Python Workshop - Intermediate

...

INNOVATORS

S.I.F.T.

InnoGators

- Design Organization
  - Help members cultivate their technical skills through design experience and collaboration.
  - Working with professors and companies to make their ideas come true!
- Design Projects:
  - 3D Smart Filament Recycler
  - Radiation Plume Tracking Drone
- www.innogators.weebly.com
    - Go to the New Members tab if you're interested!

S.I.F.T.

- Consulting Organization
  - Work with real businesses
  - Case Competitions
- Product Management, coding, business principles
- We will be recruiting this coming spring
  - Mentors
  - Project Managers
  - Analysts
- Questions?
  - Email networking.sift@gmail.com

Object-Oriented Programming

- Creates a shell which has preset properties determined by the coder
  - Helps the user easily create a database or list of objects with similar properties
  - Reusing code for repetitive instances
- Good for hierarchy (Inheritance)
  - Example: Create a Hat object which is an instance of the Clothing object and inherits the properties of said clothing object

Classes

- Classes are used to combine functions and variables into a single structure.
  - A template for objects
- You can have multiple objects that are of the same class but they can have their own unique values for the class' variables.
  - A square, triangle, and octagon (objects) are all shapes (class) and have a number of sides (variable), but have different numbers of sides.
  - To access an object's variables, you type "Object_name.variable"
- Syntax:

```
class ClassName:
        <statement-1>
        .
        .
        .
        <statement-N>
```

Example:

```
class House:
    num_people = 3
    bedrooms = 2
    bathrooms = 3

print("The house has " + str(House.bedrooms) + " bedrooms and " + str(House.bathrooms) + " bathrooms")
```

```
The house has 2 bedrooms and 3 bathrooms
```

Functions

- A function is a block of code that executes a sequence of statements to perform a task.
  - Functions can have a parameter or parameters to pass in values to help perform its task
  - Functions can also have a return value to generate a value for use
- Syntax:

```
def function_name(argument_one, argument_two, ..., argument_n):
        body of function outlining whatever it does

        return output (if function returns anything)
```

- To call a function, you just type the function's name and pass in the arguments it needs, if any
- Example (name generator):

Code:

```
def name_gen(first_name, last_name):
    return first_name + " " + last_name

first = input("Enter your first name: ")
last = input("Enter your last name: ")
full_name = name_gen(first, last)
print("Your full name is: " + full_name)
```

Function call

Output:

```
Enter your first name: Keanu
Enter your last name: Budham
Your full name is: Keanu Budham
```

Lists

- Allow for multiple items in one variable
- Ordered, mutable, and allow duplicate values
- Support any data type even mix of different data type.
- Can access values in list through indexing, starting from 0
- len() function call will give the length of a list
- .append() and .remove() for adding and removing items
- https://docs.python.org/3/tutorial/datastructures.html
- Examples
    - arr_num = [1, 2, 3, 4]
    - arr_string = [“hello”, “world”]
    - arr_num[0] -> 1
    - Arr_string[1] -> “world”
    - len(arr_num) -> 4
    - arr_num + arr_string = [1, 2, 3, 4, “hello”, “world”]
    - arr_num[1:3] = [2, 3]
    - arr_num.append(5) -> [1, 2, 3, 4, 5]
    - arr_string.remove(“hello”) -> [“world”]

Tuples

- Very similar to lists, however they are immutable (cannot change, add, or remove items)
- All other functionality is the same for tuples.
- They are initialized with parentheses rather than square brackets.
- https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences
- Examples
  - single_tuple = (1,)
  - tuple_num = (1, 2, 3, 4)
  - tuple_string = (“hello”, “world”)
  - tuple_num[0] = 1 # will produce an error
  - len(tupel_num) = 4
  - tuple_num + tuple_string = (1, 2, 3, 4, “hello”, “world”)
  - arr_num[1:3] = (2, 3)
  - arr_num.append(5) # will produce an error
  - list(tuple_num) -> [1, 2, 3, 4] # list() commands converts tuple to a list
  - tuple([1, 2]) -> (1, 2) # vice versa

Dictionaries

- A collection that is unordered, changeable, and indexed.
- Consists of key, value pairs and mappings from one data type to another
- https://docs.python.org/3/tutorial/datastructures.html?highlight=dictionary#dictionaries
- Examples
  - car = { “brand” : “Ford”, “model” : “Mustang”, “year” : 1987}
  - car[“year”] -> 1987
  - len(car) -> 3
  - car[“color”] = “red” -> { “brand” : “Ford”, “model” : “Mustang”, “year” : 1987, “color” : “red”}
  - car.pop(“brand”) -> { “model” : “Mustang”, “year” : 1987}
  - car.keys() -> [“brand”, “model”, “year”]
  - car.values() -> [“Ford”, “Mustang”, 1987]
  - “brand” in car -> True

Sets

- Unordered collection of items
  - Unique items (no duplicates)
  - Each set element is immutable (cannot be changed)
  - Set as whole can be added to or removed from
- Perform mathematical set operations
- https://docs.python.org/3/tutorial/datastructures.html?highlight=dictionary#sets

```
# Different types of sets in Python
# set of integers
my_set = {1, 2, 3}
print(my_set)

# set of mixed datatypes
my_set = {1.0, "Hello", (1, 2, 3)}
print(my_set)
```

Output

```
{1, 2, 3}
{1.0, (1, 2, 3), 'Hello'}
```

Importing Libraries

- Accesses additional functions
  - Ex: math, random, etc
  - Python Standard Library, Many 3rd Party Libraries for AI, Web Dev, Game Dev, Data Science
  - https://docs.python.org/3/library/
  - https://wiki.python.org/moin/UsefulModules
- Must import every time you start a new project

```
import datetime

x = datetime.datetime.now()
print(x)
```

Break Out Room Practice

Download the template uploaded in the chat!

In this project you will create a virtual phone book which will keep track of new entries and be able to print out the phone book for the user

- https://medium.com/@mardiyyah/building-a-simple-phonebook-learnpythonthroughprojects-series-10-af56d527f463