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

Sol: The purpose of oops concept in python:

OOPs is an object oriented programming language which is designed in using classes and objects. Other oops concepts are: inheritance, polymorphism, encapsulation, data abstraction etc.

Object Oriented Programming is easy to understand and implement.

As the classes are shareable, the coders can use the code in multiple scenarios instead of rewriting them where Packages and Modules come into picture.

Data is safe and secure with data abstraction as unnecessary information is hidden from user.

Polymorphism allows the same interface for different objects, so coders can write code efficiently.

Encapsulation is that the coder can never modify the existing details with the help of access modifiers like Public, Private, Protected

Exception Handling is used to handle the errors using exceptions.

These are the concepts that make coders to deal with objects.

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

Sol: when an inheritance is given with an attribute, it searches in its respective parent class to acquire its properties.

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

Sol: Class Object: A class is like a object constructor or a blueprint for creating objects.

Instance Object: An individual object for a certain class.

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

Sol: The first argument of a method is called self. It is not a keyword and just a named convention. It has no special meaning in python.

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

Sol: The \_\_init\_\_ method is Python constructor in Object Oriented approach.

It is called everytime an object is created for a class. It is used to store objects attributes for a class and it is only used within classes.

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

Sol: An individual object for a class is called instance/ variable and the for creating objects for a class is:

* We can use class named functions followed by attributes to create objects for a class.

Q7. What is the process for creating a class?

Sol: A class is like a blueprint or object constructor for creating objects.

The process for creating an object is :

* Use class keyword, along with classname.
* Then, we can define method for respective class using \_\_init\_\_() and self to point the object to the class.
* Return respective output which is needed.

Q8. How would you define the super classes of a class?

Sol: The class from which a child /derived class is inherited is called super class.