Intro CS:
Let's Start With The Basics

Different Programming Languages:

Programming Languages That You Will See In the CS Progression:

- Python
- Kotlin
- Scheme
- C
- Html
- Javascript

What is Python?

- First released in 1991
  - Created by Guido Van Rossum
- Dynamic Typing
- Emphasizes readability

What Are Variables?:

Much like in math, variables in CS are representations of values that can be manipulated. They are named storage values that can be operated on in specific ways according to the type of data that is stored in that location.

Variables, Best Practice:

```
1   def function1() :
2       orange = [5, 12]
3       for banana in orange :
4           apple += banana * banana
5       lemon = sqrt(apple)
6       return lemon
```

```
1   def function1() :
2       sides = [5, 12]
3       for side in sides :
4           hypotenuse_squared += side * side
5       hypotenuse = sqrt(hypotenuse_squared)
6       return hypotenuse
```

What Are Data Types:

Data types are classifications for variables for what they can hold. It can also limit/designate what operations can be performed on a variable.

Different Types:

- Integer (ie: 4)
- Float (ie: 8.3)
- Character (ie: ‘a’)
- String (ie: “hello”)
- Boolean (ie: True)
- List (ie: [‘a’, ‘b’, ‘c’])
- Dictionary (ie: {“one”, “two”, “three”}
- (ect.)

Python, Dynamic Typing:

In Python, unlike some other programming languages like Java or C, has dynamic typing. Dynamic typing means that you do not have declare the typing of the variable. Instead, the type is determined when a value is assigned to it.

What Can You Do With Each Type?:

The types that you will work the most with in this class are integers, floats, strings, lists and dictionaries.

Ints: Mathematical operations, setting loop iterations, getting results without decimal places

Floats: Mathematical operations, getting results with decimal places

Strings: Printing words to the terminal, taking data in from the user

Lists: Storing data

Dictionaries: Storing data in key-value pairs (when two aspects need to be connected)

What Types Should These Be?:

```
1    a = 4
2    b = 8.9
3    c = "hello"
4    d = [8, 9, 10]
5    e = {"a", "b", "c"}
6    f = None
```

How Do You See Your Responses?:

Print statements will print out what you have in between the parenthesis of the function to the terminal (provided the correct syntax is used)

EX:

Print Statements As Tests:

If you are experiencing issues with your code, one of the fastest ways to troubleshoot where this issue is occurring is with a well placed print statement. This can isolate where the issue if you are having trouble reading the error messages.

How To Take User Input:

```
1    def user_input():
2        name = input("Hey, what's your name?: ")
3        print("Hello " + name + "!")
```

Running Your Code:

- Make sure you are in the right directory/folder
  - You can use the command *pwd* to see what directory you are in
  - You can use the command *ls* to see what files are in the folder you are in
- Use the command *python3 **name of your file/code*** to run your program

Common Python Error Messages:

- TypeError
  - Performing an operation that is not supported by the type
  - Iterating through a non-iterable variable
  - Incorrect argument types for built-in functions
- NameError
  - Name of a variable has not been defined
- SyntaxError
  - Unclosed strings
  - Indentation issues
  - Misusing an assignment operator
  - Misspelling keywords
  - Missing brackets or parenthesis (ect.)