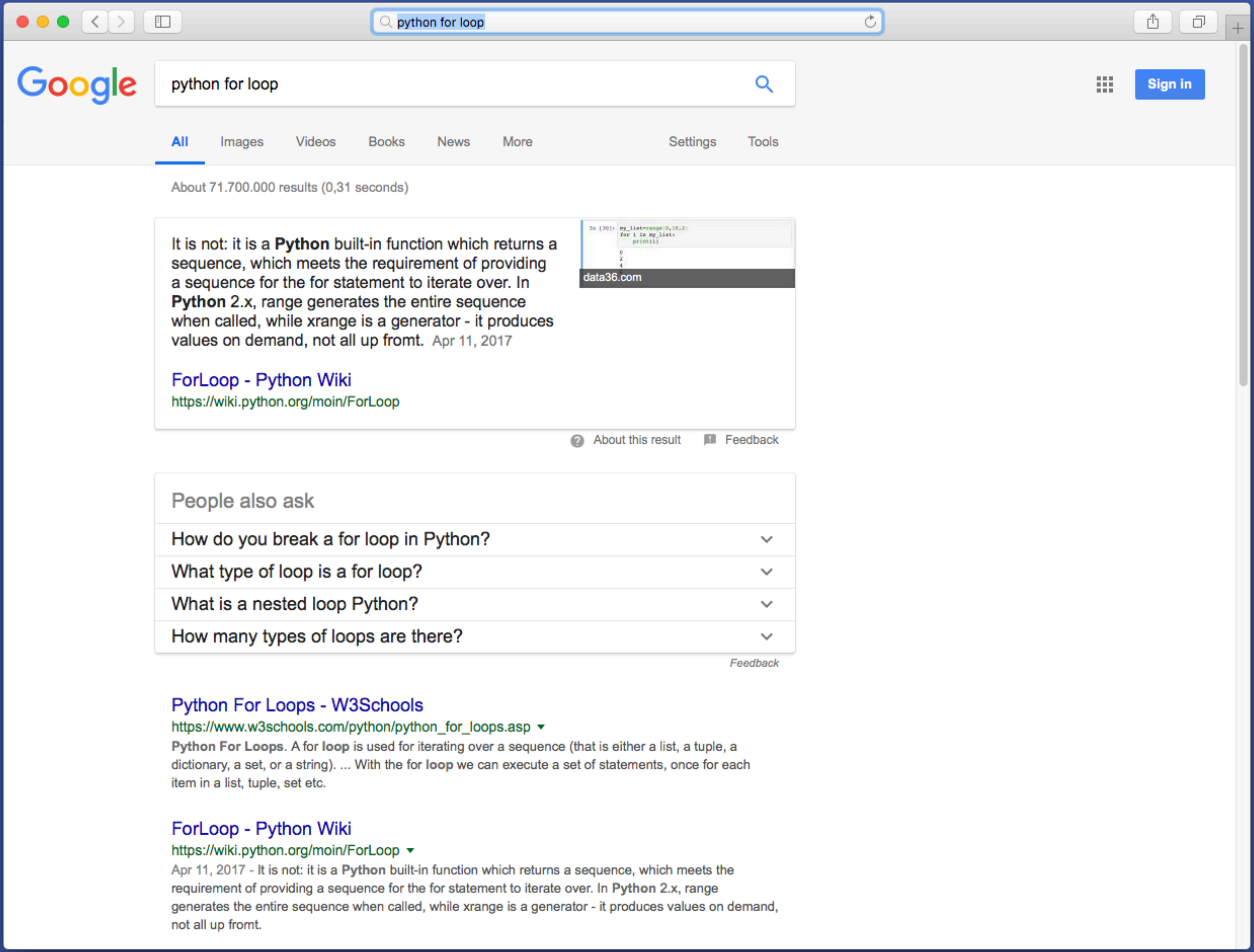
What is the purpose of a for loop?

What's the difference between a for loop and a while loop?

Feedback

Loops and iteration - JavaScript | MDN
https://developer.mozilla.org › ... › Java<wbr />Script › Java<wbr />Script Guide
Aug 12, 2018 - Loops offer a quick and easy way to do something repeatedly. This chapter of the JavaScript Guide introduces the different iteration statements ...

For Loop - MDN - Mozilla
https://developer.mozilla.org › ... › Statements and declarations



About 71.700.000 results (0,31 seconds)

It is not: it is a **Python** built-in function which returns a sequence, which meets the requirement of providing a sequence for the for statement to iterate over. In **Python** 2.x, range generates the entire sequence when called, while xrange is a generator - it produces values on demand, not all up fromt. Apr 11, 2017

data36.com

ForLoop - Python Wiki
https://wiki.python.org/moin/ForLoop

People also ask

How do you break a for loop in Python?

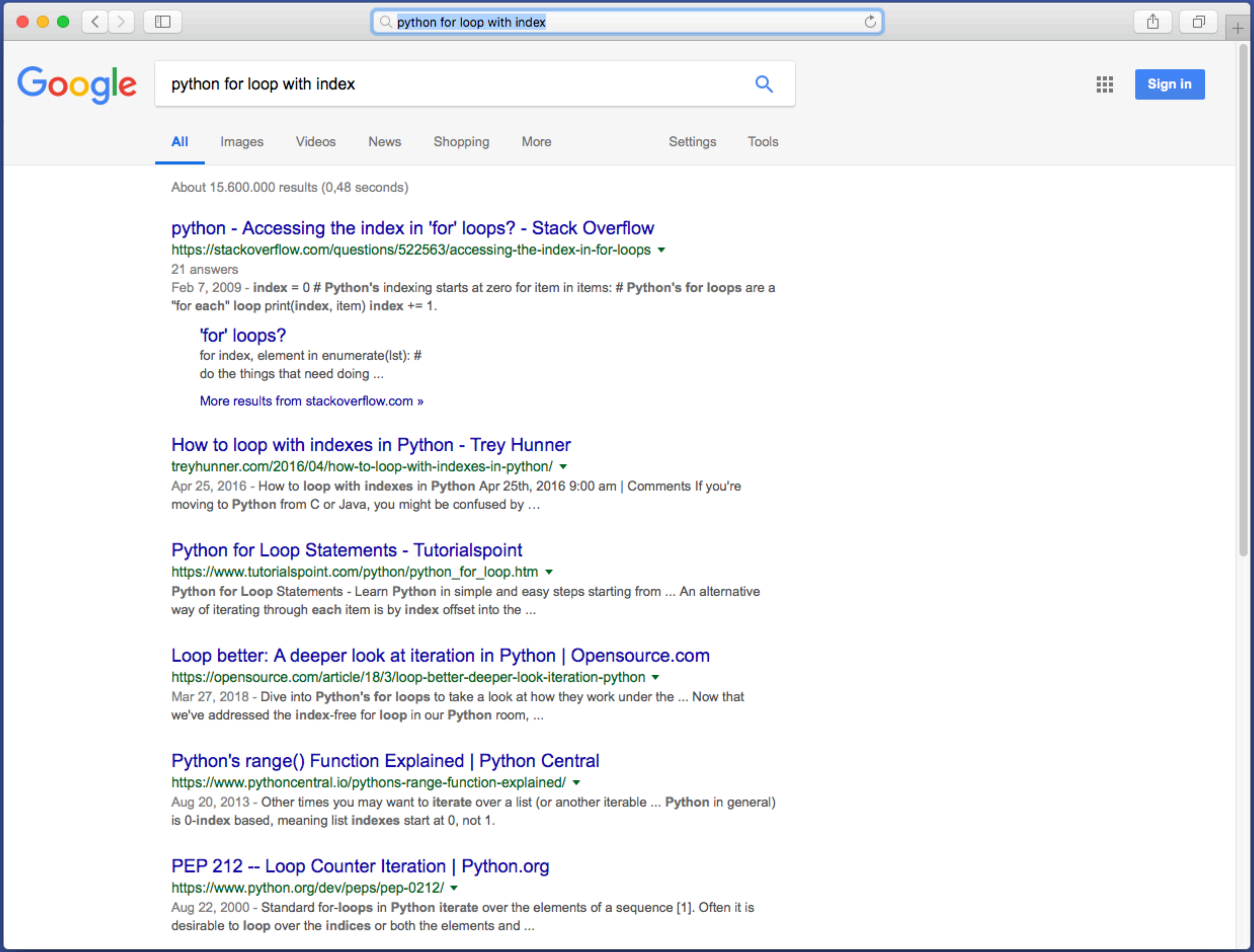
What type of loop is a for loop?

What is a nested loop Python?

How many types of loops are there?

Python For Loops - W3Schools
https://www.w3schools.com/python/python_for_loops.asp ▾
Python For Loops. A for **loop** is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). ... With the for **loop** we can execute a set of statements, once for each item in a list, tuple, set etc.

ForLoop - Python Wiki
https://wiki.python.org/moin/ForLoop ▾
Apr 11, 2017 - It is not: it is a **Python** built-in function which returns a sequence, which meets the requirement of providing a sequence for the for statement to iterate over. In **Python** 2.x, range generates the entire sequence when called, while xrange is a generator - it produces values on demand, not all up fromt.



About 15.600.000 results (0,48 seconds)

python - Accessing the index in 'for' loops? - Stack Overflow
https://stackoverflow.com/questions/522563/accessing-the-index-in-for-loops ▾
21 answers
Feb 7, 2009 - **index** = 0 # **Python's** indexing starts at zero for item in items: # **Python's for loops** are a "for **each**" **loop** print(**index**, item) **index** += 1.

'for' loops?
for index, element in enumerate(lst): #
do the things that need doing ...

More results from stackoverflow.com »

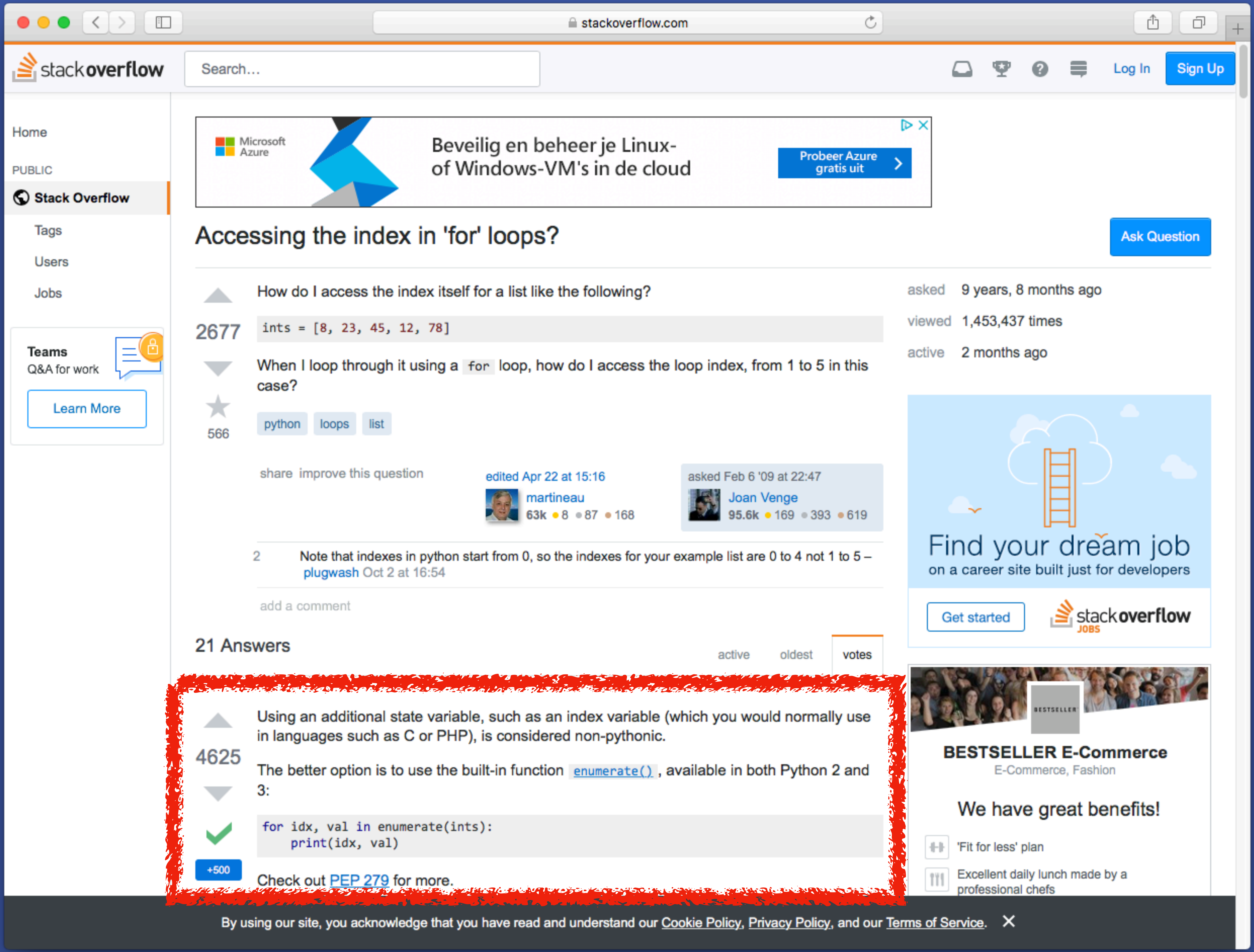
How to loop with indexes in Python - Trey Hunner
treyhunner.com/2016/04/how-to-loop-with-indexes-in-python/ ▾
Apr 25, 2016 - How to **loop with indexes** in **Python** Apr 25th, 2016 9:00 am | Comments If you're moving to **Python** from C or Java, you might be confused by ...

Python for Loop Statements - Tutorialspoint
https://www.tutorialspoint.com/python/python_for_loop.htm ▾
Python for Loop Statements - Learn **Python** in simple and easy steps starting from ... An alternative way of iterating through **each** item is by **index** offset into the ...

Loop better: A deeper look at iteration in Python | Opensource.com
https://opensource.com/article/18/3/loop-better-deeper-look-iteration-python ▾
Mar 27, 2018 - Dive into **Python's for loops** to take a look at how they work under the ... Now that we've addressed the **index**-free for **loop** in our **Python** room, ...

Python's range() Function Explained | Python Central
https://www.pythoncentral.io/pythons-range-function-explained/ ▾
Aug 20, 2013 - Other times you may want to **iterate** over a list (or another iterable ... **Python** in general) is 0-**index** based, meaning list **indexes** start at 0, not 1.

PEP 212 -- Loop Counter Iteration | Python.org
https://www.python.org/dev/peps/pep-0212/ ▾
Aug 22, 2000 - Standard for-**loops** in **Python iterate** over the elements of a sequence [1]. Often it is desirable to **loop** over the **indices** or both the elements and ...



Accessing the index in 'for' loops?

2677

How do I access the index itself for a list like the following?

```
ints = [8, 23, 45, 12, 78]
```

When I loop through it using a `for` loop, how do I access the loop index, from 1 to 5 in this case?

566

python loops list

share improve this question

edited Apr 22 at 15:16
martineau
63k ●8 ●87 ●168

asked Feb 6 '09 at 22:47
Joan Venge
95.6k ●169 ●393 ●619

2 Note that indexes in python start from 0, so the indexes for your example list are 0 to 4 not 1 to 5 – plugwash Oct 2 at 16:54

add a comment

asked 9 years, 8 months ago
viewed 1,453,437 times
active 2 months ago

21 Answers

active oldest votes

4625

Using an additional state variable, such as an index variable (which you would normally use in languages such as C or PHP), is considered non-pythonic.

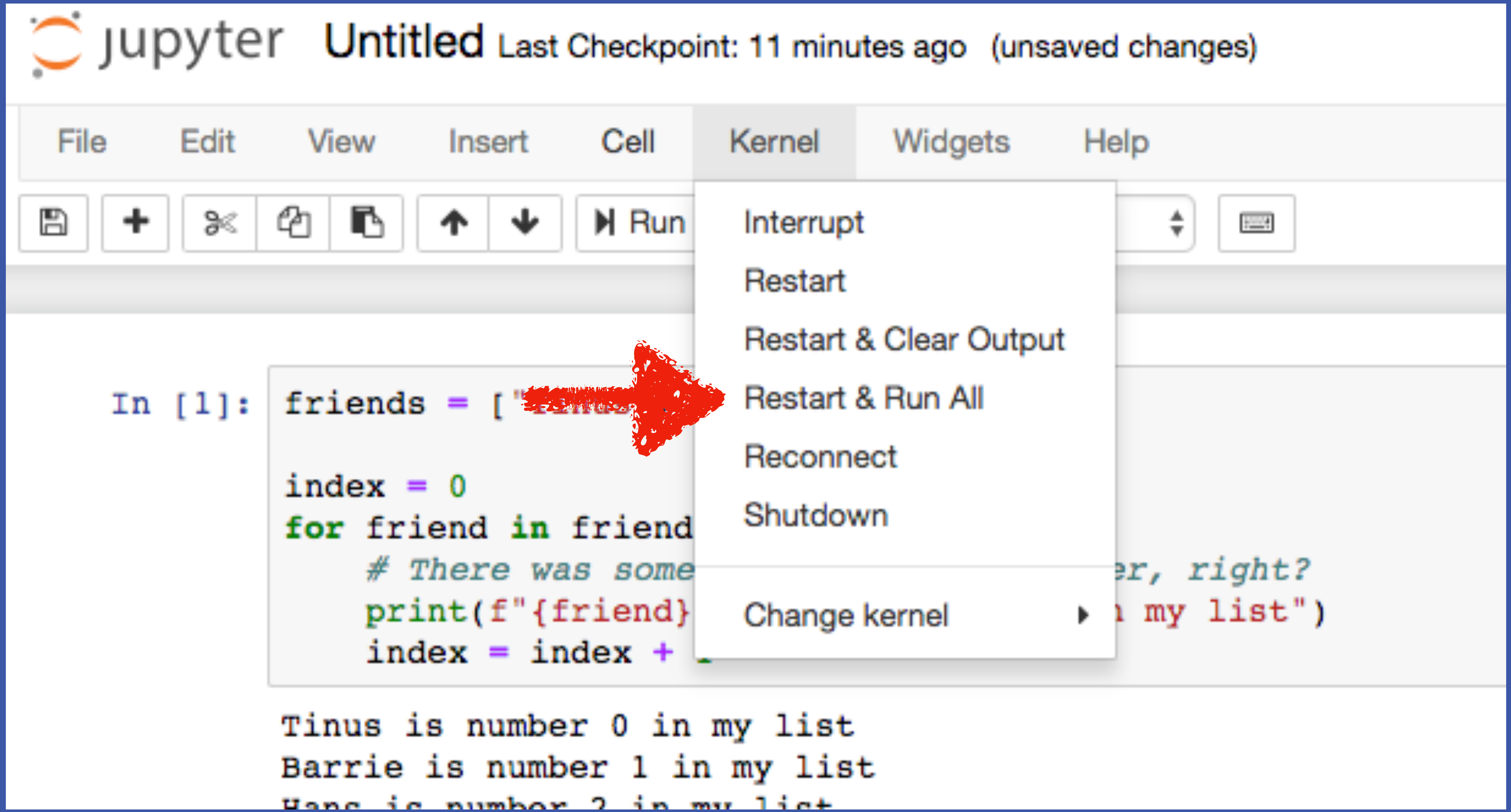
The better option is to use the built-in function `enumerate()`, available in both Python 2 and 3:

+500

```
for idx, val in enumerate(ints):
    print(idx, val)
```

Check out PEP 279 for more.

5. Turn it off and then on again



In [1]:

```
friends = ["

index = 0
for friend in friend
    # There was some                    er, right?
    print(f"{friend}                    n my list")
    index = index +
```

```
Tinus is number 0 in my list
Barrie is number 1 in my list
Hans is number 2 in my list
```

The Reddit Api

PRAW
latest
Search docs
GETTING STARTED
Quick Start
Installing PRAW
Authenticating via OAuth
Configuring PRAW
Running Multiple Instances of PRAW
Logging in PRAW
CODE OVERVIEW
The Reddit Instance
Working with PRAW's Models
Exceptions in PRAW
Edit on GitHub

Docs » PRAW: The Python Reddit API Wrapper

PRAW: The Python Reddit API Wrapper

PRAW's documentation is organized into the following sections:

- Getting Started
- Code Overview
- Tutorials
- Package Info

Documentation Conventions

Unless otherwise mentioned, all examples in this document assume the use of a **script** application. See Authenticating via OAuth for information on using **installed** applications and **web** applications.

Getting Started

Reddit API

Write a Jupyter Notebook that gets the 100 'hottest' submissions in a subreddit of your choice. Create a **list** with the **title**, **score** and **num_comments** of every submission and convert to a Pandas **Dataframe**. Then save it to a **JSON** file.

* Create a Reddit account (if you don't have that already) and an app. Note the **client ID** and **client secret**.
* Install **praw** using **pip**
* Import the **praw** and **pandas** libraries in your program.
* Create a new API instance (look at the examples-4 notebook under 'lesson 8')
* Make a request to the subreddit().hot() method.
* Create a new list
* Loop over the submissions and add new dictionaries to your new list containing the **title**, **score** and **num_comments** attributes.
* Create a new Dataframe with that list and print the first five results.
* Save your Dataframe as a JSON file.

Tips

* On Windows try using **Anaconda Prompt** to install praw: **pip install praw**
* Try to resist looking at the memes instead of the API output ;)

PLOTS IN PANDAS

Something about times and dates

Reddit Plots

Create a Jupyter Notebook that reads your Reddit JSON data and display plots about the data

- Import the **pandas** library and use the **read_json** method to read your JSON data. Show the first five items to confirm.
- A) Show a histogram with the scores, separated in 10 bins.
- B) Show a horizontal bar chart with the five highest scores.

Extended exercises

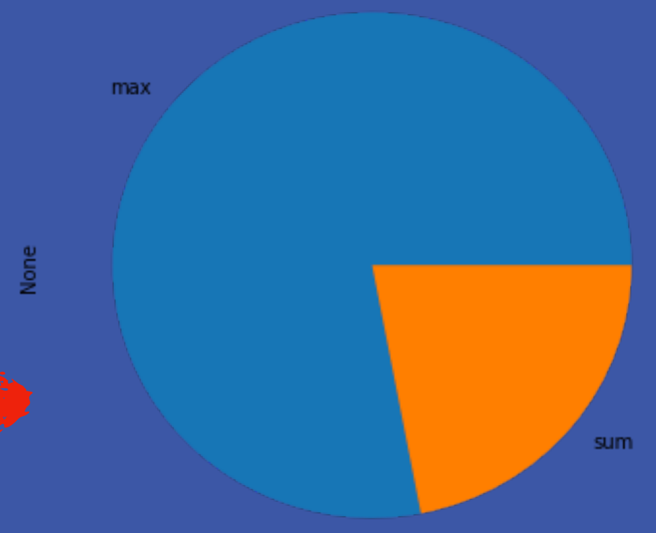
- C) Show a pie chart that shows the relation between the sum of all scores and the submission with the highest score.
- D) Show a horizontal bar chart like B) but also with the comment count of those submissions.

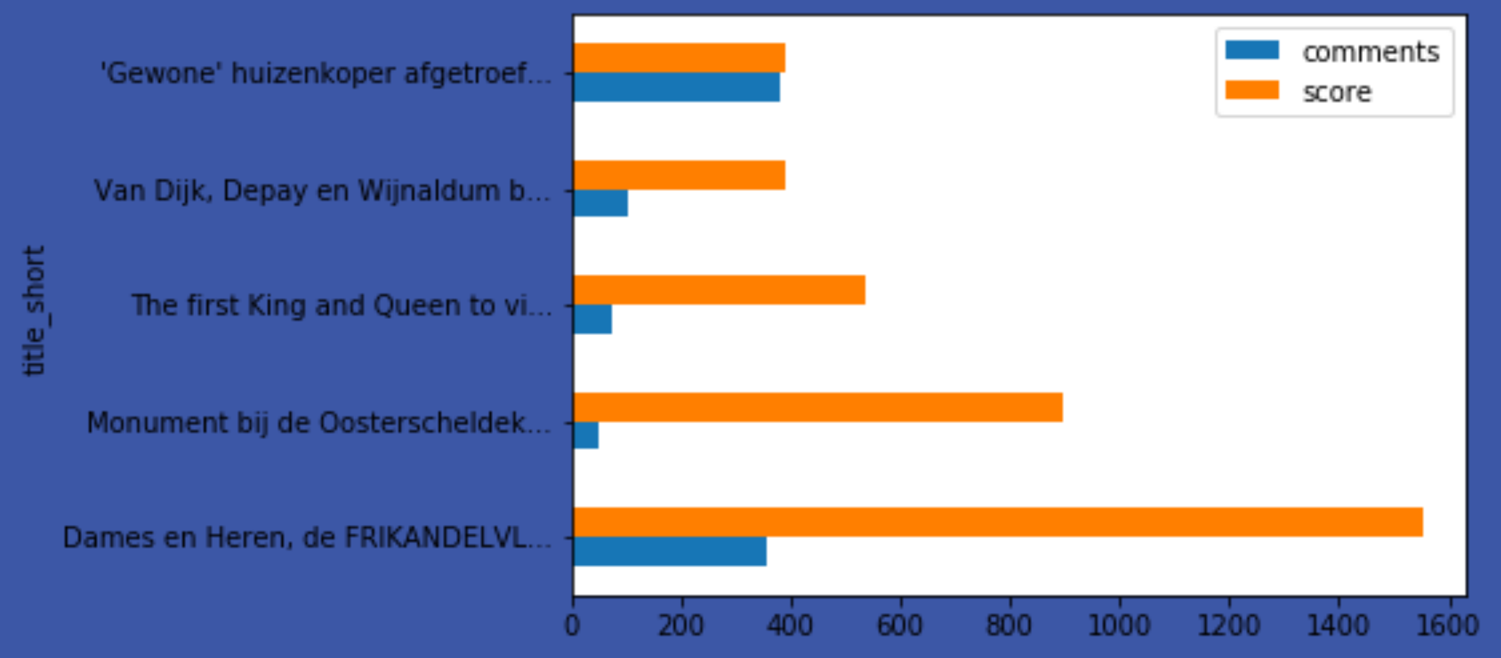
Tips

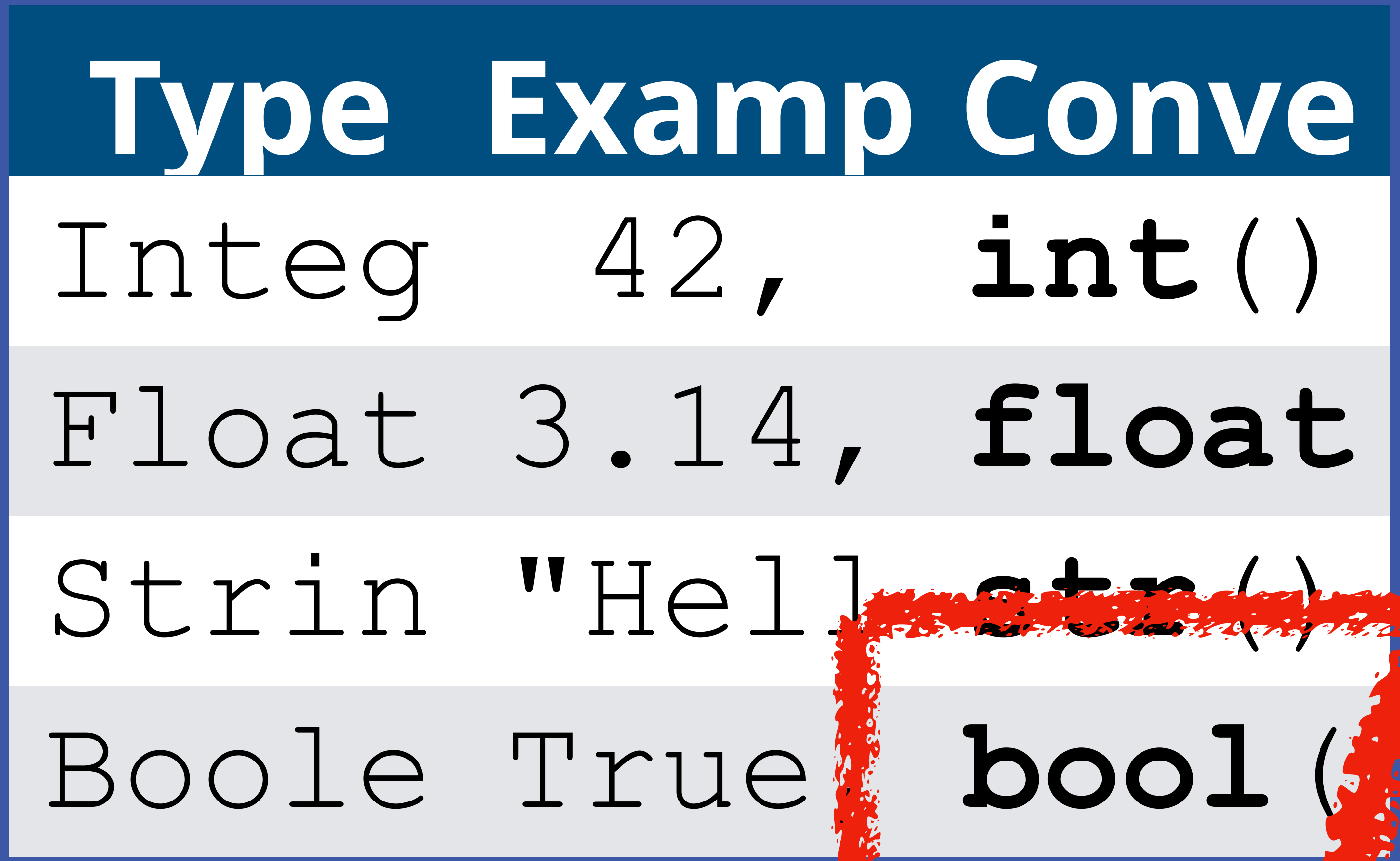
- You probably need the **sort_values()** method for B) and D)
- For **C)** you need to make a new Dataframe or Series
- For shortened titles take a look athe Pandas string function *or* use the **.apply()** method with a function

Superextended exercises

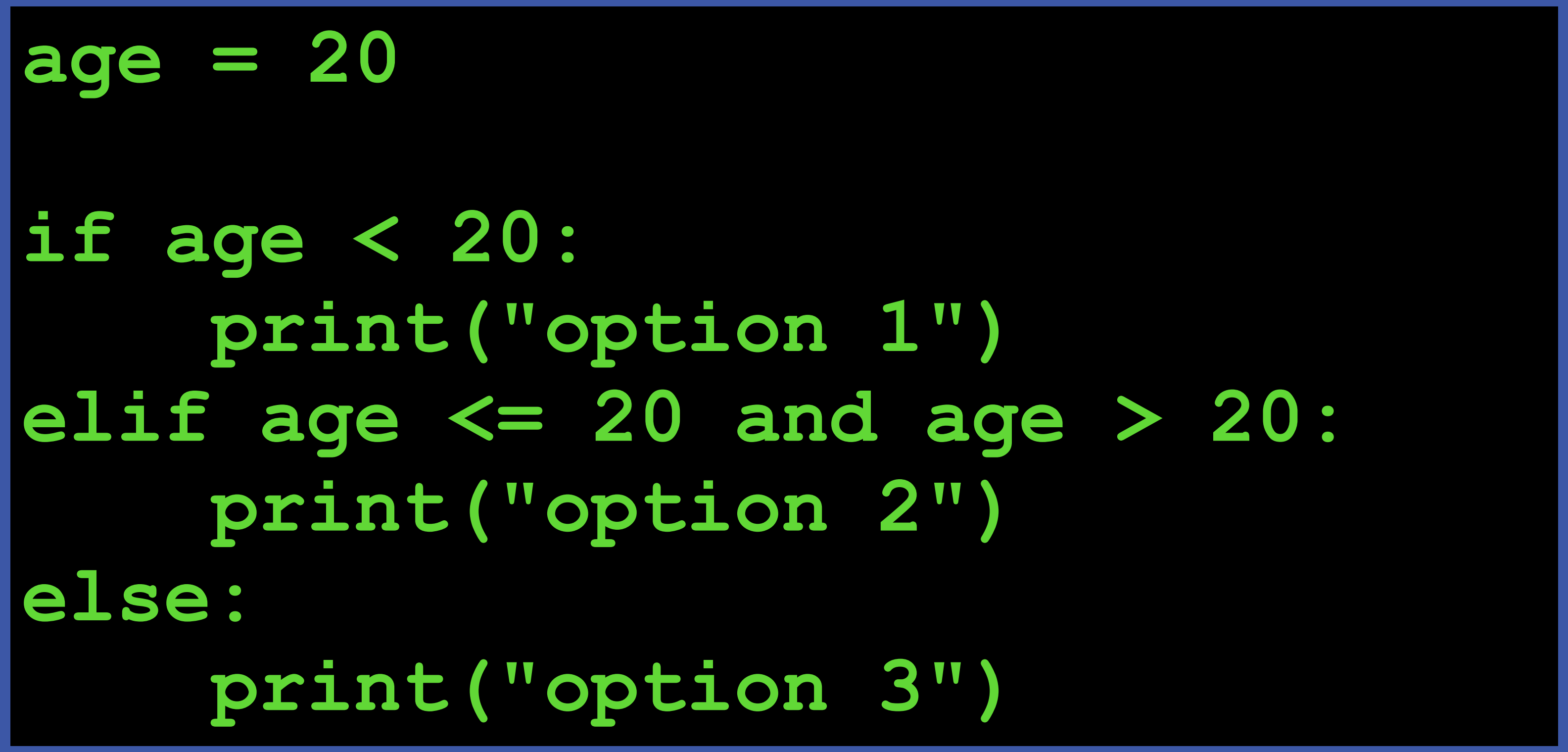
- For **B)** and **D)** include shortened titles for the y-ax
- Add more plots for interesting data points







Type	Examp	Conve
Integ	42,	**int**()
Float	3.14,	**float**
Strin	"Hel	**str**()
Boole	True	**bool**(



```
age = 20

if age < 20:
    print("option 1")
elif age <= 20 and age > 20:
    print("option 2")
else:
    print("option 3")
```

Compilation
Interpretation