2016年7月5日星期二

1：类之间的关系：**组合关系：一个类可以将别的类作为它的组件构建起来，比如车子和轮子的关系**

Eg：

class Person(object):

def \_\_init\_\_(self,name):

self.name = name

#变量，组合，有一个pet属性，pet是一个类

self.pet = None

2：字典keys()方法，返回字典的一个list

Eg：

indexKeys = PHREASE.keys()

3：random的shuffle()方法，就是洗牌，作用是将列表随机打乱

Eg：

random.shuffle(indexKeys)

4：注意：这种赋值方法，即可以使用一个等号赋多个值

question,answer = indexKey,val

5：关于class Name(object)：

小写的object代表一个类，创建新类的时候继承下来，为了最初的版本的一些类可以使用。因此就要求在定义一个类的时候总是要加上object。

6：继承父类的构造方法的格式：

Eg：

class Employee(Person):

def \_\_init\_\_(self,name,salary):

#使用父类的构造方法，注意格式，注意super中是子类和self

**super(Employee,self).\_\_init\_\_(name)**

self.salary = salary

7：上文中的self.pet = None作用是：确保self.pet属性被设置为None。

8：python支持多继承，但是我们应该尽量避免使用多继承。——大部分使用多继承的场合都可以使用合成取代，多继承应该不惜一切的避免。但是是支持的。

9：好的注释是短小精悍，主要是写清楚为什么要这样做！

2016年7月6日星期三

10：方法或者类中的pass，关键词是在python中创建一个空代码块，没有定义细节。

11：Python中覆盖和java中类似。也是三种，继承，同名覆盖，super()部分继承。

12：python支持多继承：但是应该避免使用多继承。

Class SuperFun(Child,BadStuff):

Pass

13：最常用的super()方法的用法是在基类中\_\_init\_\_()函数中使用。这也是唯一可以进行这种操作的地方。完成了对父类的初始化。

14：python中类和函数命名风格与java相同。