# 中科同向Python笔试题

## Python 中 单引号 , 双引号 , 三引号 的区别 ?

单引号，双引号混用，不用转义字符

三引号多行书写，原样输出

## 在Python 中 , list,tuple,dict,set 有什么区别,主要应用在什么样的场景?

**定义:**

**list:**链表,有序的项目, 通过索引进行查找,使用方括号”[]”;

**tuple:**元组,元组将多样的对象集合到一起,不能修改,通过索引进行查找, 使用括号”()”;

**dict:**字典,字典是一组键(key)和值(value)的组合,通过键(key)进行查找,没有顺序, 使用大括号”{}”;

**set:**集合,无序,元素只出现一次, 自动去重,使用”set([])”

**应用场景:**

**list,** 简单的数据集合,可以使用索引;

**tuple**, 把一些数据当做一个整体去使用,不能修改;

**dict,**使用键值和值进行关联的数据;

**set,**数据只出现一次,只关心数据是否出现, 不关心其位置;

## 请写出打印结果

x = [0 ,1 ]

i = 0

i , x[i] = 1 , 2

print(x)

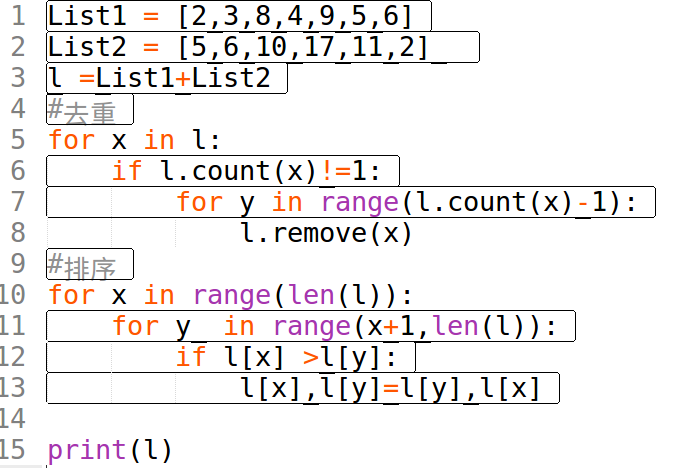
打印结果x:

[0,2]

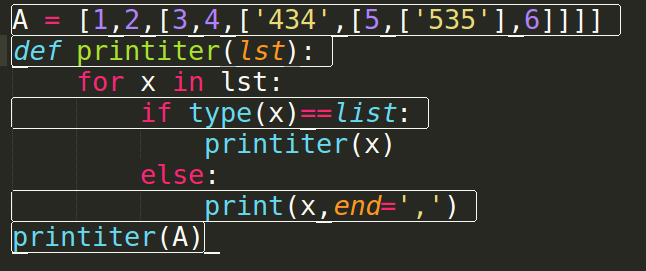
## 请用自己的算法,按升序合并如下两个List , 并去除重复的元素:

List1 = [2,3,8,4,9,5,6]

List2 = [5,6,10,17,11,2]



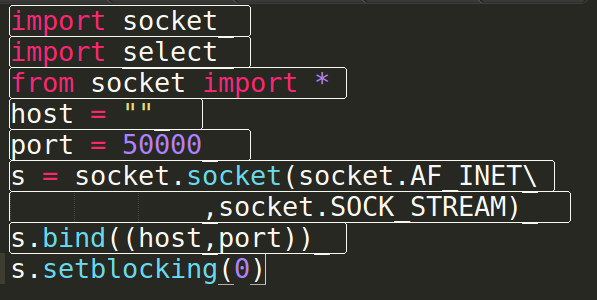
## 有一个多层嵌套的列表 A = [1,2,[3,4,[‘434’,[…]]]],请写一段代码遍历A中的美个元素并打印出来.



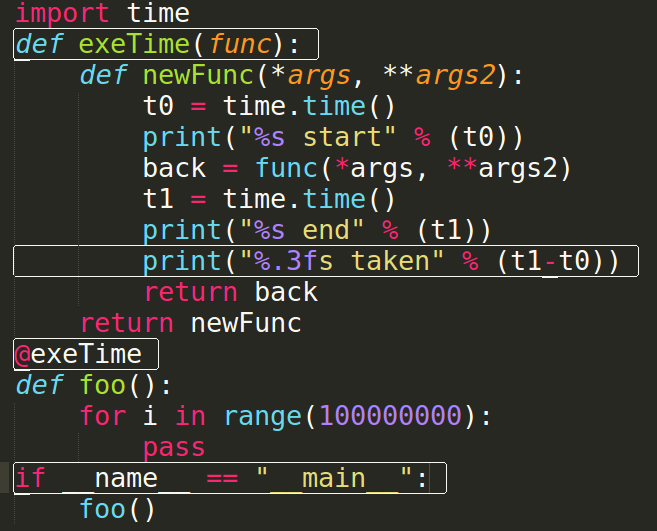
## 关系型数据库中,表与表之间有左连接,内连接,外连接. 分别解释下他们的含义及区别?

一、内连接   
内连接查询操作列出与连接条件匹配的数据行，它使用比较运算符比较被连接列的列值。   
内连接分三种：   
1、等值连接：在连接条件中使用等于号(=)运算符比较被连接列的列值，其查询结果中列出被连接表中的所有列，包括其中的重复列。   
2、不等连接：在连接条件使用除等于运算符以外的其它比较运算符比较被连接的列的列值。这些运算符包括>、>=、<=、<、!>、!<、!=和<>。   
3、自然连接：在连接条件中使用等于(=)运算符比较被连接列的列值，但它使用选择列表指出查询结果集合中所包括的列，并删除连接表中的重复列。   
二、外连接   
返回到查询结果集合中的不仅包含符合连接条件的行，而且还包括左表(左外连接时)、右表(右外连接时)或两个边接表(全外连接)中的所有数据行。   
左连接和右连接的区别：   
左连接以左表为基准进行查询,左表数据会全部显示出来,右表如果和左表匹配的数据则显示相应字段的数据,如果不匹配,则显示为NULL;右连接刚好相反

## 请用Python 实现 socket 非阻塞通讯



## 8. 使用一个装饰器,查看函数的运行时间,有什么作用



## 9. Django 的mtv 分别代表什么

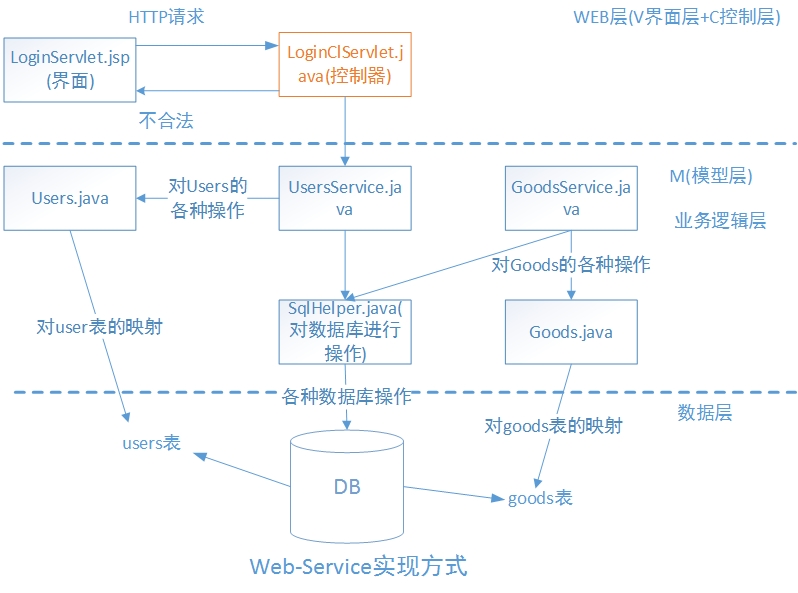
**Models：**一个抽象层，用来构建和操作你的web应用中的数据，模型是你的数据的唯一的、权威的信息源。它包含你所储存数据的必要字段和行为。通常，每个模型对应数据库中唯一的一张表。

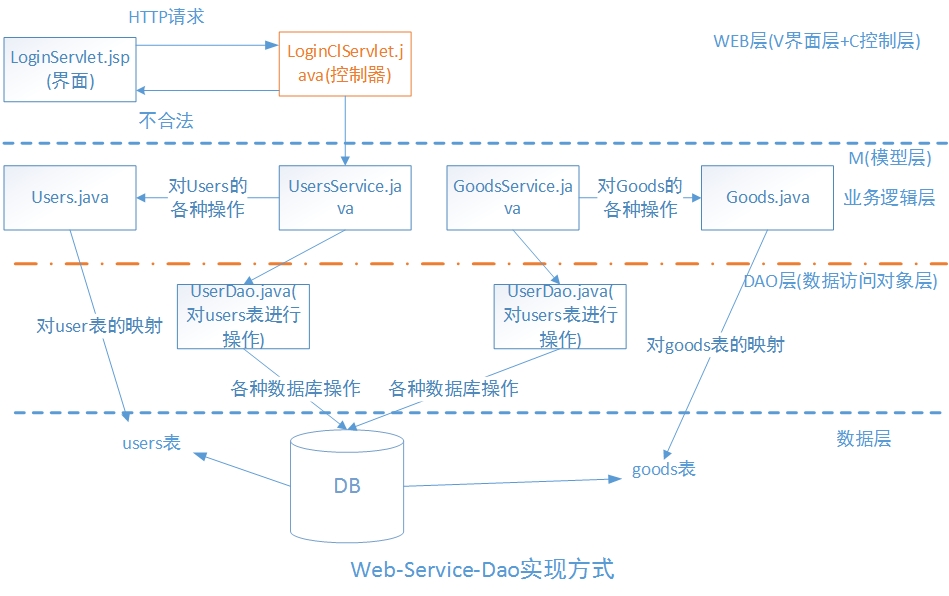
**模板(templates)**：模板层提供了设计友好的语法来展示信息给用户。使用模板方法可以动态地生成HTML。模板包含所需HTML 输出的静态部分，以及一些特殊的语法，描述如何将动态内容插入

**视图（views）**：用于封装负责处理用户请求及返回响应的逻辑。视图可以看作是前端与数据库的中间人，他会将前端想要的数据从数据库中读出来给前端。他也会将用户要想保存的数据写到数据库。

## 10. 在Django 中定义两个模型,书籍 和作者. 书籍有3个字段,分别为title,author,publish\_time ; 作者有3个字段 分别为: name , age , address, 其中书籍和作者是多对多的关系,查看一本书籍的所有作者和某个作者的全部书籍

## 11. 简述你使用的 web 框架 中是如何实现 mvc 模式的





## 12. 简述你熟悉的ORM 框架的使用机制和实现原理

**Hibernate使用机制和实现原理:**

使用机制:

主要依据反射机制

原理:

1.通过Configuration().configure();读取并解析hibernate.cfg.xml配置文件

2.由hibernate.cfg.xml中的< mapping resource=”com/xx/User.hbm.xml”/>读取并解析映射信息

3.通过config.buildSessionFactory(); //创建SessionFactory

4.sessionFactory.openSession(); //打开Sesssion

5.session.beginTransaction(); //创建事务Transation

6.persistent operate 持久化操作

7.session.getTransaction().commit(); //提交事务

8.关闭Session

9.关闭SesstionFactory