

Dive into Python

INTRODUCTION TO DATA SCIENCE IN PYTHON



Hillary Green-Lerman
Lead Data Scientist, Looker

What you'll learn

- How to write and execute Python code with DataCamp
- How to load data from a spreadsheet
- How to turn data into beautiful plots

Solving a mystery with data



Using the IPython shell

The screenshot displays the DataCamp web interface for an exercise titled "Importing Python modules". The interface is divided into three main sections: instructions, a script editor, and an IPython shell.

Instructions Section:

- Importing Python modules**
Modules help group together related sets of tools in Python. In this exercise, we'll examine two modules that are frequently used by Data Scientists:
- 1. `pandas` : a module for working with spreadsheet data; the standard alias is `pd` .
- 2. `matplotlib` : a module for creating beautiful charts and graphs; the standard alias is `plt`
- Note that each module has a standard alias. You'll almost always import under these aliases to make it easier to type out the names of each tool.

Instructions 1/3 (35 XP):

- 1 In the script editor, use an `import` statement to import `pandas` .
[Take Hint \(-10 XP\)](#)
- 2 Add an `as` statement to alias `pandas` to `pd` .
- 3 Import the submodule `pyplot` (a submodule of `matplotlib`) under the standard alias `plt` .

Script Editor:

The script editor is titled "SCRIPT.PY" and shows a single line of code at the top: `1` .

IPYTHON SHELL:

The IPYTHON SHELL section shows the prompt `In [1]:` followed by a cursor.

Buttons:

At the bottom right of the script editor, there are three buttons: a circular arrow icon (refresh), a "Run Code" button, and a green "Submit Answer" button.

Using the script editor

The screenshot shows the DataCamp interface for an exercise. On the left, the exercise title is "Importing Python modules". Below the title, a paragraph explains that modules help group related tools in Python. A list of two modules is provided: `pandas` (alias `pd`) and `matplotlib` (alias `plt`). A note mentions that each module has a standard alias. Below this, the instructions section shows three steps: 1. Use an `import` statement to import `pandas`. 2. Add an `as` statement to alias `pandas` to `pd`. 3. Import the submodule `pyplot` (a submodule of `matplotlib`) under the standard alias `plt`. On the right, the "SCRIPT.PY" editor is open, showing a blank line 1. Below the editor are buttons for "Run Code" and "Submit Answer". At the bottom, the "IPYTHON SHELL" is visible, showing the prompt `In [1]:`.

DataCamp

Course Outline

EXERCISE

Importing Python modules

Modules help group together related sets of tools in Python. In this exercise, we'll examine two modules that are frequently used by Data Scientists:

1. `pandas` : a module for working with spreadsheet data; the standard alias is `pd` .
2. `matplotlib` : a module for creating beautiful charts and graphs; the standard alias is `plt` .

Note that each module has a standard alias. You'll almost always import under these aliases to make it easier to type out the names of each tool.

INSTRUCTIONS 1/3 35 XP

- 1 In the script editor, use an `import` statement to import `pandas` .

Take Hint (-10 XP)

- 2 Add an `as` statement to alias `pandas` to `pd` .
- 3 Import the submodule `pyplot` (a submodule of `matplotlib`) under the standard alias `plt` .

SCRIPT.PY

1

Run Code Submit Answer

IPYTHON SHELL

In [1]:

What is a module?

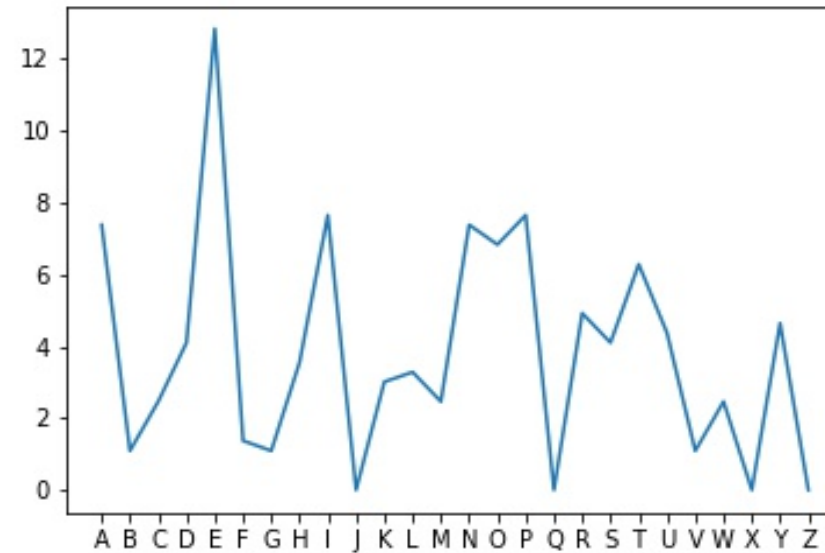
- Groups related tools together
- Makes it easy to know where to look for a particular tool
- Common examples:
 - `matplotlib`
 - `pandas`
 - `scikit-learn`
 - `scipy`
 - `nltk`

Importing pandas and matplotlib

```
import pandas as pd
from matplotlib import pyplot as plt
```

```
# Pandas loads our data
df = pd.read_csv('ransom.csv')

# Matplotlib plots and displays
plt.plot(df.letters, df.frequency)
plt.show()
```



Importing a module

- Importing a Module

```
import pandas
```

- Importing a module with an alias

```
import pandas as pd
```


Let's practice!

INTRODUCTION TO DATA SCIENCE IN PYTHON

Creating variables

INTRODUCTION TO DATA SCIENCE IN PYTHON



Hillary Green-Lerman
Lead Data Scientist, Looker

Filing a missing puppy report



```
name = "Bayes"  
height = 24  
weight = 75.5
```

Rules for variable names

- Must start with a letter (usually lowercase)
- After first letter, can use letters/numbers/underscores
- No spaces or special characters
- Case sensitive (`my_var` is different from `MY_VAR`)

```
# Valid Variables
```

```
bayes_weight
```

```
b
```

```
bayes42
```

```
# Invalid Variables
```

```
bayes-height
```

```
bayes!
```

```
42bayes
```

Error messages

```
bayes-height = 3
```

```
File "<stdin>", line 1
```

```
    bayes-height = 3
```

```
                ^
```

```
SyntaxError: can't assign to operator
```

Floats and strings

- *float*: represents an integer or decimal number

```
height = 24  
weight = 75.5
```

- *string*: represents text; can contain letters, numbers, spaces, and special characters

```
name = 'Bayes'  
breed = "Golden Retriever"
```

Common string mistakes

- Don't forget to use quotes! Without quotes, you'll get a name error.

```
owner = DataCamp
```

```
File "<stdin>", line 1, in <module>
    owner = DataCamp
NameError: name 'DataCamp' is not defined
```

- Use the same type of quotation mark. If you start with a single quote, and end with a double quote, you'll get a syntax error.

```
fur_color = "blonde'
```

```
File "<stdin>", line 1
    fur_color = "blonde'
                  ^
SyntaxError: EOL while scanning string literal
```

Displaying variables

```
name = "Bayes"
```

```
height = 24
```

```
weight = 75
```

```
print(height)
```

```
24
```


Let's practice!

INTRODUCTION TO DATA SCIENCE IN PYTHON

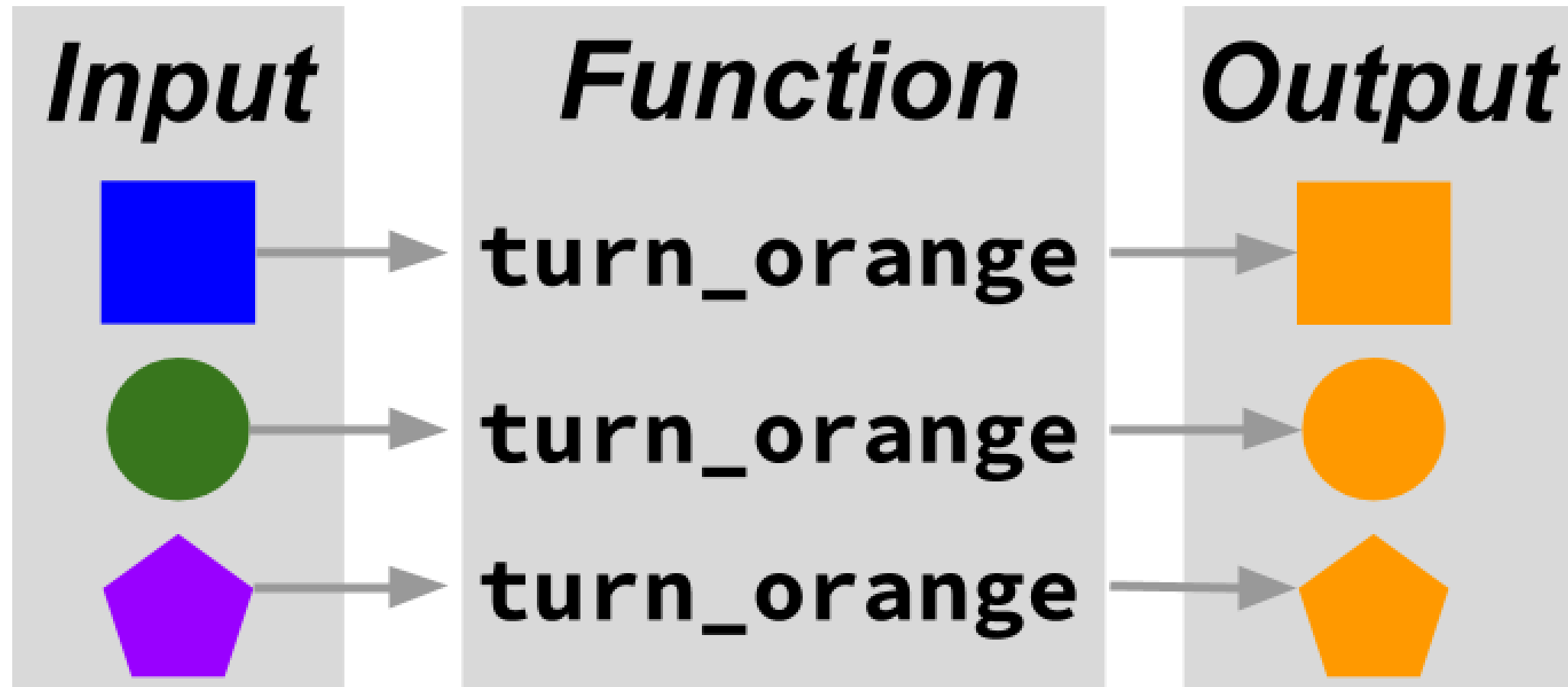
What is a function?

INTRODUCTION TO DATA SCIENCE IN PYTHON



Hillary Green-Lerman
Lead Data Scientist, Looker

A function is an action



Functions in code

```
import pandas as pd
from matplotlib import pyplot as plt

df = pd.read_csv('letter_frequency.csv')

plt.plot(df.letter_index, df.frequency, label='Ransom')
plt.show()
```

Functions perform actions:

- `pd.read_csv()` turns a csv file into a table in Python
- `plt.plot()` turns data into a line plot
- `plt.show()` displays plot in a new window

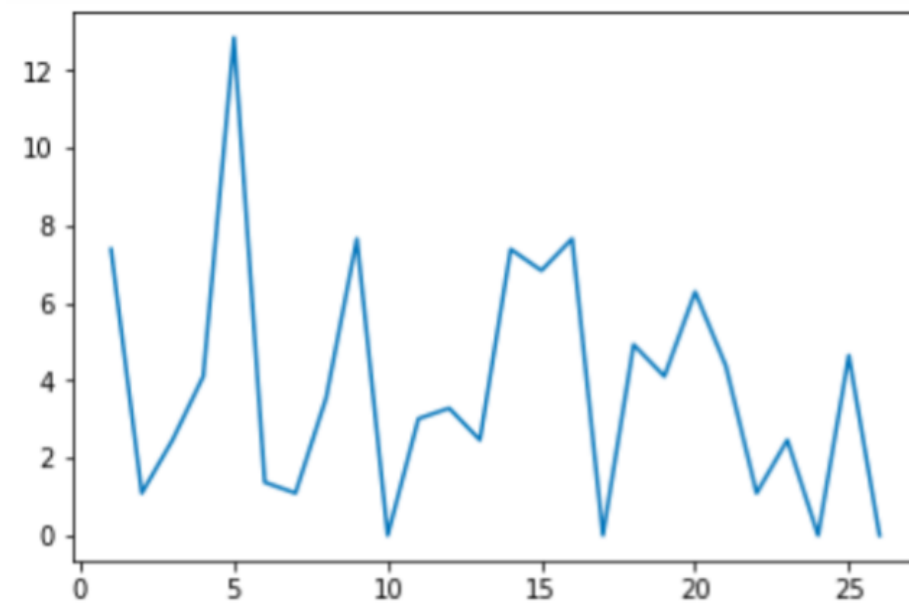
```
plt.plot(df.letter_index, df.frequency, label='Ransom')
```

Function

Positional Arguments

Keyword Argument

letter_index	letter	frequency
1	A	7.38
2	B	1.09
3	C	2.46
4	D	4.10
...



Anatomy of a function: function name

```
plt.plot(df.letter_index, df.frequency, label='Ransom')
```

Function

Function Name:

- Starts with the module that the function "lives" in (`plt`)
- Followed by the name of the function (`plot`)
- Function name is always followed by parentheses (`()`)

Anatomy of a function: positional arguments

```
plt.plot(df.letter_index, df.frequency, label='Ransom')
```

Positional Arguments

Positional Arguments:

- These are *inputs* to a function; they tell the function how to do its job
- Order matters!

Anatomy of a function: keyword arguments

```
plt.plot(df.letter_index, df.frequency, label='Ransom')
```

Keyword Argument

Keyword Arguments:

- Must come *after* positional arguments
- Start with the name of the argument (`label`), then an equals sign (`=`)
- Followed by the argument (`Ransom`)

Common function errors

- Missing commas between arguments

```
plt.plot(df.letter_index df.frequency, label='Ransom')
```



Missing commas!

- Missing closed parenthesis

```
plt.plot(df.letter_index, df.frequency, label='Ransom')
```



Missing parenthesis!

Let's practice!

INTRODUCTION TO DATA SCIENCE IN PYTHON