Q1. What is the purpose of Python's OOP?

**Ans.** It aims to implement real-world entities like inheritance, polymorphisms, encapsulation, etc

Q2. Where does an inheritance search look for an attribute?

**Ans.**The inheritance search is simply a search of the tree from bottom to top looking for the lowest occurrence of an attribute name.

Q3. How do you distinguish between a class object and an instance object?

**Ans.**Class is the blueprint of an object. It is used to declare and create objects. Object is an instance of class.

Q4. What makes the first argument in a class’s method function special?

**Ans**. The calling process is automatic while the receiving process is not (its explicit). This is the reason the first parameter of a function in class must be the object itself. Writing this parameter as self is merely a convention. It is not a keyword and has no special meaning in Python.

Q5. What is the purpose of the \_\_init\_\_ method?

**Ans.** It is called as a constructor in object oriented terminology. This method is called when an object is created from a class and it allows the class to initialize the attributes of the class

Q6. What is the process for creating a class instance?

**Ans.** When you create an object, you are creating an instance of a class, therefore "instantiating" a class. The new operator requires a single, postfix argument: a call to a constructor. The name of the constructor provides the name of the class to instantiate. The constructor initializes the new object

Q7. What is the process for creating a class?

**Ans.** We use the class keyword and write the class name you went to create .

**For Eg.** class Student

Q8. How would you define the superclasses of a class?

**Ans.**A superclass is the class from which many subclasses can be created. The subclasses inherit the characteristics of a superclass.