



Parent class method
Parent static method

Case-3: From child class, class method we cannot access parent class instance methods and constructors by using super() directly (but indirectly possible). But we can access parent class static and class methods.

```
1) class P:
2)     def __init__(self):
3)         print('Parent Constructor')
4)     def m1(self):
5)         print('Parent instance method')
6)     @classmethod
7)     def m2(cls):
8)         print('Parent class method')
9)     @staticmethod
10)    def m3():
11)        print('Parent static method')
12)
13) class C(P):
14)     @classmethod
15)     def m1(cls):
16)         #super().__init__()--->invalid
17)         #super().m1()--->invalid
18)         super().m2()
19)         super().m3()
20)
21) C.m1()
```

Output:

Parent class method
Parent static method

From Class Method of Child class, how to call parent class instance methods and constructors:

```
1) class A:
2)     def __init__(self):
3)         print('Parent constructor')
4)
5)     def m1(self):
6)         print('Parent instance method')
7)
8) class B(A):
9)     @classmethod
10)    def m2(cls):
11)        super(B,cls).__init__(cls)
12)        super(B,cls).m1(cls)
```