**Daily Report**

23rd February, 2018

1. **访问限制**

如果要让内部属性不被外部访问，可以把属性的名称前加上两个下划线\_\_，在Python中，实例的变量名如果以\_\_开头，就变成了一个私有变量（private），只有内部可以访问，外部不能访问

可以通过给类施加方法，能获得私有变量；或者通过设置方法，改变内部值

也可以通过单下划线+类名+变量名的方法进行私有变量访问（**不建议**）

1. **特殊变量**

以双下划线开头，并且以双下划线结尾的，是特殊变量，特殊变量是可以直接访问的，不是private变量

1. **单一下划线开头变量**

以一个下划线开头的实例变量名，比如\_name，这样的实例变量外部是可以访问的，但是，按照约定俗成的规定，当你看到这样的变量时，意思就是，“虽然我可以被访问，但是，请把我视为私有变量，不要随意访问”

1. **判断某一对象是否为函数**

可以通过使用types模块中的常量，一般有types.FunctionType,types.BuiltinFunctionType,types.LambdaType,types.GeneratorType

对于继承关系的判断，某一对象是否为某种类型，可以通过isinstance（），对于能用type（）判断的类型，也能用isinstance判断

也可以使用isinstance判断某变量是否为某几种类型中的其中一种，如：

>>> isinstance([1, 2, 3], (list, tuple))

True

>>> isinstance((1, 2, 3), (list, tuple))

True

1. 类似于\_\_xxx\_\_的属性和方法，在python中都是有特殊用途的， 如使用len（）方法返回长度，实际上是调用了该对象的\_\_len\_\_（）方法
2. getattr（）用于获得测试对象属性；也可以添加一个default参数，若无某对象属性，则返回该default参数

setattr（）用于设置测试对象属性

hasattr（）用于确定是否测试对象有某属性

1. **限制class实例添加属性**

可以通过定义\_\_slots\_\_变量，如：

class Student(object):

\_\_slots\_\_ = ('name', 'age') # 用tuple定义允许绑定的属性名称

使用\_\_slots\_\_要注意，\_\_slots\_\_定义的属性仅对当前类实例起作用，对继承的子类是不起作用的

除非在子类中也定义\_\_slots\_\_，这样，子类实例允许定义的属性就是自身的\_\_slots\_\_加上父类的\_\_slots\_\_

1. **@property**

在一个方法上添加@property就可以将该方法变为属性