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

OOPs has many features like inheritance, polymorphism but main aim is to keep various objects in different class and call them as and when needed to do a task. We have an option to create a public/private relationship.

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

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

Class would be the blue print with an object is built on the blue print. But Instance would be a virtual copy of the object.

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

It automatically receives a special first argument, self, that provides a handle back to the instance to be processed. Methods with two underscores at the start and end of names are special methods. The \_\_init\_\_() method is called immediately after an instance of the class is created.

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

Once an object is created in a class this method allows class to initialize the attributes in the class.

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

There are 3 steps to it. Variable declaration with variable name with object type.

Q7. What is the process for creating a class?

Creating a class is like creating a blue print of an action as how it has to be performed.

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

A superclass is a class from where many sub classes can be created.