**Python Classes:**

1. **Read about how to define a class in python, have a member variable & functions for class**

class Person:

def \_\_init\_\_(self,age):

self.age = age

Def Print(self):

print("Hello my age is " ,self.age)

p = Person(36)

p.Print()

**Output:**

Hello my age is 36

1. **Constructor of class**

class Person:

# default constructor

def \_\_init\_\_(self):

self.name = "Sangeet"

self.age= 23

p= Person()

print(p.name)

print(p.age)

**Output:**

Sangeet

23

**3. Create an object for the class**

class MyClass:

x = 5

p = MyClass()

print(p.x)

**Output:**

**5**

**4. Call the class method to modify the member variable**

**class Person:**

**def \_\_init\_\_(self,name,age):**

**self.name = name**

**self.age = age**

**def Print(self):**

**print("Hello my name is " ,self.name)**

**print("Hello my age is " ,self.age)**

**def update(self,newName,newAge):**

**self.name = newName**

**self.age = newAge**

**p = Person("sangeet",23)**

**p.Print()**

**p.update("kumar",25)**

**p.Print()**

**Output:**

**Hello my name is sangeet**

**Hello my age is 23**

**Hello my name is kumar**

**Hello my age is 25**

**5. Create a class, Car. Have attributes like weight, max\_speed, number\_of\_wheels, service\_dates(List of datetime) and methods like get\_max\_speed(), update\_max\_speed(), add\_service\_dare(), get\_service\_dates() and other methods for fetching and modifying attribute values.**

**class Car:**

**def \_\_init\_\_(self,wt,speed,wheels,dates):**

**self.weight = wt**

**self.max\_speed = speed**

**self.number\_of\_wheels = wheels**

**self.service\_dates = dates**

**def get\_max\_speed(self):**

**print("max speed is " ,self.max\_speed)**

**def update\_max\_speed(self, newSpeed):**

**self.max\_speed = newSpeed**

**def add\_service\_dates(self, newDates):**

**self.service\_dates.append(newDates)**

**def get\_service\_dates(self):**

**print(self.service\_dates)**

**c = Car(23,120,4,["01/12/2021"])**

**c.get\_max\_speed()**

**c.update\_max\_speed(140)**

**c.get\_max\_speed()**

**c.get\_service\_dates()**

**c.add\_service\_dates("02/12/2021")**

**c.get\_service\_dates()**

**OUTPUT:**

**max speed is 120**

**max speed is 140**

**['01/12/2021']**

**['01/12/2021', '02/12/2021']**

**6. Create multiple objects for this class with different values passed to the constructor.**

**class Person:**

**def \_\_init\_\_(self,name,age):**

**self.name = name**

**self.age = age**

**def Print(self):**

**print("Hello my name is " ,self.name)**

**print("Hello my age is " ,self.age)**

**def update(self,newName,newAge):**

**self.name = newName**

**self.age = newAge**

**p1 = Person("sangeet",23)**

**p1.Print()**

**p1.update("kumar",25)**

**p1.Print()**

**p2= Person("harish",24)**

**p2.Print()**

**OUTPUT:**

**Hello my name is sangeet**

**Hello my age is 23**

**Hello my name is kumar**

**Hello my age is 25**

**Hello my name is harish**

**Hello my age is 24**

**7. Extending a base class.**

**8. Override a method.**

**10. Create a base class Vehicle. Extend the class and create Car and Bus. create objects for these classes.**

**class Vehicle(): #base class**

**def description(self):**

**print("I'm a Vehicle!")**

**def goodVehicle(self):**

**print("vehicle is in good condition")**

**# subclass**

**class Car(Vehicle):**

**def description(self):**

**print("I'm a car!")**

**class Bus(Vehicle):**

**def description(self):**

**print("I'm a bus!")**

**# create an object from each class**

**v = Vehicle()**

**c = Car()**

**b = Bus()**

**c.goodVehicle()**

**#override**

**v.description()**

**c.description()**

**b.description()**

**OUTPUT:**

**vehicle is in good condition**

**I'm a Vehicle!**

**I'm a car!**

**I'm a bus!**

**9. Call an overwritten method and non overwritten method from the object of the extended class.**

**class Vehicle(): #base class**

**def description(self):**

**print("I'm a Vehicle!")**

**def goodVehicle(self):**

**print("vehicle is in good condition")**

**# subclass**

**class Car(Vehicle):**

**def description(self):**

**print("I'm a car!")**

**class Bus(Vehicle):**

**def description(self):**

**print("I'm a bus!")**

**# create an object from each class**

**v = Vehicle()**

**c = Car()**

**b = Bus()**

**c.goodVehicle()**

**#override**

**v.description()**

**c.description()**

**OUTPUT:**

**vehicle is in good condition**

**I'm a Vehicle!**

**I'm a car!**