Q1. What is the relationship between classes and modules?

Ans : The class is a blueprint for creating objects that have specific methods and attributes. And the modules is a file containing Python definitions and statements.

The relationship between classes and modules is that you can define a class in a module, and then import that module into another module or script to use the class.

Q2. How do you make instances and classes?

Ans : For instances →To create instances of the class, you need to call the class as if it were a function. This creates a new instance of the class.

And For classes →you need to define the class that you want to create instances of. This involves specifying the class name, attributes, and methods.

Q3. Where and how should be class attributes created?

class nikunj:

num = 0 # Class attribute

def \_\_init\_\_(self, city, mo):

self.city = city

self.mo = mo

nikunj.num += 1 # Accessing the class attribute and incrementing it

Q5. What does the term “self” in a Python class mean?

Ans : When you call a method on an instance of a class, Python automatically passes the instance as the first argument to the method. This argument is conventionally named self. self allows you to access the instance's attributes and methods from within the method.

Q6. How does a Python class handle operator overloading?

Ans: Python supports operator overloading, which means that operators such as +, -, \*, and / can be overloaded in Python classes. When an operator is used on an object of a user-defined class, Python checks if that operator is defined in the class, and if so, it invokes the corresponding magic method or dunder method to perform the operation.Python class handles operator overloading by defining the corresponding magic method for each operator. When an operator is used on an object of the class, Python invokes the corresponding magic method to perform the operation.

Q7. When do you consider allowing operator overloading of your classes?

Ans -> No idea

Q8. What is the most popular form of operator overloading?

Ans : The plus (+) and (\*) are the most popular form of operator overloading.

Q9. What are the two most important concepts to grasp in order to comprehend Python OOP code?

Ans : classes and objects.