# 一、问题预测

1. 让简单介绍一下 自己

你好，我叫徐涵，来自江苏，毕业于南京晓庄学院计算机科学与技术专业，是18届的毕业生。目前有一年java开发和一年测试经验。最近的一份工作是在南京未来软件，主要负责的工作是未来投标云助手的测试工作。从项目立项开始即加入项目组，全程参与到整个系统的测试，直至上线。

1. 说下自己会的内容

熟悉软件测试基础理论和测试流程，测试方法等。有web测试、接口测试、性能测试经验。

熟悉简单的sql语句，linux基本命令

了解java基础，熟悉开发思想，更容易定位bug

了解python基本语法，结合selenium协助开发简单的爬虫demo

1. 看了哪些书籍

《图解HTTP》、《测试基础》、《软件测试第二版》

1. 了解过哪些技术博客/论坛

51testing论坛

CSDN一些博客（面试经验：面试中会问具体哪些博客），用fiddler做抓包分析详解、python+selenium

公众号（搜狗测试、软件测试资源分享），软件测试资源站（爬虫搬运互联网资源，分享给自学爱好者。主攻软件测试自动化及开发方向）

1. 是否了解软件测试需要哪些知识

软件测试基础知识，流程，测试用例方法

数据库相关知识，抓包分析，接口测试、测试工具、性能测试等

1. 之前面试过，觉得自己需要补充哪些？做了哪些行动

很多公司对性能测试和自动化测试工具有要求，需要储备这些知识

1. 为什么做测试，觉得自己做测试有哪些优势

* 我觉得我个人性格比较适合做测试。我比较细心耐心，考虑事情比较全面，这样对于我在设计用例时很有帮助。
* 而且我能够很好的与人协调沟通，当我们测试和开发发生沟通上的矛盾时我也能很好的解决。
* 我平时喜欢刷微博、知乎看热门评论，喜欢考究大众心理，这有助于我站在用户角度设计测试点

1. 知道哪些bug系统

禅道，jrra

1. 测试的基本要素是

版本号、功能模块、优先级别、前置条件、预期结果、实际结果等

# 二、介绍一下公司项目

# 三、技能方面

## 1、数据库方面知识

1. 关系型数据库：把复杂的数据结构归结为简单的二元关系（即二维表格形式），通过SQL结构化查询语句存储数据。

典型产品：

Mysql：互联网领域、大中小型网站、游戏公司、电商平台等等。体积小、速度快、成本低、开放源代码

Oracle：传统大企业、大公司、政府、金融、证券等。安全性、成本高

1. 非关系型数据库：菲关系型数据库也被称为NoSQL数据库，NoSQL的本意是“Not Only SQL”。NoSql为了高性能、高并发而生
2. 其他分类

* 键值（Key-Value）存储数据库：主要是使用一个哈希表，这个表中有一个特定的键和一个指针指向特定的数据，简单、易部署、高并发

典型：Redis、Memcached

* 列存储（Column-oriented）数据库：应对分布式存储的海量数据。如果我们有person类，我们通常会一起查询他们的姓名和年龄，而不是薪资。这种情况下，姓名和年龄就会放在一个列族中，而薪资则放在另外一个列族中。

典型：Hbase

* 面向文档数据库，数据存储的最小单位是文档

典型：Mongodb、Hive

## 2、linux操作

linux搭建测试环境、比如web系统服务搭建

## 3、缺陷方面

描述一个你印象深刻的bug

## 4、用例部分

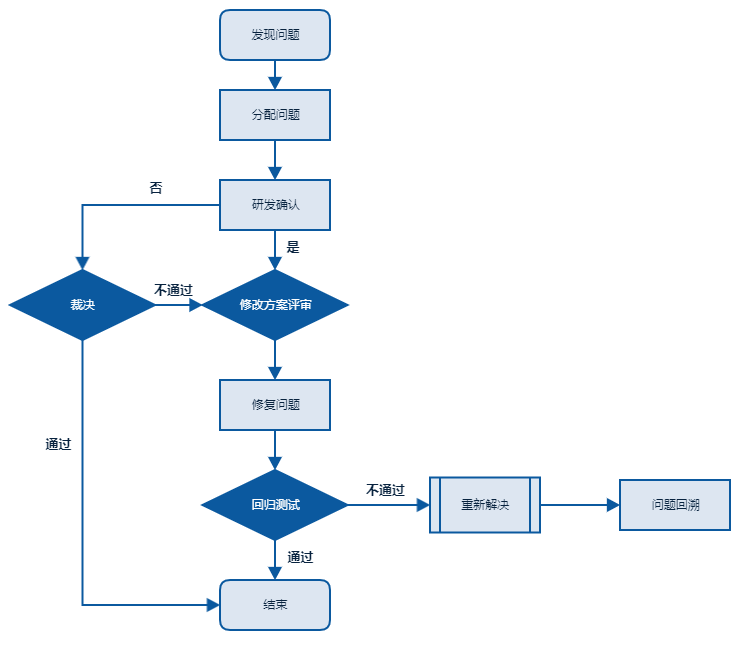
现场让你设计个用例，比如水杯、凳子怎么测试？

首先说明的是，遇到这样的测试题目，首先应该反问面试官，需求是什么样的，比如是测什么样的杯子。

因为设计测试用例的规则应该是根据需求分析文档设计用例，客户需求什么，就测试什么。

从以下几个方面进行分析：功能测试、性能测试、界面测试、安全性测试、兼容性测试、可用性测试、可靠性测试、本地化/国际化测试。

## 5、软件测试流程



2.1. 发现问题

测试发现问题后，将问题提交到 JIRA 或禅道上

2.2. 分配问题

由 PO 根据当前人员任务情况及问题优先级将问题分配到开发人员上，现网阻塞性的问题优先级最高，需要立即安排。

2.3. 研发确认

研发人员收到问题后，需要先分析问题，确认该 ”问题“ 是否真的是问题。

2.4. 裁决

如果研发认为不是问题，进入裁决环节，此环节需要 PO，产品共同决议这个 ”问题“ 是不是问题。如果确认是，开发人员必须去修改，如果还有异议，可以找研发经理。

如果裁决不是问题，那么直接将问题结束。

2.5. 修改方案评审

如果确认是问题或者裁决不通过进入此环节，在此环节中评审修改问题的方案，重点评审其对现有代码的冲击，是否会将通用性的代码改成特化版，无法再适配开发文档中的其它场景。

这里需要秉持一个原则：如果是架构的问题不适配，那就去改架构，千万不要在通用代码上胡乱打 Patch。如果认为是架构的问题，需要提出详细理由。

2.6. 修复问题

开发人员按照评审后的方案进行修改代码。

2.7. 回归测试

测试人员对该修复的问题进行测试，同时要关注修复的代码对其它功能的影响。

2.8. 重新解决

如果回归不通过，将问题打回，按发现问题之后的流程重新走一遍。

2.9. 问题回溯

由 PO 发起对此次问题回归不通过的讨论会，重点是如何改变，下次不会再犯同样的错误。

1. 公司严格规范测试流程和测试文档，首先是参与需求评审，编写测试计划、测试方案、测试用例，进行测试方案及用例的测试组内部评审，外部评审。
2. 提取部分一级用例提交研发自测，研发自测通过后开始执行一轮系统测试。
3. 测试过程中发现并提交、跟踪问题。
4. 问题修复后进行回归测试。
5. 一轮测试完成后对修复包进行冒烟测试，测试通过则进行二轮测试。
6. 二轮测试完成后会进行需求交叉测试。
7. 完成测试编写系统测试报告提交验收测试。验收测试通过输出验收测试报告。

## 6、网络相关

网络协议，如TCP/UDP的区别

1）、TCP面向连接（如打电话要先拨号建立连接）；

UDP是无连接的，即发送数据之前不需要建立连接

2）、TCP提供可靠的服务。也就是说，通过TCP连接传送的数据，无差错，不丢失，不重复，且按序到达；UDP尽最大努力交付，即不保证可靠交付

3）、TCP面向字节流，实际上是TCP把数据看成一连串无结构的字节流；UDP是面向报文的；UDP没有拥塞控制，因此网络出现拥塞不会使源主机的发送速率降低（对实时应用很有用，如IP电话，实时视频会议等）

4）、每一条TCP连接只能是点到点的；UDP支持一对一，一对多，多对一和多对多的交互通信

5）、TCP首部开销20字节；UDP的首部开销小，只有8字节

6）、TCP的逻辑通信信道是全双工的可靠信道，UDP则是不可靠信道

三次握手与四次挥手：

三次握手通俗版：

* 第一次握手：客户端要和服务端进行通信，首先要告知服务端一声，遂发出一个SYN=1的连接请求信号,”服务端哥哥，我想给你说说话”。
* 第二次握手：当服务端接收到客户端的连接请求，此时要给客户端一个确认信息，”我知道了（ACK）,我这边已经准备好了，你现在能连吗（SYN）”。
* 第三次握手：当客户端收到了服务端的确认连接信息后，要礼貌的告知一下服务端，“好的，咱们开始联通吧（ACK）”。
* 到此整个建立连接的过程已经结束，接下来就是双方你一句我一句甚至同时交流传递信息的过程了。

四次挥手断开连接通俗版：

* 第一次挥手：双方交流的差不多了，此时客户端也已经结尾了，接下来要断开通信连接，所以告诉服务端“我说完了（FIN）”，此时自身形成等待结束连接的状态。
* 第二次挥手：服务端知道客户端已经没话说了，服务端此时还有两句话要给客户端说“我知道你说完了（ACK），我再说两句&\*…%￥”…
* 第三次挥手：此时客户端洗耳恭听继续处于等待结束的状态，服务器端也说完了，自身此时处于等待关闭连接的状态，并对告诉客户端，“我说完了，咱们断了吧（FIN）”。
* 第四次挥手：客户端收知道服务端也说完了，也要告诉服务端一声（ACK），因为连接和断开要双方都按下关闭操作才能断开，客户端同时又为自己定义一个定时器，因为不知道刚才说的这句话能不能准确到达服务端（网络不稳定或者其他因素引起的网络原因）。
* 所以默认时间定为两个通信的最大时间之和，超出这个时间就默认服务器端已经接收到了自己的确认信息，此时客户端就关闭自身连接，服务器端一旦接收到客户端发来的确定通知就立刻关闭服务器端的连接。

到此为止双方整个通信过程就此终结。

这里要声明一下：断开链接不一定就是客户端，谁都可以先发起断开指令，另外客户端和服务端是没有固定标准的，谁先发起请求谁就是客户端。

* 三次握手阐述：

在第一次消息发送中，A随机选取一个序列号作为自己的初始序号发送给B；

第二次消息B使用ack对A的数据包进行确认，因为已经收到了序列号为x的数据包，准备接收序列号为x+1的包，所以ack=x+1，同时B告诉A自己的初始序列号，就是seq=y；

第三条消息A告诉B收到了B的确认消息并准备建立连接，A自己此条消息的序列号是x+1，所以seq=x+1，而ack=y+1是表示A正准备接收B序列号为y+1的数据包。

* 四次挥手阐述：

由于TCP连接时全双工的，因此，每个方向都必须要单独进行关闭，这一原则是当一方完成数据发送任务后，发送一个FIN来终止这一方向的连接，

收到一个FIN只是意味着这一方向上没有数据流动了，即不会再收到数据了，但是在这个TCP连接上仍然能够发送数据，直到这一方向也发送了FIN。

首先进行关闭的一方将执行主动关闭，而另一方则执行被动关闭，上图描述的即是如此。

（1）第一次挥手：Client发送一个FIN，用来关闭Client到Server的数据传送，Client进入FIN\_WAIT\_1状态。

（2）第二次挥手：Server收到FIN后，发送一个ACK给Client，确认序号为收到序号+1（与SYN相同，一个FIN占用一个序号），Server进入CLOSE\_WAIT状态。

（3）第三次挥手：Server发送一个FIN，用来关闭Server到Client的数据传送，Server进入LAST\_ACK状态。

（4）第四次挥手：Client收到FIN后，Client进入TIME\_WAIT状态，接着发送一个ACK给Server，确认序号为收到序号+1，Server进入CLOSED状态，完成四次挥手。

## 7、测试工具

自动化测试工具（如QTP）

性能测试工具（如Loadrunner）

测试管理类（如jira）

安全测试工具

渗透测试工具

## 8、其他概念问题

Beta测试和Alpha测试有什么区别

1. Alpha测试

Alpha测试是由用户在开发环境下进行的测试，也可以是开发机构内部的用户在模拟实际操作环境下进行的测试。开发者坐在用户旁边，

1. Beta测试
2. 验收测试