1. What is the concept of an abstract superclass?

Ans: An abstract superclass is a class in object-oriented programming that is declared abstract and is designed to be inherited by subclasses. An abstract class itself cannot be instantiated; it only serves as a blueprint for other classes to inherit from. It often defines a set of methods that must be implemented by its subclasses, while also providing some common behavior or attributes that these subclasses share.

2. What happens when a class statement's top level contains a basic assignment statement?

Ans: In many programming languages, including Python, when a class statement's top level contains a basic assignment statement, it creates a class variable that is shared among all instances of that class. This means that the variable is accessible to all instances of the class and any changes made to it will be reflected across all instances.

3. Why does a class need to manually call a superclass's \_\_init\_\_ method?

Ans: In object-oriented programming, when a class inherits from another class (the superclass or base class), it usually extends or modifies the behavior of the superclass. In some cases, the subclass might have additional attributes or require some specific initialization procedures that are not present in the superclass. Therefore, it becomes necessary for the subclass to call the superclass's \_\_init\_\_ method explicitly to ensure proper initialization of the inherited attributes and behaviors from the superclass.

4. How can you augment, instead of completely replacing, an inherited method?

Ans: To augment, or extend, an inherited method in a subclass without completely replacing it, you can follow these general steps: 1. Call the superclass's method from the subclass. 2. Perform additional actions specific to the subclass after calling the superclass's method.

In most object-oriented programming languages, the super() function is used to call methods of the superclass.

5. How is the local scope of a class different from that of a function?

Ans: The local scope of a class in object-oriented programming refers to the context within which variables and methods are defined and accessed within the class. It includes instance variables, class variables, and methods that can be accessed and manipulated through instances of the class.

On the other hand, the local scope of a function refers to the area within the function where variables are declared and can be accessed. Variables declared within a function are only accessible within that function and cannot be accessed from outside the function.