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

ANS)Many developers use OOP because it makes your code reusable and logical, and it makes inheritance easier to implement. It follows the DRY principle, which makes programs much more efficient.

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

ANS)In Python, inheritance happens when an object is qualified, and involves searching an attribute definition tree (one or more namespaces). ... attr where object is an instance or class object, Python searches the namespace tree at and above object , for the first attr it can find.

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

ANS)A class is a template for creating objects in program whereas the object is an instance of a class.

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

ANS) First argument is the class itself which is passed implicitly. Always return a valid object from \_\_new\_\_() . Not mandatory, but its main use is to create and return an object.

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

ANS) "\_\_init\_\_" is a reseved method in python classes. 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.r

**Q7. What is the process for creating a class?**

ANS) Creating a new class creates a new type of object, allowing new instances of that type to be made. Each class instance can have attributes attached to it for maintaining its state. Class instances can also have methods (defined by their class) for modifying their state.

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

ANS) A superclass is the class from which many subclasses can be created. ... The superclass is also known as the parent class or base class.