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

Ans:- The oop is used to develop an application. In Python we easily create a class and object using the oop concept. The classes provide bunding data and functionality together.

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

Ans: When you access an attribute of object, So in python it first the attribute in the object's instance dictionary. If the attribute is not found there, Python then searches for the attribute in the class hierarchy of the object. This search is known as the "attribute lookup" or "inheritance search".

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

Ans: For class→ It is defined using class keyword.A class object is the definition of the class, which includes the attributes and methods that all instances of the class will have. An instance object is a specific occurrence of the class, created from the class definition, with its own unique set of attribute values.

For an instance → A class object is created once, usually when the program is loaded into memory. An instance object is created whenever a new object is needed, using the new keyword or some other mechanism provided by the programming language.

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

ANS : In object-oriented programming, the \_\_init\_\_ method (short for "initialize") is a special method that gets called automatically when an instance of a class is created. Its purpose is to initialize the instance's attributes and perform any other setup that is required before the instance can be used.

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

Ans: First we need to define the class and then we need to write the class name. To create an instance of the class, you need to call the class as if it were a function. This creates a new instance of the class. Once you have created an instance of the class, you can set its attributes by accessing them using the dot notation and assigning a value to them.

Q7. What is the process for creating a class?

Ans :- First need to define class and then we need to write the class name. And then need to define class attributewhich are shared by all instances of the class. These are defined outside of any method and are typically initialized to default values. You can define methods within the class definition, which are functions that can be called on instances of the class. Methods are defined using the same syntax as regular functions, but they always take the self parameter as their first argument, which refers to the instance of the class that the method is being called on.

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

Ans : In Python, a superclass is a class that another class inherits from. To define the superclasses of a class, you need to include them in the class definition using parentheses after the class name.