15/6

Types of methods:

(i) instance method
(ii) class method
(iii) Static method

→ Instance method :-

it is a regular method like what we use

Eg: class Demo:
    def m1(self):
        print("I'm a instance method")

o=Demo()

[There is no need of creation of object as there is no instance variable]

object creation is mandatory if we have instance variable.

Class Method :-

```
class Demo:
    a=10 #class variable
    @classmethod
    def m1(self): #class method   o/p: 10
        print(Demo.a)

Demo.m1()
```

Between class & method, if a variable is defined it is known as class method

Static Method :-

```
class Demo:
    @staticmethod
    def m1():
```

print("I'm a static method")

Demo.m1()

o/p:- I'm a static method.

If we don't have instance variable then we can use static

In @staticmethod @staticmethod is not necessary but in classmethod @classmethod is necessary.

Ex:- class Demo:

    def m1(self):

        print("I'm instance method")

Demo.m1()

o/p: Error

Ex-2:- class Demo():

    @staticmethod

    def m1():

        print("I'm static method")

Demo.m1()

o/p:- I'm static method.

To represent in coding we use decorator.

Ex-3:- class Demo:

```
    a=10
    @staticmethod
    def m1():
        print(Demo.a)

Demo.m1()
```

O/p:- 10

Ex-4:- class Demo:

```
    a=10
    @classmethod
    def m1(self):
        print(Demo.a)

Demo.m1()
```

O/p:- 10

In classmethod self is mandatory.

Ex-5:- class Demo:

```
    def m1(self):
        self.a=10 #instance variable
    def display(self):
        print(self.a)

o=Demo()
o.m1()
o.display()
```

O/p:- 10

Ex-6:-
```
class Demo:
    class_variable=20
    def m1(self):
        self.a=10 #instance variable
    def display(self):
        print(self.a)
    @classmethod
    def dis(self):
        print(Demo.class_variable)
o=Demo()
o.m1()
o.display()
Demo.dis()
```
O/p:- 10
20

Let continue above program as
```
    @staticmethod
    def m2():
        print("I'm static method")
o=Demo()
o.m1()
o.display()
Demo.dis()
Demo.m2()
```
O/p:- 10
20
I'm Static method.