1.

Modules are collections of methods and constants. They cannot generate instances. Classes may generate instances (objects), and have per-instance state (instance variables).
Modules may be mixed in to classes and other modules. The mixed in module’s constants and methods blend into that class’s own, augmenting the class’s functionality. Classes, however, cannot be mixed in to anything.
A class may inherit from another class, but not from a module.
A module may not inherit from anything

2.

Call ClassName() to create a new instance of the class ClassName . To pass parameters to the class instance, the class must have an __init__() method. Pass the parameters in the constructor of the class.

3.

```
class ObjectClass():
def __init__(self):
self. attribute1 = "attribute1"
def newAttr(self, attr):
setattr(self, attr, attr)
objectClass = ObjectClass()
print(objectClass. attribute1)
setattr(objectClass, "newAttribute", "new attr")
```

4.

An instance attribute is a Python variable belonging to one, and only one, object. This variable is only accessible in the scope of this object and it is defined inside the constructor function, __init__(self,..) of the class.

5.

self represents the instance of the class. By using the “self” keyword we can access the attributes and methods of the class in python. It binds the attributes with the given arguments.

6.

The operator overloading in Python means provide extended meaning beyond their predefined operational meaning. Such as, we use the "+" operator for adding two integers as well as joining two strings or merging two lists. We can achieve this as the "+" operator is overloaded by the "int" class and "str" class.

7.

For an operator to be overloaded, at least one of the operands must be a user-defined object. Only existing operators can be overloaded. You cannot overload new operators.

8.

A very popular and convenient example is the Addition (+) operator. Just think how the '+' operator operates on two numbers and the same operator operates on two strings. It performs “Addition” on numbers whereas it performs “Concatenation” on strings.

9.

here are four fundamental concepts of Object-oriented programming – Inheritance, Encapsulation, Polymorphism, and Data abstraction.