1. What is the concept of an abstract superclass?

- It allows you to create a set of methods that must be created within any child classes built from the abstract class. A class which contains one or more abstract methods is called an abstract class. An abstract method is a method that has a declaration but does not have an implementation.

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

- An assignment statement sets and/or re-sets the value stored in the storage location(s) denoted by a variable name; in other words, it copies a value into the variable. In most imperative programming languages, the assignment statement (or expression) is a fundamental construct.

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

- By doing so,we can access those methods of the super-class (parent class) which have been overridden in a sub-class (child class) that inherits from it.

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

- The way to do that in Python is by calling to the original version directly, with augmented arguments.

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

- In class, if the variable is declared without self then it is accessible within that function only. However if it declared using self like self.variable_name = 'somevalue', then it is accessible via any object but not via the class name.

- Whereas, if a variable is declared within a function then it is a local variable and is accessible to that function only.