

Encapsulation in Python

Saturday, August 20, 2022 11:04 AM

A

How to access a Private Method?

```
class Car:
    def __init__(self):
        self.__updateSoftware()

    def drive(self):
        print('driving')

    def __updateSoftware(self):
        print('updating software')

redcar = Car()
redcar.drive()
redcar._Car__updateSoftware()
```

Private method can be called using
redcar._Car__updateSoftware()

In python the private is describe by "__" two underscores but in it is A constructor not an private methods

How to access a Private Method?

To change the value of a private variable, a setter method is used

```
def setMaxSpeed(self,speed):
    self.__maxspeed = speed

redcar = Car()
redcar.drive()
redcar.__maxspeed = 10 # will not
change variable because its private
redcar.setMaxSpeed(320)
redcar.drive()
```

So as we all know the private methods cannot be accessed by the user

So we will call it by class as a setter in a particular syntax "_class name__private method name ()"

```
1 class SomeClass:
2     def public(self):
3         print("Public Function")
4
5     def __private(self):
6         print("Private Function")
7
8 obj = SomeClass()
9
10 obj.public()
11 obj._SomeClass__private()
12
```

Here we get the output

Public Function

Private Function