

Classes and Objects in python

A class is a blueprint or a template for creating objects, providing initial values for state (member variable or attributes), and implementations of behavior (member function or methods). The user defined objects are created using the class keyword.

Creating a class:

Let us now create a class using the class keyword

```
class Python:  
    library="Django"  
    version=3.10
```

Creating an Object:

Object is the instance of the class used to access the properties of the class now let's create of the class.

Example:

```
class Python:  
    library="Django"  
    version=3.10  
  
py=Python()  
print(py.library)  
print(py.version)
```

Self-parameter:

This self-parameter is a reference to the current instance of the class, and is used to access variables that belongs to the class.

It must be provided as the extra parameter inside the method definition.

Example:

```
class Person:
    name="Munawar"
    occupation="Programmer"
    salary=40000

    def information(self):
        print(f"{self.name} is a {self.occupation}")
b=Person()
b.name="Kamal"
b.occupation="Developer"

b.information()
```

Source Code

```
class Person:
    name="Munawar"
    occupation="Programmer"
    salary=40000

    def information(self):
        print(f"{self.name} is a {self.occupation}")

p=Person()

p.name="Munawar Johar"
p.salary=50000
#print(p.name,p.occupation)
p.information()

b=Person()
b.name="Kamal"
b.occupation="Developer"

b.information()

c=Person()
c.information()
```

Thank You

