COMS BC1016

Introduction to Computational Thinking and Data Science

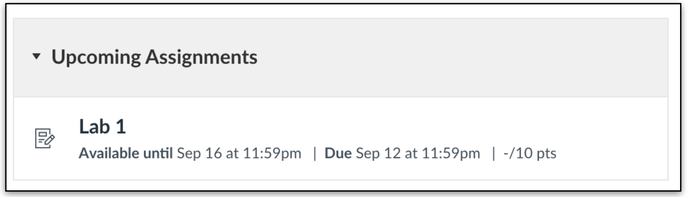
Lecture 3: Introduction to Python

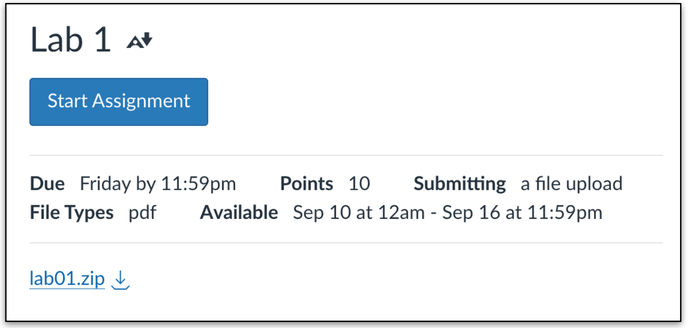
Prof. [illegible]025

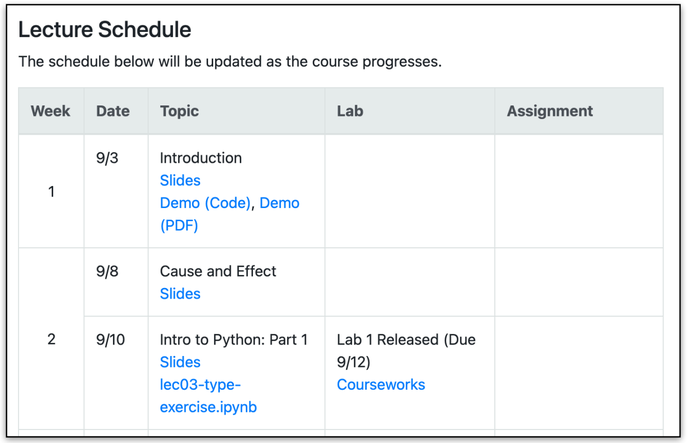
Sept 10, 2025

Logistical Updates

- Labs and HWs will be posted on Courseworks
  - Download the files from the assignment page
  - 1017 sections share the same Courseworks
- Lectures and in-class activities will still be posted on course website
  - Slides uploaded by 11:30am day of
  - Preliminary versions of slides may appear earlier







Lecture Schedule

The schedule below will be updated as the course progresses.

Week	Date	Topic	Lab	Assignment
1	9/3	Introduction Slides Demo (Code), Demo (PDF)		
2	9/8	Cause and Effect Slides		
	9/10	Intro to Python: Part 1 Slides lec03-type-exercise.ipynb	Lab 1 Released (Due 9/12) Courseworks	

Lab Updates

- The other 1016 section didn't have enough students enrolled, so we lost a lab section :(
  - If you need help getting into a different lab section, please talk to me!
- Minor changes to the other labs
  - Section 4 (W 2:30p) lab starts at 2:40p
- Email your TA if you'll be late or missing!
  - 50% of your lab grade is attendance!
  - One unexcused absence + lowest lab dropped

Labs:

- W 2:40pm - 4:00pm, Milstein Center 516 (Ken Mah)
- W 4:00pm - 5:30pm, Milstein Center 516 (Erin Ma)
- Th 9:40am - 11:10am, Milstein Center 516 (Amaya Kejriwal)
- Th 11:20am - 12:50pm, Milstein Center 516 (Justin Zeng)

Misc Updates

- No TA office hours this week
  - I'll still be having office hours if you want to stop by
- We're (probably) not getting a larger room
  - Apparently this room has a capacity of 70???
- First HW will be released on Monday

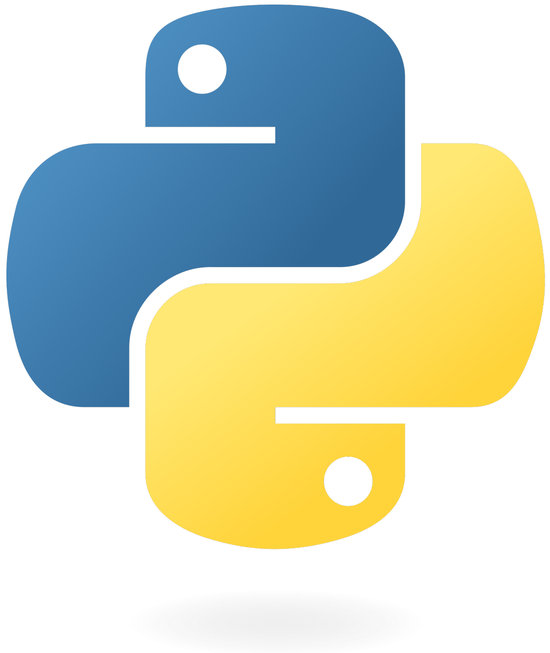
Python

Python Intro

- Popular programming for software development

- Especially popular for data science

- Learning programing is about learning how to think computational & transfers to other languages

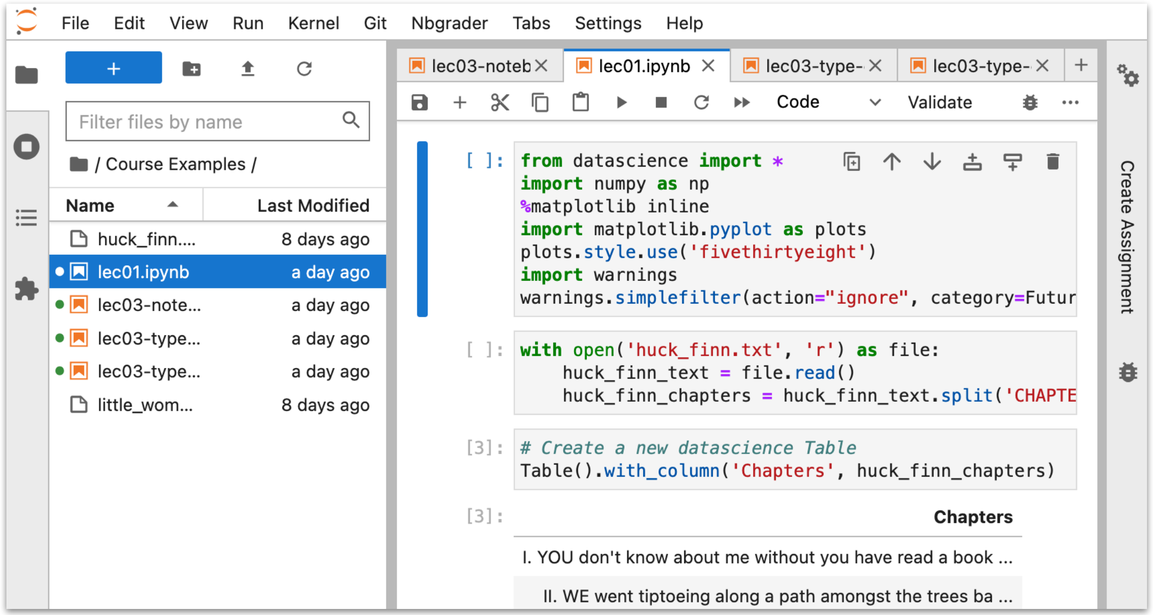


How to Approach Programming in BC1016

- Programming is all about practice
  - Class: Demos introduce terms & rules
  - Labs & Homework: Try out programming yourself
    - Ultimate goal: writing your own code that can solve new problems!

Jupyter Notebooks

- Can be run locally or in cloud-based environments
  - For this class: JupyterHub Server
- Benefits of cloud-based
  - Access anywhere
  - Expandable compute resources



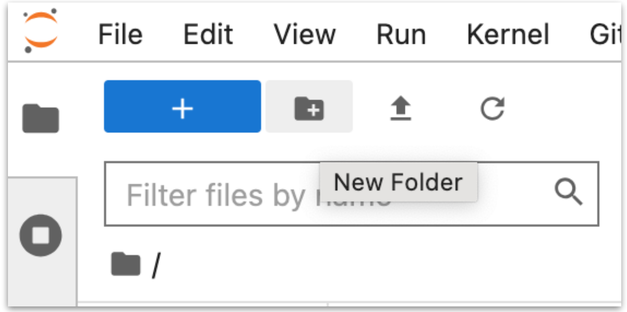
Data Types and Demos

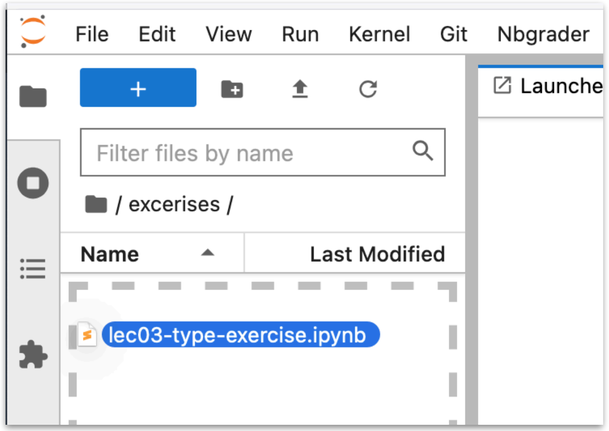
Follow Along!

1. Download the class exercise python notebook (.ipynb) from the course website
2. Go to the course Jupyter Hub link
3. Create a folder called “exercises”
4. Drag the lec03-type-exercise.ipynb file into the folder

For the first half, follow along in the lecture and we’ll try things together

For the second half, I’ll give examples that you have 10 minutes to work out for each one

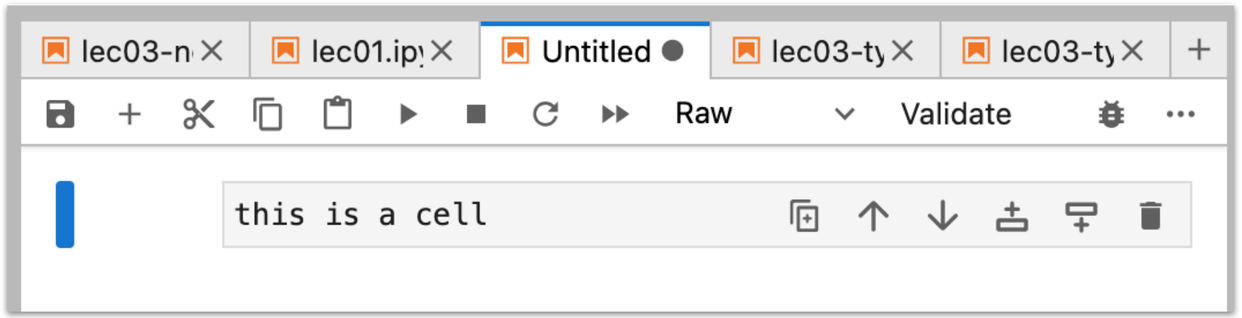


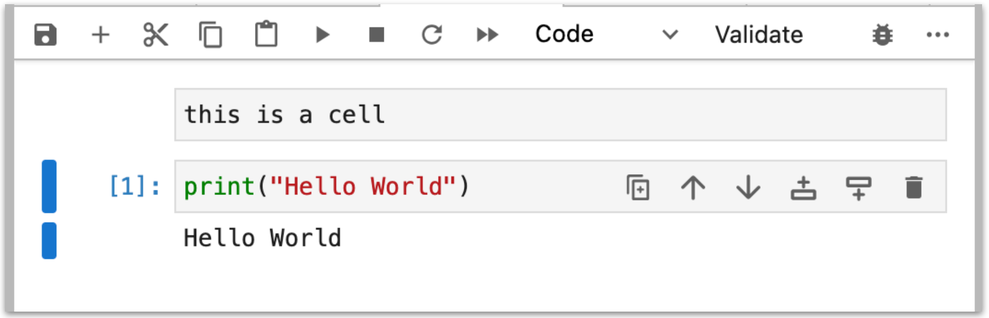


Jupyter Notebooks: Terms

- Cells: block or section for writing code / notes / etc

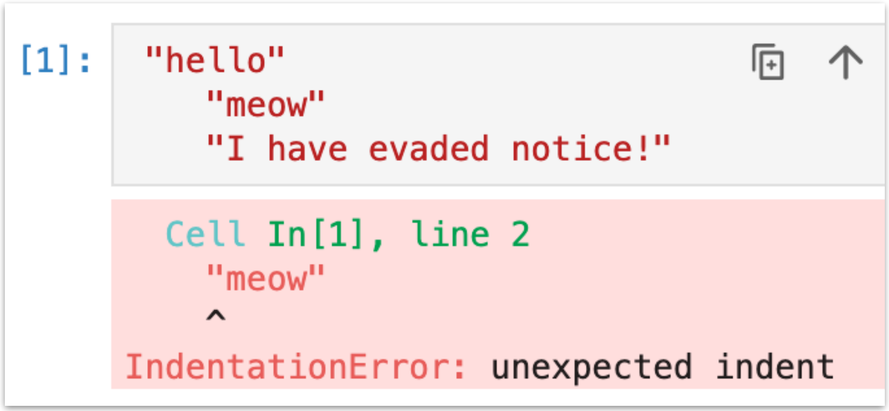
- Kernel: executes code in cells (when you run a “cell”)





Some notes about Python

- Spaces and new lines matter
  - Each new instruction should be on a new line
  - Be careful not to add extra spaces or indentations at the beginning of the line!
- Python runs line-by-line
  - It'll stop as soon as it runs into an issue and tell you what's wrong
- Lines starting with # are comments and are ignored



Numbers

- Integers: Whole numbers
  - e.g., `3, -10, 25`
- Floats: Anything with decimals
  - e.g., `3.1, -10.2, 2.0`
- Basic calculations
  - e.g., `+, -, *, /`

Numbers

- Integers: Whole numbers
  - e.g., `3, -10, 25`
- Floats: Anything with decimals
  - e.g., `3.1, -10.2, 2.0`
- Basic calculations
  - e.g., `+, -, *, /`

Some questions:

1. What happens if we add different number types?
   a. Multiply or divide?
2. What if I wanted 2 raised to the power of 4?

Functions

- Type of abstraction for pre-defined set of code or instructions
  - Commonly takes inputs, performs a computation, and produces output
  - Many useful things built in!
  - Some useful functions aren't built in, so we need to `import` them

```
abs(-1)
```

```
1
```

```
max(4,200,7)
```

Function

inputs separated by commas

```
200
```

```
import numpy
numpy.sqrt(4)
```

We tell Python this function is from here

```
2.0
```

Assignment Statements

- Assignments change the meaning of the name to the left the = symbol

- `Variables` are values you can assign values to
  - “Variable” because they can change

- You can assign outputs of functions and operations to variables

```
max_of_list = max(4,200,7)
max_of_list + 5
```

```
205
```

Strings

- Text in python!
  - `"a"`
  - `'This is a sentence'`
  - `"This is another sentence. Wow!"`
- You can convert values to a string using `str(..)`
  - `str(5)` becomes `"5"`
- You can convert strings of numbers to numbers
  - `int('12'),float('1.2')`

Some questions:

1. What happens if you start a string with a double quote and end it with a single quote?
2. What if I want to write the following sentence as a string using single quotes?

   `a. That's weird`

Booleans

- True or False values
- Useful for conditional statements
  - Usually of the form “if some statement is true, then execute some code. Otherwise, do something else.”
- Use == to check equivalence

Arrays

Note: In this course we'll mostly be using a special version of arrays, which will be covered in more detail next lecture

- Arrays are sequences of values
- Start with brackets and each element is separated by commas
  - [2,4,6,8]
  - [‘apples’, ‘bananas’, ‘oranges’]
  - [‘apples’, ‘bananas’, ‘apples’, ‘apples’]
- Arrays are zero-indexed
  - The first element is the 0th and the second is 1st

Types

Type	Example
Int	**3**
Float	3.0
String	‘three’
Boolean	True
Arrays	[1, 2, 3, 4]
Functions	abs(-5)

Type Exercise

Let's say you have defined the following variables in your notebook

```
x = 3
y = '4'
z = '5.6'
```

What would the source of the error in these examples?

How could you fix it?

```
1. x + y
2. x + int(y + z)
3. str(x) + int(y)
4. y + float(z)
```

Type Exercise

Take 10 min to work on your own or with a neighbor!

Let's say you have defined the following variables in your notebook

```
x = 3
y = '4'
z = '5.6'
```

What would the source of the error in this example:

```
x + y
```

How could you fix it?

Type Exercise

Take 10 min to work on your own or with a neighbor!

Let's say you have defined the following variables in your notebook

```
x = 3
y = '4'
z = '5.6'
```

What would the source of the error in this example:

```
x + int(y + z)
```

How could you fix it?

Type Exercise

Take 10 min to work on your own or with a neighbor!

Let's say you have defined the following variables in your notebook

```
x = 3
y = '4'
z = '5.6'
```

What would the source of the error in this example:

```
str(x) + int(y)
```

How could you fix it?

Type Exercise

Take 10 min to work on your own or with a neighbor!

Let's say you have defined the following variables in your notebook

```
x = 3
y = '4'
z = '5.6'
```

What would the source of the error in this example:

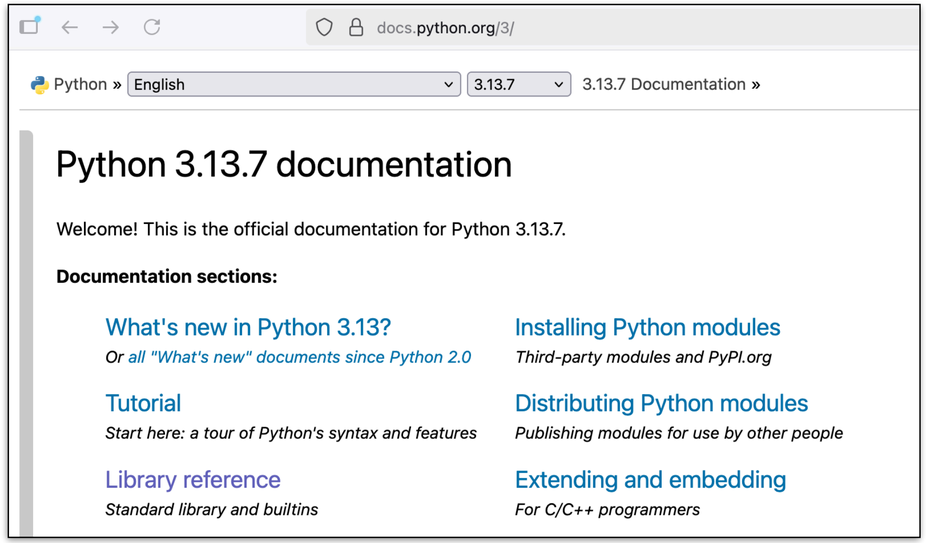
```
y + float(z)
```

How could you fix it?

Python Reference

https://www.data8.org/sp25/reference/

https://docs.python.org/3/



Next Class

- Today
  - Jupyter Notebooks
  - Expressions
  - Data Types
- Monday
  - Tables
- Wednesday
  - Charts & Visualization