1. **What is the concept of an abstract superclass?**

Solution - An abstract class is a class, but not one you can create objects from directly. Its purpose is to define how other classes should look like, i.e. what methods and properties they are expected to have.

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

Solution - When a class statement's top level contains a basic assignment statement, it means that a class-level variable is being defined and assigned a value.

3. **Why does a class need to manually call a superclass's __init__ method?**

Solution - In object-oriented programming, when a class inherits from a superclass (also known as a parent class or base class), it may need to manually call the superclass's __init__ method in its own __init__ method. This is done to ensure that the initialization code of the superclass is executed before the initialization code of the subclass.

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

Solution - To augment, or extend, an inherited method in a subclass without completely replacing it, you can follow these steps:

- Define the subclass that inherits from the superclass.
- Declare a method in the subclass with the same name as the method you want to augment in the superclass.
- Inside the subclass method, call the superclass's method using the super() function.
- Add any additional functionality or modifications to the method in the subclass.

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

Solution – In Python, the local scope of a class and a function is different in terms of accessibility and lifespan.

Accessibility:

Class Local Scope: The local scope of a class includes the class attributes and methods, which are accessible within the class definition. These attributes and methods can be accessed by all the methods of the class, including the __init__ method, instance methods, and static methods.

Function Local Scope: The local scope of a function includes the variables and parameters defined within the function itself. These variables and parameters are accessible only within the function and cannot be accessed outside of it.

Lifespan:

Class Local Scope: The local scope of a class exists throughout the lifespan of the class. It is created when the class is defined and remains in memory as long as the class definition is active. Class attributes and methods retain their values and are accessible to instances of the class.

Function Local Scope: The local scope of a function is created when the function is called and is destroyed when the function completes its execution. Local variables and parameters defined within the function exist only during the function's execution and are discarded once the function exits.