Python Class

Take 3

PyCharm, Python, Github

Let’s Review Some Terms

- Variable
- List
- Dictionary
- Function
- IF Statements
- For / While Loops
- Class
- Object

Etc

- What are some different types of functions?
  - Built-In Functions
    - print()
    - input()
  - User-defined Functions
  - Recursive functions
  - Lambda Functions
  - Higher Order Functions
- args vs kwargs
- What is OOP in Python?

A Mini Project

Collatz conjecture (3x+1)

This is a mathematical theorem that states that - for any given positive, whole integer, repeating one of two arithmetic operations will eventually transform every positive integer into 1.

- If the number is even, divide it by two.
- If the number is odd, triple it and add one.

This is a pretty famous, and unsolvable math problem. And for today's lesson we are going to … not solve it, obviously. But we can use some of the python technique we have learned, to write a small program to help us determine different aspects around 3x+1.

What we want to be able to do

- Accept a user input for any POSITIVE integer
  - This means to add checks for non integer values, non positive values, as well as floats.
- Apply one of the two rules to integer
  - If the number is even, divide it by two.
  - If the number is odd, triple it and add one.
- Take the result from that and reapply the rules until we hit 1.

From there we would like to:

- Output the entire sequence.
- Output how many times results it takes for us to get to 1.
- Output how many results are even before hitting 1.
- Output how many results are odd before hitting 1.
- Output the highest integer achieved before hitting 1, when applying the rules.

BONUS : Edit user input section to allow for MULTIPLE inputs and have the output send a CSV file with one row per answer.

HomeWork

- Finish our mini review project.
- 1 Hour of Udemy, Complete Python Bootcamp Zero to Hero Section 8
- Complete the following reading assignments:
  - In Depth w/ Attributes
  - In Depth with Methods
- OOP Exercise 1
- Create a Class (ANYTHING you want):
  - Ensure your class has *BOTH*:
    - Class specific Attributes
    - Object specific Attributes
      - For the object specific attributes ensure you ensure that the variable types are specified (ie - I cannot put a string where the answer would be expect to be a integer)
  - Create 2-3 methods for your class
    - Ensure that you use methods that require self as well as a user specified variable.
- Create a Python class called SalesPerson that has the following:
  - Dynamic variables called firstName and lastName that provide the salesperson's first and last name
  - A static variable (starts with 0) called sales, to track their sales over time.
  - A method called makeSale that takes an *integer* and adds that integer to sales.
  - A method called salesReport that prints a string saying "My total sales are {sales}!"
    - Once you have all of this, create a SalesPerson instance with firstName = 'Annie' and lastName = 'Mae' and have them make a sale of 250, and run the saleReport method.
    - Create a second SalesPerson instance with firstName = "Manda" and lastName = "Lynn" and have them make a sale of 150.
    - Create a third SalesPerson instance with firstName = "Dick" and lastName = "Biscuits" and have them make a sale of 6969.
  - BONUS create a method that will print all instances of SalesPerson in order of their sales made greatest to least.
- Upload the last three bullet points to your HW repositories in Github.