

python3使用class变量

有些时候可能一些公共的函数需要写成classmethod的方式，即不用把class给先实例化之后再使用，可以直接采用《类名.方法名》的方式使用。

但是这种场景如果需要需要先调一个实例化的底层变量，然后包进当前的class以classmethod的方式进行使用，此时就需要把这个实例化的底层变量以class 变量的方式存在，不然这个实例化信息无法传递到class method那。

举例：

```
class b(object):
    def __init__(self, name):
        self.name = name
    def print(self):
        print(f'haha, b.name = {self.name}')
```

```
class a(object):
    b_inst = None
    def __init__(self, name):
        a.b_inst = b(name)    #注意此处使用的是class 变量，即类名.变量名

    @classmethod
    def print(cls):
        a.b_inst.print()
```

```
if __name__ == '__main__':
    a_inst = a('name_b')    #此处实例化，仅仅只是为了给a的类变量赋值
    a.print()               #调用class a的classmethod 方法里有使用到class a的类变量。
```