1. What is the concept of an abstract superclass?

>>>The purpose of an abstract superclass is to provide common functionality and structure to its subclasses, promoting code reuse and maintaining a consistent interface across related classes.the abstract superclass provides the functionality for the subclasses to get it by execution.the abstract supercalss does not have any instance it uses its subclasses for the execution.

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

>>>Class-level attributes are attributes that are shared among all instances (objects) of the class, rather than being specific to individual instances.

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

>>>In object-oriented programming, when a class inherits from a superclass (also known as a parent class), it acquires the attributes and methods of the superclass. In Python, to ensure that the superclass's initialization code is executed before initializing the attributes of the subclass, the subclass needs to manually call the superclass's \_\_init\_\_ method.

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

>>>when a subclass inherits a method from its superclass, it has the option to augment or extend the behavior of the inherited method without completely replacing it. This process is known as method overriding or method redefinition. Method overriding allows the subclass to provide its implementation of a method that is already defined in its superclass, while still retaining the original functionality defined in the superclass.

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

>>>In Python, the local scope of a class refers to the scope inside the class body, where class-level attributes, methods, and nested classes are defined.

* Class-level attributes are variables defined within the class, outside of any method, and they are shared among all instances (objects) of the class.