



Python Fundamentals

Thursday: Putting it all together

Colby Witherup Wood

NU IT Research Computing Services

About this bootcamp

Cameras are **not** required during these lectures.

During lectures, ask questions in the Zoom chat. If you know the answers, feel free to answer questions in the chat, too.

If my internet goes out during a lecture, that means everyone gets a 10-minute break! (Has never happened, but it's good to have a plan.)



About this bootcamp

From 3 pm to 4 pm Central I will be back here (same Zoom channel) to answer any questions about the homework or quiz.

You can also email me questions at colby.witherup@northwestern.edu and I will do my best to answer as soon as I can.

Plan for the week - **6 basic objects**

LEARN PYTHON

Monday

- Objects
 - **integers**
 - **floats**
 - **booleans**
 - **strings**
- Functions
 - basic operators
 - convert between objects
 - string functions
- Concepts
 - naming variables
 - indexing strings

Tuesday

- Objects
 - **lists**
- Functions
 - list functions
- Concepts
 - for loops
 - if statements
 - error handling
 - importing functions from modules

Wednesday

- Objects
 - **dictionaries**
 - files
- Concepts
 - writing files
 - looping through dictionaries
 - writing your own functions

Thursday

- Concepts
 - interactive coding
 - writing scripts
 - running scripts
 - putting it all together

DATAFRAMES

Friday

- Objects
 - pandas dataframes
 - pandas series
 - plots
- Functions
 - dataframe functions
 - series functions
- Concepts
 - selecting data
 - filtering data
 - plotting data

TODAY

We will be running a script outside of a Jupyter Notebook.

Two ways to participate:

1. Use Jupyter Lab on your own computer.
2. Use repl.it online. Go to <https://repl.it/> and click the blue button that says "<>start coding". Then choose Python and Create Repl.

Everyone needs to download the materials to their own computer today. Go to:

github.com/agithasnoname/pythonbootcampthursday

Click the green Clone button and choose Download Zip. Unzip the folder.

Python is a language

Your computer also has a language for talking to your operating system. We call this a **command line language**.

On Windows it is called PowerShell. On a Mac or a Linux server (like Quest) it is called Unix Bash (or just Bash).

Python is a language

You can talk to your operating system through a **command line terminal or shell**.

On Windows this is PowerShell or Anaconda Prompt.
On a Mac this is Terminal.

Python is a language

At the command line, you can also switch the shell to speak Python, as long as you have Python installed on your computer.

When you start Python this way, it is **interactive coding** – line by line. I'll show you.



Python console

There are also shells/terminals/consoles just for interactive programming in Python.

All IDEs include an interactive shell.

Let's experience some interactive coding.

Console demos - repl.it and Jupyter Lab

```
6 + 4
```

```
name = "Molly"
```

```
for i in name:  
    print(i)
```

```
for i in name:  
    if i.isupper():  
        print(i)
```

```
name.replace("M", "H")
```

When might you use a Python console?

When you're testing out code while writing a script

When you need to check a piece of code or do something quickly



Scripts

A Python script is a plain text file that you save to end in .py

A script runs all the code in the file from top to bottom

Scripts

Also called a computer program

A Python module like Statistics is just a Python script containing several functions

Scripts

There are two ways to run a script:

1. On the command line – you type:
`python myscript.py`
2. In an IDE



Scripts

Let's look at a Python script

When might you use a Python script?

When you want to run code while you do something else

This could be because your code is doing complicated calculations or looping through a very large dataset

You can run scripts on your own computer or a remote server

When might you use a Python script?

When you want to combine your Python code with other scripts and other software programs in a computational pipeline

You don't want to have to be there to start each new step in your workflow

When might you use a Python script?

When you want to parallelize your code and run the same code on multiple datasets at the same time.

When might you use a Python script?

When you want to write a piece of software.

When you want to write your own Python module.

When might you use a Jupyter Notebook?

When you are exploring your data.

When you are cleaning data.

When you are visualizing your data.

When you want to include text, images, links alongside your code.

When might you use a Jupyter Notebook?

Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks

Adam Rule, Amanda Birmingham, Cristal Zuniga, Ilkay Altintas, Shih-Cheng Huang, Rob Knight, Niema Moshiri, Mai H. Nguyen, Sara Brin Rosenthal, Fernando Pérez, Peter W. Rose

PLOS Computational Biology July 25, 2019

<https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1007007>



THURSDAY HOMEWORK

**Write your own
conversion script (or
notebook)**

thursdayHW.docx
contains your
instructions

You can either do the work in a new Jupyter Notebook or try to write a script.

You are going to create a conversion calculator.

In today's folder, there is a file called conversionMeasures.csv.

THURSDAY HOMEWORK

**Write your own
conversion script (or
notebook)**

thursdayHW.docx
contains your
instructions

unit 1, factor, unit 2

```
kilometer,1000,meter|
meter,100,centimeter
inch,2.54,centimeter
foot,30.48,centimeter
mile,1.609,kilometer
centimeter,0.3937,inch
meter,39.37,inch
kilometer,0.6214,mile
square_meter,10.76,square_foot
square_mile,640,acres
square_foot,929,cm2
acre,43560,square_foot
liter,1000,cm3
liter,1.057,quart
liter,61.02,cubic_inch
liter,0.03532,cubic_foot
cubic_meter,1000,liter
cubic_meter,35.32,cubic_foot
```

unit 1 x factor = unit 2

continues...



THURSDAY HOMEWORK

**Write your own
conversion script (or
notebook)**

thursdayHW.docx
contains your
instructions

Write a script or notebook that converts a given value from its original unit to a different unit:

- store the conversion data in some format
- include a function to convert between units
- print out a full sentence response
- anticipate some errors
- run your script on the provided test examples



THURSDAY HOMEWORK

**Write your own
conversion script (or
notebook)**

thursdayHW.docx
contains your
instructions

There are also BONUS challenges included, if you complete the original assignment.



THURSDAY HOMEWORK

Complete by 10 am
Central tomorrow

Tomorrow's lecture
starts at 10 am

See you (maybe) at 3 pm Central

Or email me questions about your
script/notebook.