

Practical Questions

CSCA08 Fall 2017

Question # 1

Create the following class with the appropriate methods.

- a) Create a Book class which has instance variables for the author and title, and methods to set/change these variables
- b) Create a Swimming Pool class takes in a value for the length, width and depth of the pool and which, when printed, gives the volume of the pool.
- c) Create a Pentagon class that takes in the length of one side upon creation, and has methods for getting the perimeter and area.
- d) Create a Country class that takes in a countries name, population, area, and has methods to change and retrieve the values for the three instance variables of your class, as well as to calculate the population density (avg # of people per square m)
- e) Create a Shape class that take in the shape's name, area, perimeter and has a method that returns True is the area of the first Shape object is larger than the area of the Shape object (ex. square1.is_larger(sqaure2))
- f) Create a Course class that takes in the course code, the number of students and the location of the class and has a __str__ method that will return "(course code) is held in (location) and has (student_nuber) of students enrolled"
- g) Create a Student class with variables and methods that are sensible for students

h) (Challenge)

Add a method enroll to your Course class from part f) that takes a Student object (as created in part g). Then add methods to print a full student list, and calculate average GPA for all students in the course.

i) (Challenge)

Create a Point class that takes in the x and y coordinates for that point and has a distance_to_origin method which returns the distance from the origin to this Point object, a distance_between_points method that returns the distance between 2 Point objects and has a slope_between_points method which returns the distance and slope between 2 Point objects

Discussion Questions

Question # 2

2 types of methods that are usually found in different classes are getter, which returns specific instance variables of your class, and setters, which change the values associated with a specific instance variable. What does using getters and setters allow you, as a programmer, to do? Why do you want to use getters and setters instead of allowing outside users access to the instance variables inside your class?

Question # 3

Object Oriented Programming (creating classes and using methods) is different from the programming you have done before in this class. Initially, classes may just look like any other data structure (lists, dictionaries, sets) as they seem to just store information but as you get more experience with classes, you will start to see how they are different and how they are much more powerful. Classes are thought to have attributes, like a Person class is of height x or a country class has population y. Some people have said that “object oriented programming resembles more how humans work than how computers work”? Do you agree or disagree with this statement.

Question # 4

Unlike many other languages, Python doesn't have any mechanism to enforce that instance variables are actually private. Putting an underscore before a name encourages developers not to mess with it, but it doesn't actually stop anyone. What are the advantages/disadvantages of this approach?

Question # 5, The “Logic Question” **(Challenge)**

You have built a device to protect eggs from falling (a classic high school science experiment). Your goal is to test the device and determine the highest floor from which you can drop your device, containing an egg, and have it survive. A few rules:

- This is a completely deterministic process, if the egg breaks from floor n , it will always break from floor n , if it doesn't break from that floor, it will never break from that floor
- If an egg will break from floor n , it will break on all floors $> n$, if it will survive a drop from floor n , it will survive a drop from all floors $< n$
- The machine either fully saves the egg or it does not. So dropping it 100 times from a lower floor doesn't damage the egg in any way, the result will be the same on the 101st drop as it would've been on the first.
- You have no idea how good your device is, it could break from the first floor, or survive from the 10,000,000th floor.

Let's work through a few scenarios (turning the screw):

1. You have a building with 100 floors, and only 1 egg. What is your strategy?
2. You have a building with 100 floors, and as many eggs as you need, but you want to minimize your number of drops. What is your strategy?
3. You have a building with 100 floors, and only 2 eggs. What is your strategy?
4. You have a building with an infinite number of floors, and an infinite number of eggs. What is your strategy?