

【测试】

# 测试 @property 装饰器

```
class employee:
    def __init__(self):
        pass
    @property
    def __work(self):
        print("好好工作")
```

```
me = employee()
# me._employee__work()
me._employee__work
```

【测试】对输入数据进行过滤

# 测试 @property 装饰器

```
class employee:
    def __init__(self, salary):
        self.__salary = salary

    # 获取方法变属性的值的方法（输出）
    @property
    def salary(self):
        return self.__salary

    # 获得值时的操作（输入）
    @salary.setter
    def salary(self, salary):
        if salary < 0:
            print("error")
        else:
            self.__salary = salary
```

```
me = employee(3000)
print(me.salary)
me.salary = -10
print(me.salary)
```