Q1. What is the meaning of multiple inheritance?

If a child class is inheriting the properties of a single other class, we call it single inheritance. However, if a child class inherits from more than one class, i.e. this child class is derived from multiple classes, we call it multiple inheritance in Python.

Q2. What is the concept of delegation?

Delegation is a design pattern in which an object, called the delegate, is responsible for performing certain tasks on behalf of another object, called the delegator. This can be done by the delegator forwarding method calls and attribute access to the delegate.

Q3. What is the concept of composition?

Composition is a concept that models a has a relationship. It enables creating complex types by combining objects of other types. This means that a class Composite can contain an object of another class Component. This relationship means that a Composite has a Component.

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

A bound method in Python is the method that has an object associated with it. Unbound method doesn't have an object associated with it. A bound method in Python is the method that has an object associated with it. Unbound method doesn't have an object associated with it.

Q5. What is the purpose of pseudoprivate attributes?

This trick can avoid potential name collisions in the instance, but note that it does not amount to true privacy. If you know the name of the enclosing class, you can still access either of these attributes anywhere you have a reference to the instance by using the fully expanded name (e.g., I.\_C1\_\_X = 77). On the other hand, this feature makes it less likely that you will accidentally step on a class's names.