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

Ans:

**the purpose of object oriented programming is to create a code which is organised and usable.**

**The code in OOP is written in terms of classes and methods which can be reused multiple times within any piece of code.**

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

Ans: **it first looks for the attributes in the instance object, then in the class the instance was created from, then in all superclasses.**

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

Ans: **all the class objects are build based on the blueprint of a particular class. An instance is a virtual copy of the object. A class object is created using a constructor of the class and that object will then be called an instance of the class.**

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

Ans: **The first argument is a pointer which helps any particular instance of the class to bind with the class and the methods of the class. The pointer is a reference to the current instance of the class. Usually pointer is denoted by ‘self’ but it can be denoted by any other name.**

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

Ans: **init method is a constructor which is used for initialization. It is called automatically when the instance of the class is created. So the purpose of this method is to pass the required parameters to the class when any class object is created.**

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

**Ans**: **Class instance can be creating by assigning the class to the object and if the constructor is used in the class then it requires you to pass the parameters as well.**

Q7. What is the process for creating a class?

Ans: **The process of creating a class is to use the ‘class’ keyword and then define methods inside the class including constructor which is not entirely compulsory.**

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

Ans: **the superclass is defined the same way as other classes are defined but while inheriting the super class in the child class, the superclass name is passed in the child class parameter and super() function is used in the subclass.**