

一面

面试官全程没有问知识点，说哪些东西太死了，来面试的基本都是背的，他就从实际项目角度出发，问的都是一些实际项目开发中的场景，让提出自己的设计思路以及如何去实现。

（每提出一种设计思路后，他都会考虑下你当前思路在一些特殊场景是不是也没问题，用户体验是否良好，有问题如何修改）

以下是面试内容：

1.介绍一个自己做过的项目，项目描述，自己做了哪些东西，然后就开始提问，内容如下：

1) 数据库表是怎么设计的，他提了一个功能模块，让说一下有关那个模块所有表结构的设计（如果你设计的表结构不合理比如没合理使用范式，数据冗余等现象，都会当场给指出来，该怎么修改）

2) 用户登录权限验证怎么做的；如何做到方法级权限；什么时候验证如何验证；为什么有了用户表和权限表还要加一个角色表，优点体现在哪？

3) 另外问了我自己之前做的项目中一个需求上的问题

2.如果要做一个城市基础信息维护，国家、省、市、区的信息如何去维护，表结构如何设计并且要建立树形对应关系；如何查询所有省级别的信息；如果将某个市由 A 省修改到 B 省，该市下面的所有区级信息如何弄？

要注意不能以地区编号作为主键，因为主键不可修改，选一个 id 做主键，每条记录存个等级和父节点编号就好了

(id, class, name,, level, parentclass)

上面的是我当时说的一种思路，用 level 表示国家 省 市 区的对应级别，用 parentclass 存储父节点编号

3.多人同时在线编辑一个文档，如何去做并保持良好用户体验。

这个我一开始说的悲观锁互斥的思想，真是傻了，面试官就说了句这种设计情况下如果当前用户去做其它任务了，忘了提交，那其他用户不是都一直等着？

然后就明白改成多人实时编辑的思路了，另外提问了如何发现冲突，如何解决冲突，这一部分我是用 git 协同开发的角度回答的。

二面

1.自我介绍

2.问我本科和研究生方向，主要学了哪些课程

3.算法题

给一个类似于 A(B(,),C(d,e))的字符串，建立一颗二叉树，括号里面的表示子树，比如 A 有 BC 两颗子树，C 有 de 两颗子树。（简单递归问题）

```

import java.util.Scanner;
class Node{
    char ch;
    Node left;
    Node right;
    public Node(char ch){
        this.ch = ch;
    }
}

public class Solution {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String str = sc.next();
        Node root = dfs(str);
    }

    //A(B(,),C(,))
    public static Node dfs(String str){
        if (str.length() <= 4){
            return new Node(str.charAt(0));
        } else {
            Node node = new Node(str.charAt(0));
            String temp = str.substring(2, str.length() - 1);
            String[] childs = temp.split(regex: ",");
            node.left = dfs(childs[0]);
            node.right = dfs(childs[1]);
            return node;
        }
    }
}

```

牛客@Bjutliuya

4.TCP 为什么需要三次握手？

这部分我按书上说的，面试官说不不对，然后他给给说了一种我觉得需要调研一下

5.TCP 连接信息中客户端和服务端需要交互哪些东西？

6.说了个场景，让设计数据库表

7.反问

三面

三面应该是 cto 面的吧，流程很快，面试官上来自己先自我介绍起来了，说简历都已经看过不用我再介绍了

1.学习生活上的事聊一聊

2.聊了聊本科期间参加的编程竞赛和数模竞赛

3.手撕一个代码（比较简单）