

# Python Fundamentals

## Monday: Objects and Functions

---

Colby Witherup Wood

NU IT Research Computing Services



# About me

Please call me Colby.

Colby is a type of cheese.

I use she/her/hers pronouns.

I am a Data Scientist in NUIT Research Computing Services.

I love writing and teaching Python code.

My GitHub name is a Game of Thrones reference, but I'm actually only a medium fan of the show, and only of the first 7 seasons plus the part in Season 8 Episode 3 where...

# About this bootcamp

Hopefully you read the word doc I sent out in the email, which includes my helpful tips for learning Python, like "drink coffee, wear pajamas, and eat your favorite snacks".

Each day will include a live "lecture", two or three "homework assignments", and a "quiz". Nothing will be graded, but answer keys are provided.

# 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

Lectures will not be recorded (sorry!), but if you miss a lecture, you should be able to work through the slides and lecture notebook on your own, plus email me or show up to the Q&A if you have questions.

# 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](mailto:colby.witherup@northwestern.edu) and I will do my best to answer as soon as I can.



# Python



# Python is a language

# How do we talk to our computer in Python?

- **Interactive programming:** through a shell (interpreter, console) one line at a time
- **Batch programming:** running a whole script (a plain text file containing one to many lines of code)
- With the help of a GUI (graphical user interface). GUIs for coding are called **IDEs (Integrated Development Environments)**.

# How do we talk to our computer in Python?

- **Interactive programming:** Thursday
- **Batch programming:** Thursday
- **IDE (Jupyter Notebook):** Monday, Tuesday, Wednesday, Friday

# Python is:

## **50% Syntax**

- Which words to use
- Punctuation
- Order
- Indentation
- Shortcuts

## **50% Logic**

- Which tools to use in which order
- Creative
- Specific to one problem

**Try not to worry if you don't understand everything today**

# Looking things up

You do not have to memorize everything you learn.

Coders of all levels rely on Google to help them recall the syntax for a particular task.

Google will point you to the official Python documentation or forums like Stack Overflow or blog tutorials on sites like Medium.

# Objects and functions

---

The two main concepts of Python.

Object: a particular piece of data (like a noun)

Function: something you can do to/with an object (like a verb)

# Objects and functions

You can use certain functions with certain classes of objects.

Some functions are shared between classes of objects, and some are not.

# Objects and functions



**Molly**

- Feed Molly
- Pet Molly
- Draw Molly



**My bicycle**

- Pedal my bicycle
- Pet my bicycle (limited use)
- Draw my bicycle

# Objects and functions



## **Molly (class Dog)**

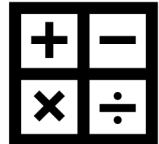
- Feed a dog
- Pet a dog
- Draw a dog



## **My bicycle (class Bicycles)**

- Pedal a bicycle
- Pet a bicycle (limited use)
- Draw a bicycle

# Data types (same as classes in Python)



## **3 (class Integer)**

- Divide ( $3 / 3 = 1$ )
- Convert 3 to 3.0
- Add ( $3 + 3 = 6$ )



## **"3" (class String)**

- Replace ("3" becomes "1")
- Convert "3" to 3.0
- Add ("3" + "3" = "33")

# Plan for the week

## 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

# Plan for the week - 6 basic objects

LEARN PYTHON				DATAFRAMES
Monday	Tuesday	Wednesday	Thursday	Friday
<ul style="list-style-type: none"><li>• Objects<ul style="list-style-type: none"><li>• <b>integers</b></li><li>• <b>floats</b></li><li>• <b>booleans</b></li><li>• <b>strings</b></li></ul></li><li>• Functions<ul style="list-style-type: none"><li>• basic operators</li><li>• convert between objects</li><li>• string functions</li></ul></li><li>• Concepts<ul style="list-style-type: none"><li>• naming variables</li><li>• indexing strings</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Objects<ul style="list-style-type: none"><li>• <b>lists</b></li></ul></li><li>• Functions<ul style="list-style-type: none"><li>• list functions</li></ul></li><li>• Concepts<ul style="list-style-type: none"><li>• for loops</li><li>• if statements</li><li>• error handling</li><li>• importing functions from modules</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Objects<ul style="list-style-type: none"><li>• <b>dictionaries</b></li></ul></li><li>• files</li><li>• Concepts<ul style="list-style-type: none"><li>• writing files</li><li>• looping through dictionaries</li><li>• writing your own functions</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Concepts<ul style="list-style-type: none"><li>• interactive coding</li><li>• writing scripts</li><li>• running scripts</li><li>• putting it all together</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Objects<ul style="list-style-type: none"><li>• pandas dataframes</li><li>• pandas series</li><li>• plots</li></ul></li><li>• Functions<ul style="list-style-type: none"><li>• dataframe functions</li><li>• series functions</li></ul></li><li>• Concepts<ul style="list-style-type: none"><li>• selecting data</li><li>• filtering data</li><li>• plotting data</li></ul></li></ul>

# Today's objects

---

- Integer (whole number) 3
- Float (number with digits following a decimal point) 3.0
- String (text or characters) "3" or "three"
- Boolean (True or False)
- *Assigning objects to variables*
- *Indexing to get only part of an object*

# Today's functions

---

- Basic operators: + - \* /
- Functions to convert between data types
- String functions

# Options to open mondayLecture.ipynb:

## **If you are running Jupyter Lab on your own computer:**

- Go to <https://github.com/aGitHasNoName/pythonBootcampMonday>. Click on the green Clone button and choose Download ZIP. Unzip that folder. Open Anaconda Navigator and open Jupyter Lab. Navigate to the folder you just unzipped. Open mondayLecture.ipynb

## **If you are using Google Colab (online)**

- Go to [colab.research.google.com](https://colab.research.google.com). Choose GitHub on the orange menu. Search for agithasnoname/pythonbootcampmonday. Choose mondayLecture.ipynb

# Prepare your desktop setup

---

- You can set my Zoom share up next to your own notebook
- You can run the code in your version of the notebook while just listening to me as I walk through it
- You can just sit back and watch me run through it until it is time to do an exercise

# Let's code!



# Variable names

- No spaces
- Case matters - x is not X
- Must start with a letter
- Should be meaningful

Style conventions:

- camelCase
- separate\_with\_underscores

# Today's objects

- ✓ Integer (whole number) 3
- ✓ Float (number with digits following a decimal point) 3.0
- ✓ String (text or characters) "3" or "three"
- ✓ Boolean (True or False)
- ✓ Assigning objects to variables
  - *Indexing to get only part of an object*

# Today's functions

- ✓ Basic operators: + - \* /
- ✓ Functions to convert between data types
  - String functions

# Python functions

Parentheses immediately follow a function name.

```
print("Hello World")
```

The information inside the parentheses is called an **argument**.

We **passed** the argument "Hello World" to the print function.

# Two types of Python functions

1. Can stand on its own. Takes an object as an **argument**.  
These types of functions usually do something **with** the object.

```
my_name = "Colby"
```

```
print(my_name)           Colby
```

```
len(my_name)            5
```

# Two types of Python functions

2. Follows an object. Some take arguments and some don't.  
These types of functions usually do something **to** the object.  
These are sometimes called **methods** in Python.

```
my_name = "Colby"  
  
my_name.replace("C", "c")          "colby"  
  
my_name.upper()                  "COLBY"
```

# Python functions

When a function takes multiple arguments, they are separated by a comma:

```
my_name.replace("C", "c")      "colby"
```

In Python indexing, we  
start counting with 0

# Indexing strings

```
"H   e   l   l   o       w   o   r   l   d   ! "
 0   1   2   3   4   5   6   7   8   9   10  11
```

"Hello World!" [0] is "H"

"Hello World!" [4] is "o"

# Indexing strings

```
"H   e   l   l   o       W   o   r   l   d   !"  
 0   1   2   3   4   5   6   7   8   9   10  11
```

"Hello World!"[6:12] is "World!"

To take a substring, you start with the position of the first character you want included, and end with the position 1 past the last character you want.

# Indexing strings

```
"H   e   l   l   o       w   o   r   l   d   !"  
0   1   2   3   4   5   6   7   8   9   10  11
```

```
"Hello World!"[6:12] is "World!"
```

This is because behind the scenes Python is actually labeling the invisible spot between each letter in our string.

# Let's code!



# MONDAY HOMEWORK

Complete by 10 am  
Central tomorrow

Answer keys are in  
today's folder with the  
homework and quiz

- mondayHW1.ipynb
- mondayHW2.ipynb
- mondayQuiz.ipynb

The homework contains new skills, it  
isn't just repetition.

See you (maybe) at 3 pm Central