**Python Advance assignment 5**

**Q1. What is the meaning of multiple inheritance?**

Inheritance is the mechanism to achieve the re-usability of code as one class(child class) can derive the properties of another class(parent class). It also provides transitivity ie. if class C inherits from P then all the sub-classes of C would also inherit from P.  
When a class is derived from more than one base class it is called multiple Inheritance. The derived class inherits all the features of the base case.

**Q2. What is the concept of delegation?**

Delegation is an object oriented technique (also called a design pattern). Let's say you have an object x and want to change the behaviour of just one of its methods. You can create a new class that provides a new implementation of the method you're interested in changing and delegates all other methods to the corresponding method of x.

**Q3. What is the concept of composition?**

Composition is one of the [fundamental concepts in object-oriented programming](https://stackify.com/oops-concepts-in-java/). It describes a class that references one or more objects of other classes in instance variables. This allows you to model a has-a association between objects.

**Main benefits of composition:** Given its broad use in the real world, it’s no surprise that composition is also commonly used in carefully designed software components. When you use this concept, you can:

1. reuse existing code
2. [design clean APIs](https://theamiableapi.com/2012/01/16/java-api-design-checklist/)
3. change the implementation of a class used in a composition without adapting any external clients

**Q4. What are bound methods and how do we use them?**

If a function is an attribute of class and it is accessed via the instances, they are called bound methods. A bound method is one that has ‘[self](https://www.geeksforgeeks.org/self-in-python-class/)‘ as its first argument. Since these are dependent on the instance of classes, these are also known as instance methods.

#### Need for these bound methods

The methods inside the classes would take at least one argument. To make them zero-argument methods, ‘[decorators](https://www.geeksforgeeks.org/decorators-in-python/)‘ has to be used. Different instances of a class have different values associated with them.

**Q5. What is the purpose of pseudoprivate attributes?**

We all know: In Python, there is no similar

private

Such keywords to declare private methods or properties.

To declare a private attribute in Python, you need to add a double underscore before the attribute (but no double underscore at the end), such as:

self.\_\_a

. However, such a method is not really private, but "pseudo private."

Python's pseudo-private attributes actually implement variable name localization through variable name compression (mangling). The rule of variable name compression: add an underscore to the head of the initial variable name, plus the name of the class, and finally the name of the initial variable name.