CSC 242: Introduction to Computer Science II

Assignment 2

SEE D2L FOR DUE DATE

-10 points for not clarifying how you collaborated with on this assignment or not specifying "no one"

Reading

Read Chapter 8 in **Introduction to Computing using Python: An Application Development Focus** by Ljubomir Perković.

Logistics

Please read the Academic Integrity policy beginning this assignment. It is crucial that you understand what kind of collaboration is allowed and what kind is disallowed on assignments.

In this class programming assignments may be completed in consultation with up to two other classmates. You must identify the classmates with whom you collaborate in a comment at the top of the assignment, and the number of collaborators on any assignment **may not exceed two other people**. You must also submit a comment in your submission for each assignment that describes in detail how each collaborator contributed to the assignment. If you did not collaborate with anyone on the assignment, you must include a comment that says that. You may not under any circumstances discuss the assignments with classmates' other than your identified collaborators. Working so closely with anyone other than your identified collaborators Mr. Zoko, or lab assistant, so as to produce identical or near identical code is a violation of the Academic Integrity policy. This policy will be strictly enforced.

Please include the following with your assignment submission:

1. A comment at the top of your Python file identifying any classmates with whom you discussed or in any other way collaborated on the assignment. You may work (directly or indirectly) with **no more than two** other people.
2. Add a comment at the top of your Python file that describes for each person what they contributed to the assignment. This must be at least 2-3 sentences and be **very specific and detailed**.

A submission that does not include a list of collaborators and a comment indicating how you collaborated with classmates will earn a 0. If you worked alone, you must put a comment at the top of your file that indicates that or you will also receive point off. There will be no exceptions to this rule.

Again, you are subject to all the rules specified in the Academic Integrity policy. Please read it carefully before beginning this assignment.

Assignment

Begin the assignment by downloading the template **csc242hw2.py** found on the D2L site. You will complete the classes and function header found there. **You must include appropriate doc strings** (e.g. strings that appear on the line following the class or method header) for the class and functions that clearly and concisely describe what the class and functions are doing. A submission without doc strings will not earn full credit.

Please note that you should never define an instance variable in the class with the same name as a method in the class. For example, in a class you would never create an instance variable such as self.name if there is a name() method in the class. If you do this, it can cause strange behavior with your objects. It also means you won't earn full credit on the assignment.

Problem 1: Implement and test the Engine class (50 points):

Engine Class:

You will implement a class named account that simulates the behavior of an engine. You can assume that any increase or decrease in speed are passed in as numbers. You must not allow someone to withdraw more than they have in the account. You must infer the behavior of the engine based on the screenshots. Some general rules your code must follow:

1) You cannot change the speed on an engine that is off.
2) The speed of a running engine cannot go below 0 or exceed the engine maximum.
3) You cannot turn off an engine that is at a speed above 0.

A complete code template is not provided. You need to fill in the gaps in behavior based on the screenshots.

Note: Anywhere you see text appear on the screen after calling a method, that means the method is printing text to the screen. If you see a number, that means the method is returning the number and idle is just displaying it to the screen.

```
e=Engine()
Traceback (most recent call last):
  File "<pyshell#94>", line 1, in <module>
    e=Engine()
TypeError: Engine.__init__() missing 1 required positional argument: 'name'
e=Engine('Mistubisi')
str(e)
'Mistubisi is off speed is 0 '
e
Engine('Mistubisi',0,'off')
e=Engine('Mistubisi',100)
str(e)
'Mistubisi is off speed is 0 '
e=Engine('Mistubisi',100,'on')
str(e)
'Mistubisi is on speed is 0 '
e.increaseSpeed(50)
50
str(e)
'Mistubisi is on speed is 50 '
```

Note. If the max speed on an engine is set to 0, it can never go above 0.

```
e=Engine('Craftsman',100)
e
Engine('Craftsman',100,'off')
e.turnOn()
Engine turned on.
e.increaseSpeed(101)
Cannot exceed Engine Speed of 100
0
e.increaseSpeed(77)
77
e.increaseSpeed(2)
79
e
Engine('Craftsman',100,'on')
e.turnOff()
Speed must be zero to turn off the engine!
```

```
e.decreaseSpeed(-1)
Speed must be greater than 0.
0
e.decreaseSpeed(0)
Speed must be greater than 0.
0
e.decreaseSpeed(1)
Invalid speed. Cannot go below 0.
0
e.increaseSpeed(1)
Cannot exceed Engine Speed of 0
0
```

```
e.decreaseSpeed(79)
0
e.turnOff()
Engine turned off.
e
Engine('Craftsman',100,'off')
```

```
e.getName()
'Craftsman'
e.getMaxSpeed()
100
e.getSpeed()
0
```

Problem 2: Implement the ability to save and load accounts (50 points):

You must write two functions:

a. saveEngineInfo : takes in a file name and a list of engines. If will write the engines to a file along with the state of the engine. it returns true if it is successful writing the data and false if it isn't. Previous engine information is overwritten in a file if it exists.

```
engines=[]
engines.append(Engine('Honda',100,'on'))
engines.append(Engine('BMW',50,'off'))
engines.append(Engine('Misubishi',80,'off'))
for engine in engines:
    print(engine)
```

```
Honda is on speed is 0
BMW is off speed is 0
Misubishi is off speed is 0
saveEngineInfo('engines.txt', engines)
True
```

Contents of file:

b. loadEngines: takes in a file name and returns a list of engine objects based on the data in the file.

```
loadedEngines = loadEngines('engines.txt')
for engine in loadedEngines:
    print(engine)
```

```
Honda is on speed is 0
BMW is off speed is 0
Misubishi is off speed is 0
```

```
for engine in loadedEngines:
    print(engine.getMaxSpeed())
    print(engine.getName())
    print(engine.getState())
```

```
100
Honda
on
50
BMW
off
80
Misubishi
off
```

Preparing for next week

Think about what issues could arise with bad inputs being passed into our object. What are some conditions we could test? How could we make our class more robust? What are some conditions we might want to test for?

Submitting the assignment

You must submit the assignment using the assignment 2 dropbox on the D2L site. Submit a Python file (csc242hw2.zip) with your short answers in a text file and your two coding solutions in a .py file. implementation in it and comments describing your collaboration status. Submissions after the deadline listed above will be automatically rejected by the system. See the syllabus for the grading policy.

Grading

The assignment is worth 100 points. Any student who does not submit comments in the Python file describing the contributions of each team member or indicating that he/she worked alone will earn a 0 on the assignment.