Penn Engineering
ONLINE LEARNING

Assignment 4.7: Classes and Objects

Summary

In this graded assignment, you will create and work with Python objects.

Learning Outcomes

In completing this assignment, you will:

- Define Python objects based on a plain-English description
- Implement a program that performs a calculation using Python objects to represent data

Description

In a governmental election, campaign officials may want to know what percentage of the population voted in the previous election, so that they can decide whether to focus voter turnout efforts in that area in order to encourage more people to vote.

In this activity, you'll complete the program below so that it determines the name of the county that had the highest voter turnout in a previous election, as well as the percentage of the population who voted. The data we have provided is based on voter turnout for the 2004 US Presidential election in the state of Pennsylvania:
https://sites.allegheny.edu/vta/pennslyvania-counties-archive/

```
1   # implement County class here
2
3   def highest_turnout(data) :
4
5     # implement the function here
6
7     return # modify this as needed
8
9   # your program will be evaluated using these objects
10  # it is okay to change/remove these lines but your program
11  # will be evaluated using these as inputs
12  allegheny = County("allegheny", 1000490, 645469)
13  philadelphia = County("philadelphia", 1134081, 539069)
14  montgomery = County("montgomery", 568952, 399591)
15  lancaster = County("lancaster", 345367, 230278)
16  delaware = County("delaware", 414031, 284538)
17  chester = County("chester", 319919, 230823)
18  bucks = County("bucks", 444149, 319816)
19  data = [allegheny, philadelphia, montgomery, lancaster, delaware, chester, bucks]
20
```

```
21 result = highest_turnout(data) # do not change this line!
22 print(result) # prints the output of the function
23 # do not remove this line!
```

When you run the starter code that we have provided, you will get a NameError because the County class is not defined. So first, implement the County class starting on line 1. The County class should have three attributes: “name”, “population”, and “voters”. The constructor should accept the parameters in that order and should set the attributes accordingly. Keep in mind that the constructor function must be named “__init__” with two underscores before and two underscores after the word “init”.

Note that lines 4-10 of the starter code are already using the County class and creating County objects, using the name, population, and voters arguments.

Once you have implemented the County class and a constructor with the appropriate number of parameters, the error message will go away and the program will print “None”, since the highest_turnout function does not yet have a return value.

Now complete the implementation of the “highest_turnout” function so that it does the following:

- First, find the County that has the highest turnout, i.e. the highest percentage of the population who voted, using the objects’ population and voters attributes
- Then, return a tuple containing the name of the County with the highest turnout and the percentage of the population who voted, in that order; the percentage should be represented as a number between 0 and 1

Use the “Run” button to run the program, which will invoke the highest_turnout function using the “data” argument and then display the results of any “print” functions, as well as the last one which prints the return value of the function. Note that your highest_turnout function should correctly determine the County with the highest turnout for any input list, not just the one we have provided.

Hints:

- *Review previous activities and lessons for examples of iterating over a list and finding the largest value, keeping in mind that you will need to perform some calculations since we’re not simply looking for the largest population or number of voters.*
- *Also review the previous lesson if you need a reminder about the syntax of creating classes and accessing object attributes.*
- *Part of the challenge of this activity is knowing in advance whether your program is generating the correct output. When you submit your work, your code will be graded*

using the seven County objects we have provided, but you can create your own list of County objects and pass them to the highest_turnout function for testing purposes.

- *As in previous activities, don't forget that you can use the "print" function to print out intermediate values as your code is performing operations so that you can see what is happening, and you can use the "Run" button to run the program and see those outputs.*
- *However, unlike lists, when you attempt to print an object, Python will not print the contents/attributes of the object but will by default print the memory address at which it is stored, which probably isn't very helpful to you! So be sure to print the individual attributes that you're interested in.*

You may run into a Python syntax or runtime error while implementing the County class or the highest_turnout function. Here are some of the common ones:

- **NameError:** The message "name 'County' is not defined" means that the class is not properly defined, or is not defined at all. Make sure you're using the correct syntax for defining a class and that the body of the class definition is properly indented.
- **TypeError:** The message "object() takes no parameters" probably indicates that Python could not find the constructor for your class. Be sure you use the correct name and define the function correctly.
- **TypeError:** Error messages of the format "[function] takes [x] positional arguments but [y] were given" or "[function] missing [x] required positional arguments" indicate a mismatch between number of function parameters in the definition and number of arguments when the function is invoked; keep in mind that first parameter to constructor must be "self" even though it's not passed as an argument
- **AttributeError:** A message like "'County' object has no attribute [x]" means that the attribute or function is not defined in the class. This is the same message for class functions, even though it's called "AttributeError".

After you have run your program using the "Run" button and believe that it is producing the correct output, accept the terms of the Coursera Honor Code and then click the "Submit Quiz" button below to submit this assignment and have it graded.

A few seconds after you submit the assignment, you will see the result at the top of the screen. If it reads "Congratulations! You passed!" then you are ready to proceed to the next lesson.

However, if it reads "Try again once you are ready." then this means that the automatic grading program indicated that your program is not correct. You can find the error message from the grading program beneath your code. In that case, click the "Retake" button to go back to the quiz and try again.

If you believe that your code was correct, be sure to click the "Run" button before submitting and inspect the result of the "print" statement that is right before the "return" statement. This allows you to see the output that your code is producing before it is evaluated. If it looks right, but the

automatic grading utility is still indicating that it is incorrect, then post a message on the discussion board to ask for assistance, but please do not post your code so that you do not reveal your solution to other learners.