Q1. What is the concept of a metaclass?

Ans: A metaclass in Python is a class of a class that defines how a class behaves.

Q2. What is the best way to declare a class's metaclass?

Ans: The first argument is the metaclass itself.

The second argument is the class name.

The third argument is the superclasses (in the form of tuple)

The fourth argument is the attributes of class (in the form of dictionary)

Q3. How do class decorators overlap with metaclasses for handling classes?

Ans: A metaclass in Python is a class of a class that defines how a class behaves. A class is itself an instance of a metaclass. A class in Python defines how the instance of the class will behave.

Q4. How do class decorators overlap with metaclasses for handling instances?

Ans: Metaclasses operate at the lower level and allow you to change the structure or behavior of the class, like the class methods, attributes, and inheritance. Decorators, however, are used to modify the functions' behavior. They allowed you to add functionality to the existing functions without changing the code.