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

In Python, object-oriented Programming (OOPs) is a programming paradigm that uses objects and classes in programming. It aims to implement real-world entities like inheritance, polymorphisms, encapsulation, etc.

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

In Python, inheritance happens when an object is qualified, and involves searching an attribute definition tree (one or more namespaces). Every time you use an expression of the form object.

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

An object is an instance of a class. When a class is created, no memory is allocated. Objects are allocated memory space whenever they are created

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

Self is the first argument to be passed in Constructor and Instance Method. By using the “self”  we can access the attributes and methods of the class in python. It binds the attributes with the given arguments.

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

The \_\_init\_\_ method lets the class initialize the object's attributes

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

To create instances of a class, you call the class using class name and pass in whatever arguments its \_\_init\_\_ method accepts.

Q7. What is the process for creating a class?

In Python, a class can be created by using the keyword class, followed by the class name

class Employee:

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

The class from which a class inherits is called the parent or superclass