**[python 学习（二）--关于类](https://www.cnblogs.com/zhenfei/p/6241523.html)**

1、没有权限控制，在类方法或变量前加 "\_\_" 两下划线，则变为“私有”变量（实际通过\_<类名>\_\_<变量或方法名> 可以访问）

2、类的若干自动触发的内置方法：

* **\_\_new\_\_(cls[, ...])**     //该方法返回一个本类的实例，传入的参数会原封不动传递给 \_\_init\_\_() 方法. 该方法一般在继承一个不可修改的类时使用，例如 str

class CapStr(str):

　　def \_\_new\_\_(cls, string):

　　　　string = string.upper()

　　　　return str.\_\_new\_\_(cls, string)

a = CapStr("I love you");

* **\_\_init\_\_(self[, ...])**    //该方法在实例创建后最早被调用，用于初始化对象

class Rectangle:

　　def \_\_init\_\_(self, x, y):

　　　　self.x = x

　　　　self.y = y

　　def getPeri(self):

　　　　return(self.x + self.y)\*2

aa=Rectangle(3, 4);

aa.getPeri()

* **\_\_del\_\_(self)**   //该方法在类实例被释放，最后垃圾回收的时候才被触发

c1=C()

c2=c1

c3=c2

del c3   //不会被触发

del c2  //不会被触发

del c1  //全部引用消除，触发 \_\_del\_\_

* \_\_add\_\_(self,other) //两个类相加时触发
* \_\_sub\_\_(self,other) //两个类相减时触发
* \_\_mul\_\_(self,other) //乘法触发
* \_\_truediv\_\_(self,other) //真除法