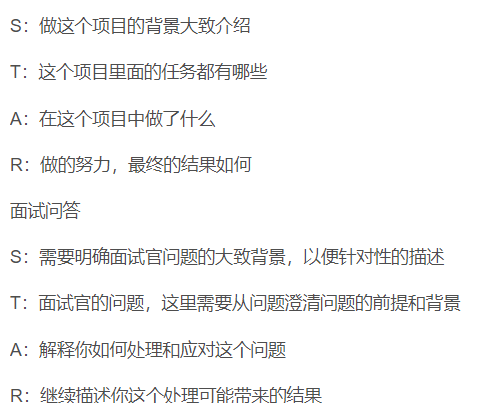
1/27/22🡪1/31/2022

5/18/22





问：给你一个网站，你如何测试?

答.首先，查找需求说明、网站设计等相关文档，分析测试需求。

制定测试计划，确定测试范围和测试策略，一般包括以下几个部分：功能性测试;界面测试;性能测试;数据库测试;安全性测试;兼容性测试

设计测试用例：

功能性测试可以包括，但不限于以下几个方面：

链接测试。链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回。

提交功能的测试。

多媒体元素是否可以正确加载和显示。

多语言支持是否能够正确显示选择的语言等。

界面测试可以包括但不限于一下几个方面：

页面是否风格统一，美观

页面布局是否合理，重点内容和热点内容是否突出

控件是否正常使用

.对于必须但未安装的控件，是否提供自动下载并安装的功能

文字检查

性能测试一般从以下两个方面考虑：

压力测试;负载测试;强度测试

数据库测试要具体决定是否需要开展。数据库一般需要考虑连结性，对数据的存取操作，数据内容的验证等方面。

安全性测试：

基本的登录功能的检查

是否存在溢出错误，导致系统崩溃或者权限泄露

相关开发语言的常见安全性问题检查，例如SQL注入等

如果需要高级的安全性测试，确定获得专业安全公司的帮助，外包测试，或者获取支持

兼容性测试，根据需求说明的内容，确定支持的平台组合：

浏览器的兼容性;

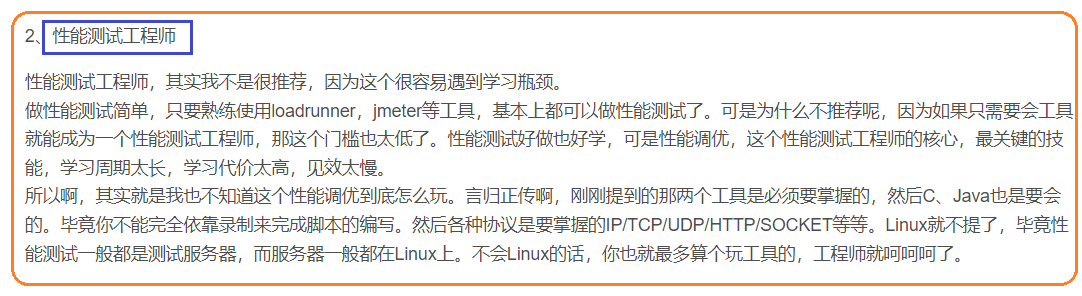
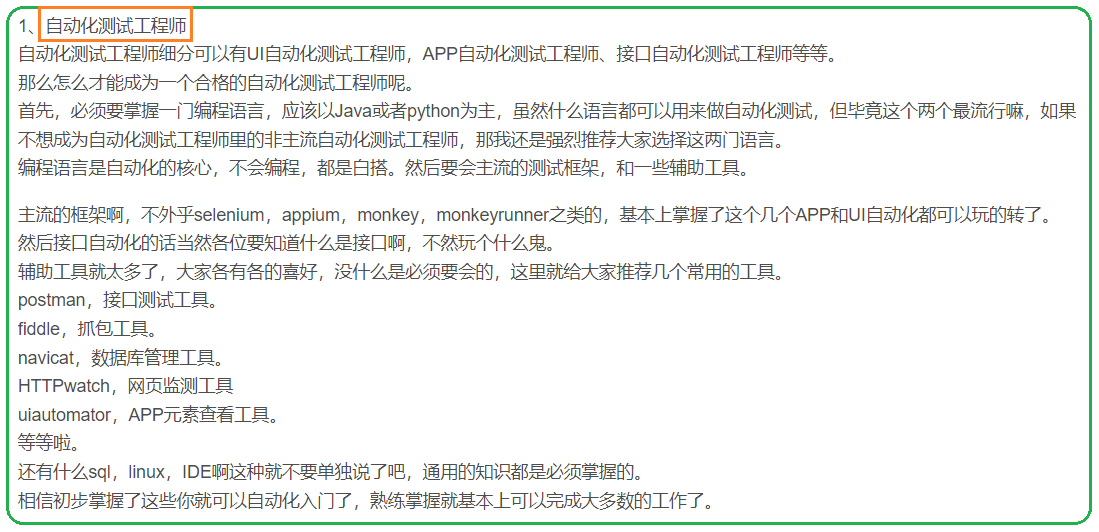
操作系统的兼容性;

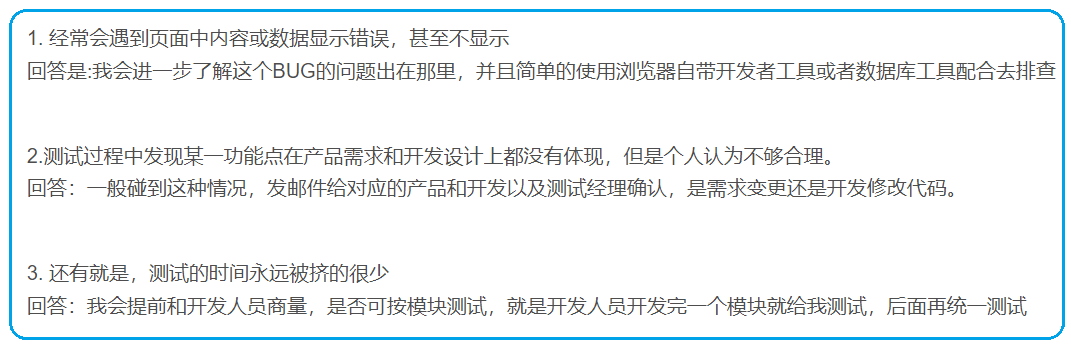
软件平台的兼容性;

数据库的兼容性

开展测试，并记录缺陷。合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系(例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容)。

定期评审，对测试进行评估和总结，调整测试的内容。【推荐了解：软件测试课程】



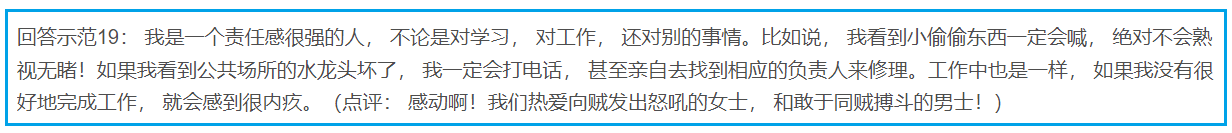


# 软件测试工程师面试遇到的问题

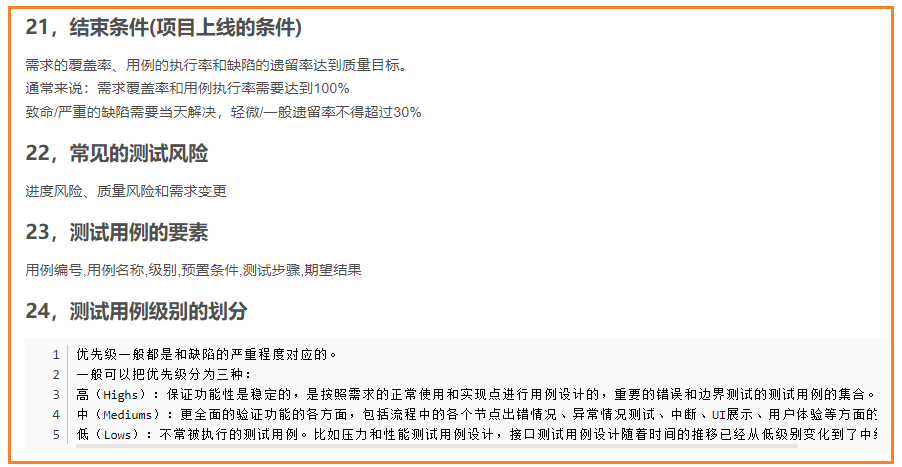
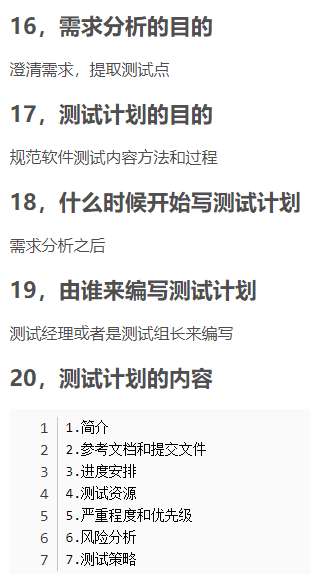
# <https://blog.csdn.net/weixin_42421116/article/details/106670709?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_title~default-0.pc_relevant_paycolumn_v3&spm=1001.2101.3001.4242.1&utm_relevant_index=3>

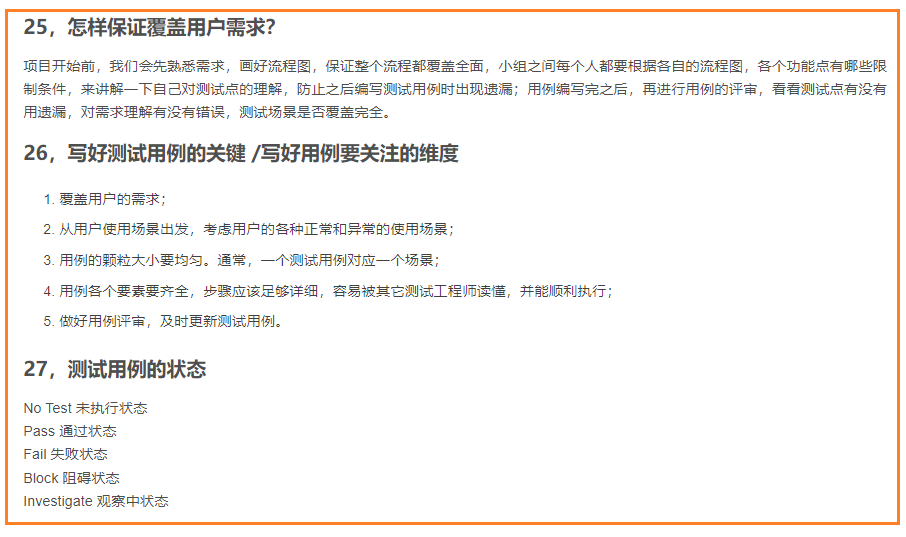
# 

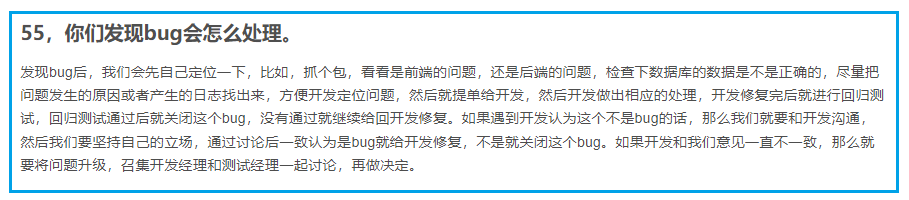
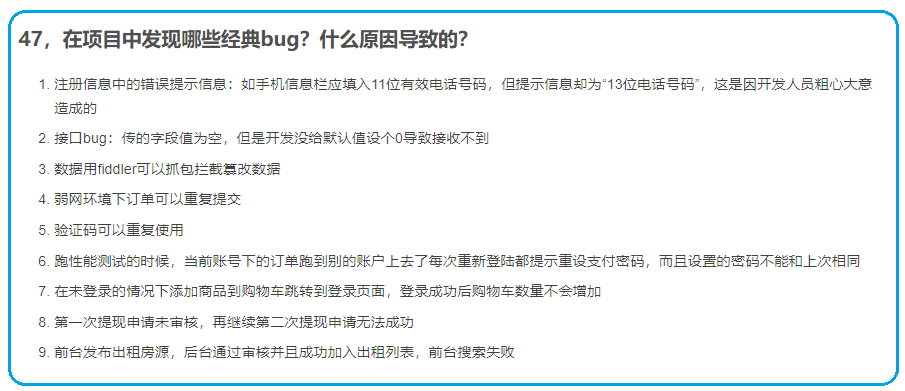
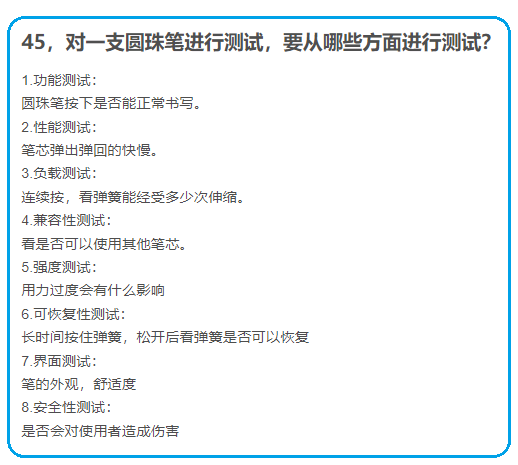
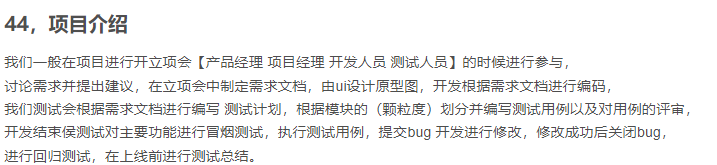
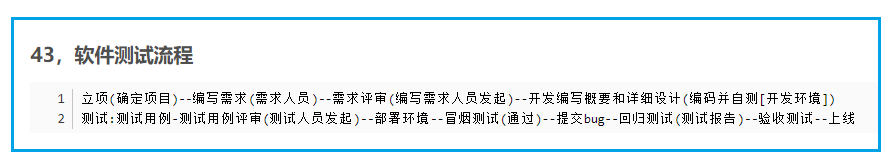
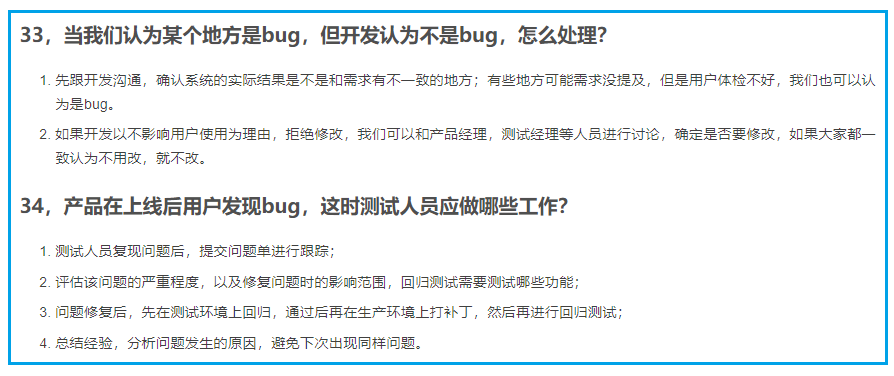
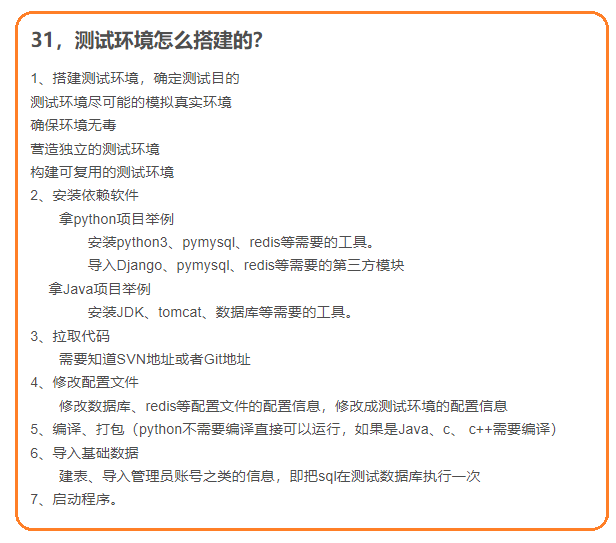






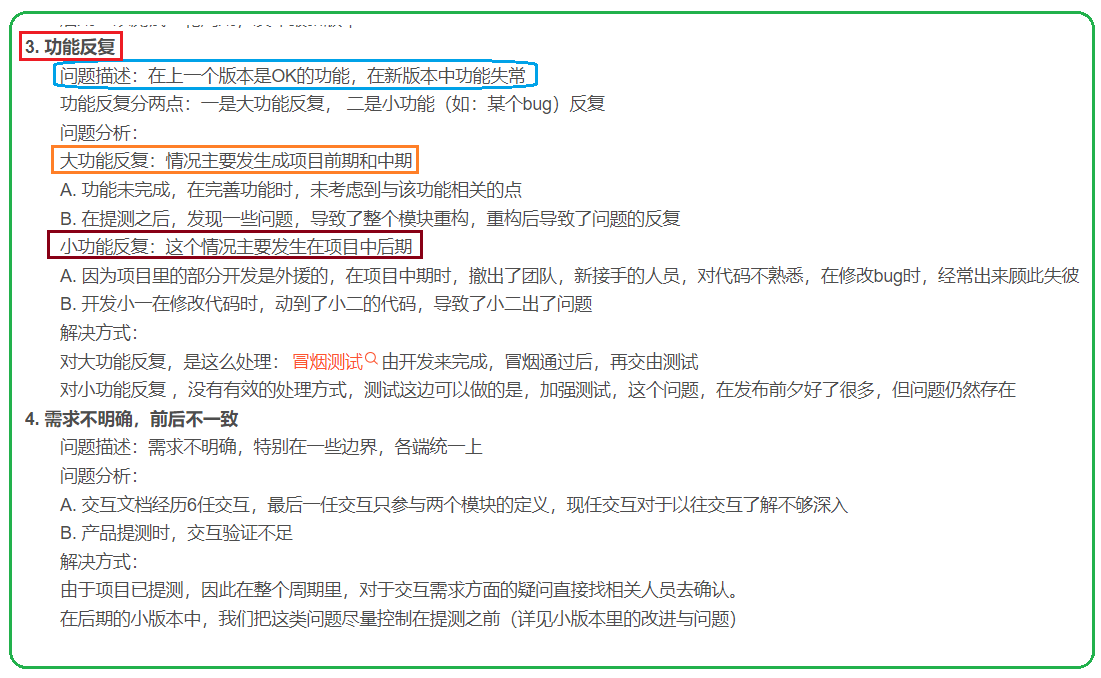
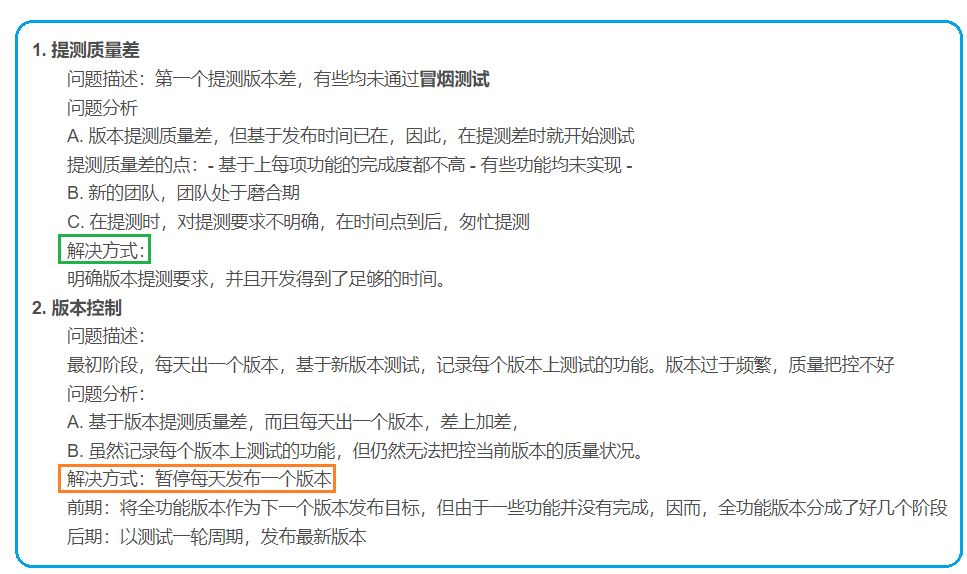


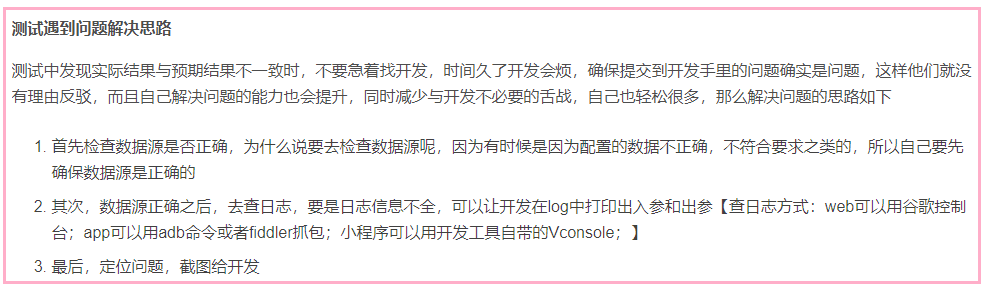
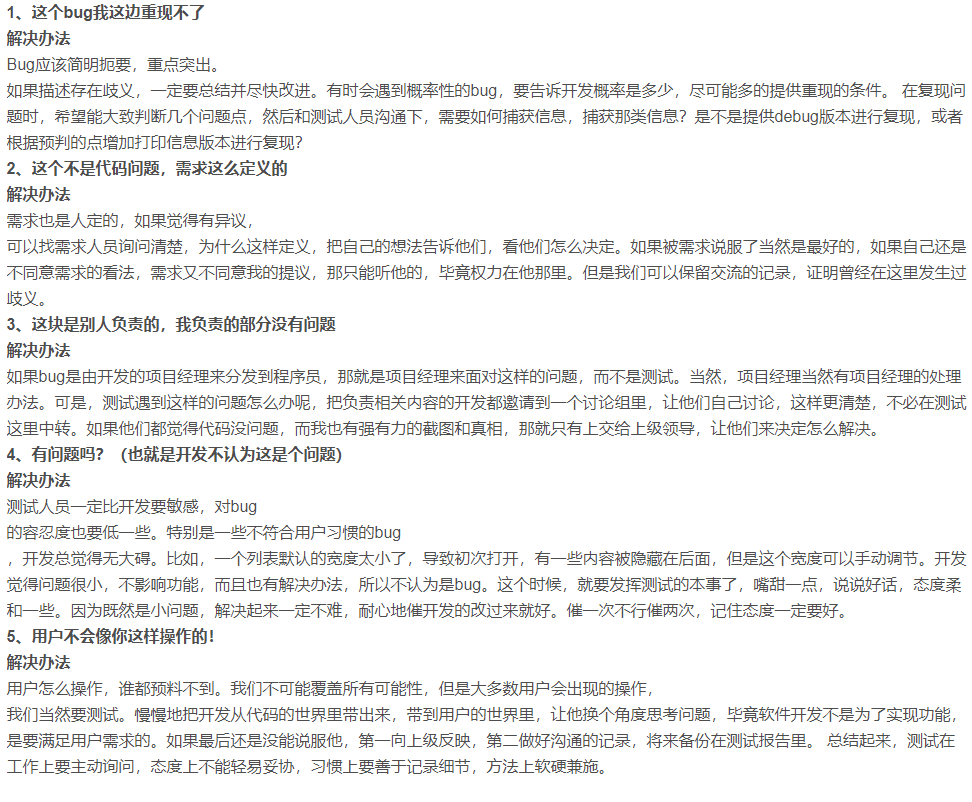
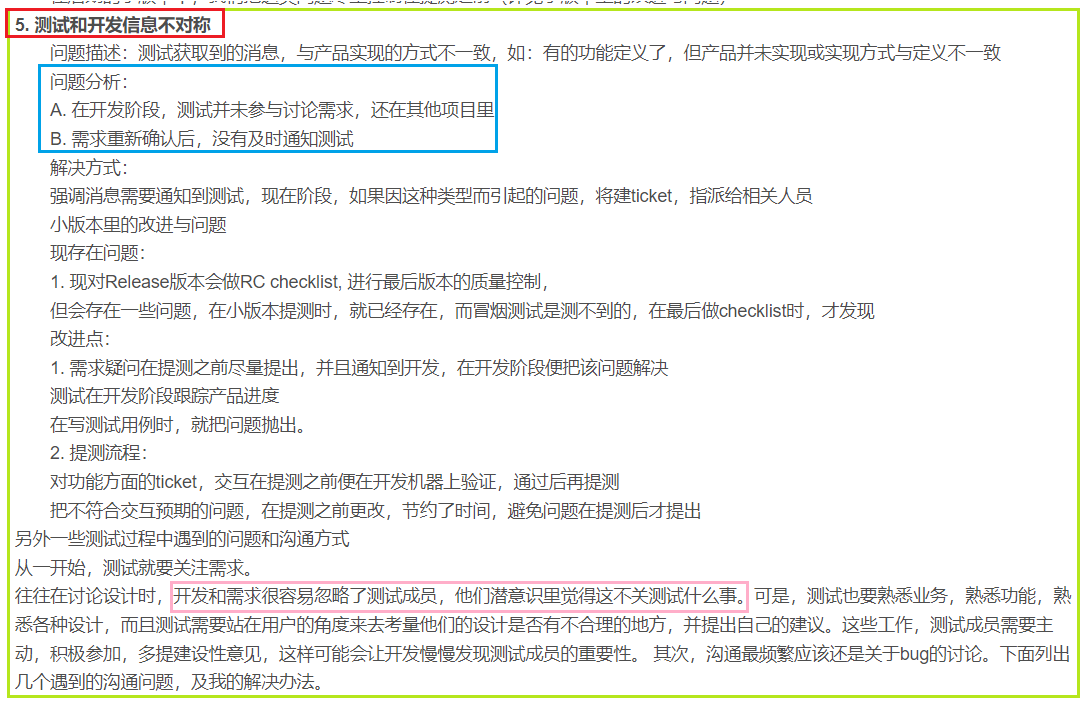


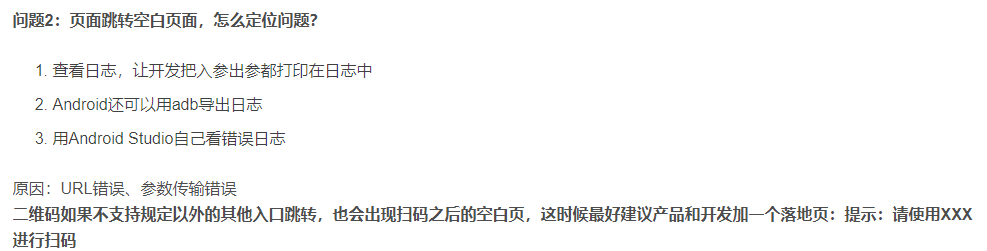


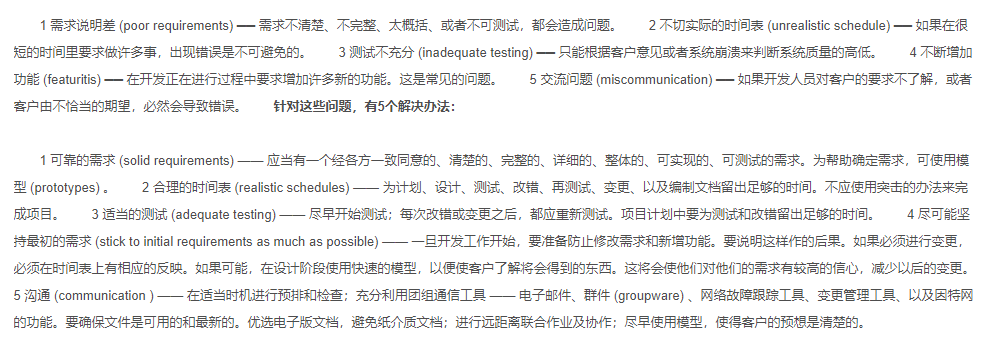
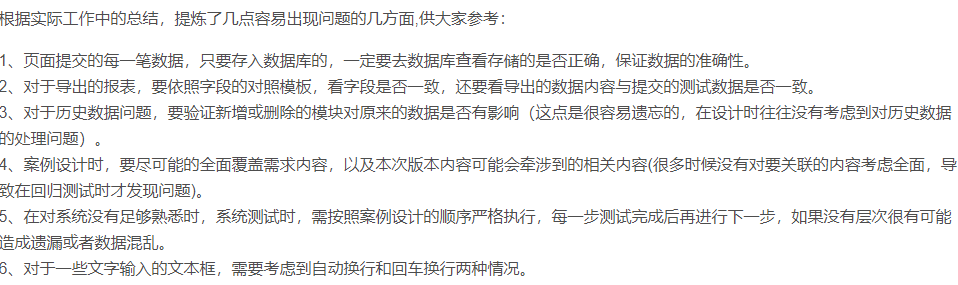
# 软件测试常见的问题

<https://blog.csdn.net/weixin_47295201/article/details/111879647?spm=1001.2101.3001.6650.17&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-17.pc_relevant_paycolumn_v3&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-17.pc_relevant_paycolumn_v3&utm_relevant_index=24>









6/30/21-> 7/12/21

# 2021测试开发面试题大全及答案

<https://blog.csdn.net/weixin_56502375/article/details/119648335?spm=1001.2101.3001.6650.10&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-10.pc_relevant_default&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-10.pc_relevant_default&utm_relevant_index=18>

# 为什么说功能测试是巨坑？

# <https://blog.csdn.net/weixin_56502375/article/details/120832253>

# <https://blog.csdn.net/m0_58026506/article/details/119841377?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_title~default-0.pc_relevant_aa&spm=1001.2101.3001.4242.1&utm_relevant_index=3>

# <https://blog.csdn.net/kele9998/article/details/120183940?spm=1001.2101.3001.6650.2&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-2.pc_relevant_aa&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-2.pc_relevant_aa&utm_relevant_index=5>

# <https://blog.csdn.net/qq_36595013/article/details/88231747?spm=1001.2101.3001.6650.3&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-3.pc_relevant_default&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-3.pc_relevant_default&utm_relevant_index=6>

# python接口测试面试题

# <https://blog.csdn.net/cyjs1988/article/details/76064744?spm=1001.2101.3001.6650.4&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-4.pc_relevant_default&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-4.pc_relevant_default&utm_relevant_index=9>

# python自动化测试面试题大全带答案\_自动化测试面试题及答案大全

# <https://blog.csdn.net/weixin_39957027/article/details/111451651?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_baidulandingword~default-1.pc_relevant_default&spm=1001.2101.3001.4242.2&utm_relevant_index=4>

# python接口自动化测试面试题\_接口自动化测试面试题

# <https://blog.csdn.net/weixin_39884412/article/details/110053490?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_baidulandingword~default-4.pc_relevant_default&spm=1001.2101.3001.4242.3&utm_relevant_index=7>

# <https://blog.csdn.net/PYTHONwuyou/article/details/113351714?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_baidulandingword~default-5.pc_relevant_default&spm=1001.2101.3001.4242.4&utm_relevant_index=8>

# <https://blog.csdn.net/weixin_39884412/article/details/110053490?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_title~default-4.pc_relevant_default&spm=1001.2101.3001.4242.3&utm_relevant_index=7>

1. 软件测试工程师面试

<https://blog.csdn.net/>

* **回答问题，做几个项目。记住细节**。
* Linux
* <https://so.csdn.net/so/search?q=%E6%B5%8B%E8%AF%95%E8%BD%AF%E4%BB%B6%E5%B7%A5%E7%A8%8B%E5%B8%88%E9%9D%A2%E8%AF%95Linux&t=&u=&urw=>
* **搭建测试环境 for UI and API testing.**
* **Performance testing**
* **软件测试面试题**

# 运维工程师面试的经历及面试相关问题

# Linux运维工程师-面试题总结

# Linux系统运维基础测试题

# 软件测试工程师面试题之数据库

# <https://blog.csdn.net/qq_42434318/article/details/114138619?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522162580169216780357220023%2522%252C%2522scm%2522%253A%252220140713.130102334.pc%255Fall.%2522%257D&request_id=162580169216780357220023&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~first_rank_v2~rank_v29-9-114138619.pc_search_result_control_group&utm_term=%E6%B5%8B%E8%AF%95%E8%BD%AF%E4%BB%B6%E5%B7%A5%E7%A8%8B%E5%B8%88%E9%9D%A2%E8%AF%95&spm=1018.2226.3001.4187>

1. **软件测试工程师求职踩过的坑——实战经验的分享啊**

<https://blog.csdn.net/weixin_50829653/article/details/115493417?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-4.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-4.control>

我们都知道，在面对面试官的时候，询问问题的来源大部分就是来自简历。面试的开始流程首先一定是做个**自我介绍**。（小编面试经验也不是很丰富，就是希望大家能够不要睬小编已经踩完的坑。）那在做自我介绍的时候，如果是对自己的项目经验不是十分丰富的话，那么在一开始的时候，就尽量不要让话题首先走到项目这方面。这个时候的技巧就是，多介绍自己。

**就职于几家公司，我之前做过的项目有哪些，什么时间做的，接触的种类都有哪些，从这些项目中你都学习到了那些，一直坚持的副业有那些，或者爱好等等”**（控制一个时间的长度，增加你的个人曝光面积，分散面试官的注意力，或者是让对方充分的了解你，记住回答的时候，千万不要太过于高大上，我们都是凡人，朴实一些，低调才是最大的实力者）。

这个地方有一个在面部表情上需要注意的部分，那就是我们需要做到**面无表情**，来为大家普及一个小知识点。在博弈的时候，面无表情让对方觉得你很有威慑力。其实，面试的本质不就是博弈吗？

面试官：“（这里可能会问道的问题，我是猜不出出来的，**但一定是从你上面的自我介绍而来的，这个时候，就要想好自己肯尼个会面临的问题，这个是需要你前期花时间进行准备的，切记不要回来踩雷的这种。**）**介绍一下自己的项目**吧（这个是一定会走到的部分。）”

面试官：“你当时是怎么测试这个模块的？”

**上面这两类问题，不妨有层次的回答些吧，如果是项目的话，你可以简单的介绍一下自己的项目是做什么的，自己是负责的模块，主要做的测试类型有哪些，比功能、接口、性能、或者是测试报告、测试计划的完成部分等**

**模块的话**，也是类似的回答，主要是让面试官对你做的事情有一个层次性的了解，逐步深入。可以先介绍自己模块是干什么的，测试的关注点是什么，你是怎么进行测试的，测试的时候，都有用到哪些工具。是怎么使用这些工具的，发现的问题有奥谢，或者发现的有趣的bug是什么，等等可以加上自己的一些扩展…

如果你回答让面试官时候比较的满意的话，**面试官会主动和聊一些技术性的问题**。例如你使用的工具是loadrunner，那么面试官就会问你，loadrunner脚本的调试方法有哪些？你在什么地方用到l ?等等。

面试官：**“你们工地的测试流程是什么**？”（这个问题我理解的话，就是看你有没有完整参与过一个项目，或者说，看你们公司的测试流程是怎样的与求职的公司是否是相符的，或者说你们公司的测试流程是否是规范的，等等一系列，看你自己给自己的定位是怎样的吧。）

还有要注意的部分就是，就是你在说自己项目的时候，描述的可能不是很好，对了，想起来了，**模块的测试思路**一定要说出来，或者是**用例的设计思路**，也就是发现问题的出发点。如果前面说的不是很好，对方环视比较满意你这个人的，就会多与你探讨一些工具的用法，例如接口、性能什么的，拉回来你分，想留下你的。如果你的这次表现不是很好的，或者遇到的问题是你之前么有预料到的，可以多王这方面拉拉，主动交谈你对工具的熟悉程度或者是一些设置个人看法。

需要准备的问题是很多的，所以，准备好是一件很重要的事情。最后小编要说一件事情，真实的项目是很关键的，而且**简历上的经验一定要与项目经验结合在一起**。**技术的关键，除了经验剩下的就是知识的储备量**。因此，多点谈资是很关键的。

1. Basic

- 基础**数据库**/ SQL查询和概念的一些实践  
- 一种基本的**脚本语言**（用于自动化测试）  
- 网络和系统**管理概念**将帮助您在系统中把控项目。

1. **如何做自我介绍**

每个面试官都需要通过你用最快的时间去了解你这个人，了解你以前干过什么项目，做过几年，有没有项目经验。

1. **测试工作流程**

为了探明你的能力，他会问一些简单的问题，比如：有没有编写过测试用例，你编写测试用例的时候用到什么方法？你上一家公司的测试流程是什么样子的？简言之，他需要充分了解你在上一家公司工作的流程。

因为他需要与他们自己这家公司现在的流程进行一个比较，需要知道你们上一家公司的工作流程跟他们现在的工作流程是不是一样的，如果不一样，那差距到底大不大？你上家公司的流程是不是正规？是不是一个比较完善的一个流程？了解了这些，他才能判断你是否能够迅速上手，胜任工作。

**一家正规的公司，它们的测试流程应该是**：

1、软件测试的流程是什么？

  分析：每当HR问一个问题的时候我们都可以用1~2s的时间去想HR想要从这个问题中获取什么信息，这点搞清楚之后再去回答就很好回答了。如果有工作经验，直接按照公司流程回答即可，如果是刚转行或者刚实习，那按标准回答即可，文中回答仅供参考；

  回答： **项目经理或者PD把项目需求文档提前下发给相关的研发人员，研发人员抽出一定的时间记录文档内需求不明确或者遗漏的点为后面的评审做准备；在需求评审会议上，各研发人员提出自己的疑问并解决，需求评审最终通过之后会出一份最终的需求规格说明书**；（需求评审阶段）

    需求规格说明书评审通过后，开发经理开始编写开发计划，**测试经理开始编写测试计划，计划评审通过后开发人员开始进行程序的开发，测试人员开始测试用例的编写**，等程序的第一个版本出来后，开发人员进行第二个版本的迭代，这时测试人员对程序进行测试并记录追踪管理缺陷，直到程序迭代完毕。（产品研发阶段）

**程序迭代完毕并修复大部分缺陷后，测试人员开始进行工作的总结，并最终输出一份测试报告书，记录此次的测试工作共，程序存在的相关问题**。

用户把自己想要的东西说出来之后，由产品人员来记录，并且转化成一个**需求文档**。

**测试人员跟开发人员需要在产品人员拿出需求文档之后，进行一个需求的评审**，

需要了解他们用户到底想要一个什么样的功能，想要一个什么的软件。

在评审的过程当中，首**先需要对需求进行一个测试**，测试什么？测试他需求文档中有没有二义性的内容，**有没有描述不准确**，或者是理解不清楚的一些东西，包括你在参加这个需求文档这个会议的期间，产品在讲需求的过程当中，**你有没有觉得这个功能有没有必要**，或者可以删简，可以留到第二个版本在做。这就是第一件事情，需求评审。

需求评审通过之后，**测试人员需要由测试的组长或经理来编写一份测试的计划**，这个计划里边包含的内容会非常的多，这个具体包含什么这里就不细说了，以后有机会的时候再给大家细讲测试计划。

测试计划写完之后，部门会在把所有的测试人员召集到一起开会，对测试计划再进行一个评审。

**评审测试计划**里边什么内容安排的是不是合理，时间的安排是不是真的够用，包括里边的风险是不是规避掉了，还有测试机的准备，系统的准备，测试的一些方式方法，时间的一些这个限制，都是需要写在里边。

评审通过之后再去干什么事情？就是**编写测试用例**。**在测试计划里边会给每一个人进行人员分工，可能张三负责注册模块，李四负责登录模块，王五赵六负责会员中心模块，他们每个人都有自己的负责的那一块功能，他们需要对自己的负责那一块编写测试用例，人员分工安排下去之后，每人编辑好测试用例，就可以开始测试了**。

开始测试用例的评审的目的在于评审他有没有遗漏的点，评审通过之后开始执行测试用例，然后第一轮测试迭代，第二轮测试迭代第三轮测试迭代，直到它验收测试，然后发布上线编写咱们的测试报告，整个这一套流程结束。

**每一轮测试结束之后，都需要给出一个阶段性的测试报告**，第一轮测试结束了，需要给一份测试报告，第二轮结束还是要给测试报告，最后总体的结束了，需要汇总，把所有的bug已解决的未解决的，包括遗留的都需要一个汇总。

**要注意避免的坑**

如果面试单位需要了解到你上一家公司你主要负责的是哪一块业务，那么我建议大家，如果你们去面试的时候，千万不要说你负责注册登录模块。

因为这个工作没有什么技术挑战，什么样的人领导才会分配这样的任务呢，那就是实习生去做，领导绝对不会把这个模块分配给一个技术能力强的人去测。

你应该要说什么？\*\***我是负责什么下单流程的**，或者***负责支付流程***的，或者是负责这个退款流程，这些流程都是比较有逻辑性的内容。\*\***这些东西会涉及到的前后台**，包括审核这个环节都会有。比如你去发布一件商品，需要后台审核通过才能发布，需要涉及到数据库，所以说需要涉及到后台，需要涉及到前台的展示，这些都涉及到很多的逻辑测试。这样的工作是比较有技术含量的。

你认为你在测试过程中遇到了一个**比较逻辑性最强的一个bug**是什么？

问这个问题就需要你们去想想，曾经你们在测的时候遇到了一个逻辑性特别强的bug呢？

面试官这个问题问的目的是什么？

**问的目的是了解你到底有没有真正的测试过**。

还有就是有的面试官会故意的说错一些东西，然后看你的反应，通过这些都能了解你到底会不会，所以想验证一个人到底会不会使用一个工具，不一定非得要考他。

再往深入一点，面试官还会问你有没有**性能测试方面的基础**？**功能测试**这方面，实际上我觉得主要考验人的就是逻辑思维能力，还有你的细心程度能力。

**中级工程师面试秘笈**

针对于中级职位，一般都会把功能问一遍，面试一个中级测试工程师，**着重会问性能自动化跟接口，这是三大重中之重，还有数据库**。

数据库都是其次的，为什么是其次？因为数据库在大学里有讲，基本上上过大学的都会数据库，都会懂得增删改查，再往深入说，就是表连接子查询的问题了，实际工作当中用的也不多。因为我工作这么多年了，在工作当中用到表连接子查询的机会并不是很多。当然这是衡量一个人的技术水平的一个标杆。

着重要说的是**性能测试**，性能的话主要问**稳定性测试**或者**压力测试**，不要求你全会，最起码达到掌握或者了解，在这里面有个误区：面试希望你会修理飞机，但是招进来后却是个拧螺丝的活～没错～这时候要求求职者需要掌握的或者了解的知识面广一些，即使今后用不到，但是不代表面试官不会问到。

举个例子，jmeter都会问到什么是断言，断言干嘛使的，都有哪些断言，怎么连接APP，假如我要测试一个手机的性能测试的时候，要怎么设置，包括它这个聚合报告里边每个指标代表什么意思？它的塑型图，塑型结果怎么看，怎么看它的请求，怎么看它的返回值，每个请求代表什么意思？什么是post，什么是get？这些都会。还有接口测试怎么测？

划重点：往往很多面试者在面试的时候回答的不够完整，比方说接口测试，那就需要从你项目中什么时候开始接口测试，如何使用测试工具，然后说一下测试工具使用和细节，最后还要说一下测试结果（永远要说结果，咱们测试人员重要的就是结果）。

这些都是中级应该会的，如果我阐述的这些问题你都会了那么你就具备中级的测试能力了。

**高级工程师面试秘笈**

**高级面试部分还需要你会写Java会写Python**，需要能解决一些问题，遇到一些疑难杂症的时候，别人解决不了，你能解决，脚本录不了的地方你能录，不用录的方法能写的出来。这就是高级工程师。

高级还能干嘛？不仅能看得懂代码，看得懂脚本，还能找到问题的原因，**知道这个bug是怎么出现的，是由于什么导致这个bug出现的，怎么去解决它**！虽然不用自己去解决，但是告诉开发人员这个问题是由于什么原因导致的，你需要把接口的哪一个代码改掉，把这个参数给换了才能解决这个问题，你需要知道这个问题是怎么出现的，包括解决的方案，并且能够把控整个项目的进度，包括它的时间节点，包括他的所有的人员分工跟安排， 你才能够敢说你自己是一个高级测试工程师。

**问：你在测试中发现了一个bug，但是开发经理认为这不是一个bug，你应该怎样解决**?

答：首先呢将问题提交到缺陷管理库里面进行备案。

然后就要获取判断的依据和标准：

根据需求说明书、产品说明、设计文档等，确认实际结果是否与计划有不一致的地方，提供缺陷是否确认的直接依据;

如果没有文档依据，可以根据类似软件的一般特性来说明是否存在不一致的地方，来确认是否是缺陷;

根据用户的一般使用习惯，来确认是否是缺陷;

与设计人员、开发人员和客户代表等相关人员探讨，确认是否是缺陷;

合理的论述，向测试经理说明自己的判断的理由，注意客观、严谨，不参杂个人情绪。

等待测试经理做出最终决定，如果仍然存在争议，可以通过公司政策所提供的渠道，向上级反映，并有上级做出决定。

**问：给你一个网站，你如何测试?**

答.首先，查找需求说明、网站设计等相关文档，分析测试需求。

制定测试计划，确定测试范围和测试策略，一般包括以下几个部分：**功能性测试;界面测试;性能测试;数据库测试;安全性测试;兼容性测试**

**功能性测试**可以包括，但不限于以下几个方面：

**链接测试**。链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回。

**提交功能**的测试。

多媒体元素是否可以正确加载和显示。

多语言支持是否能够正确显示选择的语言等。

**界面测试**可以包括但不限于一下几个方面：

页面是否风格统一，美观

页面布局是否合理，重点内容和热点内容是否突出

控件是否正常使用

.对于必须但未安装的控件，是否提供自动下载并安装的功能

文字检查

**性能测试**一般从以下两个方面考虑：

**压力测试;负载测试;强度测试**

**数据库测试**要具体决定是否需要开展。数据库一般需要考虑连结性，对数据的存取操作，数据内容的验证等方面。

**安全性测试**：

基本的**登录功能**的检查

是否存在溢出错误，导致系统崩溃或者**权限泄露**

**兼容性测试**，根据需求说明的内容，确定支持的平台组合：

**浏览器的兼容性**;

**操作系统的兼容性**;

软件平台的兼容性;

数据库的兼容性

开展测试，并记录缺陷。合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系(例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容)。

定期评审，对测试进行评估和总结，调整测试的内容。【推荐了解：软件测试课程】

1. 问：一台客户端有三百个客户与三百个客户端有三百个客户对服务器施压，有什么区别?

答：　300个用户在一个客户端上，会占用客户机更多的资源，而影响测试的结果。线程之间可能发生干扰，而产生一些异常。

300个用户在一个客户端上，需要更大的带宽。

IP地址的问题，可能需要使用IP Spoof来绕过服务器对于单一IP地址最大连接数的限制。

所有用户在一个客户端上，不必考虑分布式管理的问题;而用户分布在不同的客户端上，需要考虑使用控制器来整体调配不同客户机上的用户。同时，还需要给予相应的权限配置和防火墙设置。

1. 问：软件的安全性应从哪几个方面去测试?

答：软件安全性测试包括程序、**数据库安全性测试**。根据系统安全指标不同测试策略也不同。

**用户认证安全的测试要考虑问题**： 明确区分系统中**不同用户权限** 、系统中会不会出现用户冲突 、系统会不会因用户的权限的改变造成混乱 、用户登陆密码是否是可见、可复制 、是否可以通过绝对途径登陆系统(拷贝用户登陆后的链接直接进入系统)、用户退出系统后是否删除了所有鉴权标记，是否可以使用后退键而不通过输入口令进入 系统 、系统网络安全的测试要考虑问题 、测试采取的防护措施是否正确装配好，有关系统的补丁是否打上 、模拟非授权攻击，看防护系统是否坚固 、采用成熟的网络漏洞检查工具检查系统相关漏洞(即用最专业的黑客攻击工具攻击试一下，现在最常用的是 NBSI 系列和 IPhacker IP ) 、采用各种木马检查工具检查系统木马情况 、采用各种防外挂工具检查系统各组程序的外挂漏洞

**数据库安全考虑问题**： 系统数据是否机密(比如对银行系统，这一点就特别重要，一般的网站就没有太高要求)、系统数据的完整性(我刚刚结束的企业实名核查服务系统中就曾存在数据 的不完整，对于这个系统的功能实现有了障碍) 、系统数据可管理性 、系统数据的独立性 、系统数据可备份和恢复能力(数据备份是否完整，可否恢复，恢复是否可以完整)

1. 问：简述什么是静态测试、动态测试、黑盒测试、白盒测试、α测试 β测试

答：**静态测试**是不运行程序本身而寻找程序代码中可能存在的错误或评估程序代码的过程。

**动态测试**是实际运行被测程序，输入相应的测试实例，检查运行结果与预期结果的差异，判定执行结果是否符合要求，从而检验程序的正确性、可靠性和有效性，并分析系统运行效率和健壮性等性能

**α测试**是由一个用户在开发环境下进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的受控测试，Alpha测试不能由程序员或测试员完成。

**β测试**是软件的多个用户在一个或多个用户的实际使用环境下进行的测试。开发者通常不在测试现场，Beta测试不能由程序员或测试员完成。

1. **问：如何测试一个纸杯?**

答：**功能度**：用水杯装水看漏不漏;水能不能被喝到

**安全性**：杯子有没有毒或细菌

**可靠性**：杯子从不同高度落下的损坏程度

可移植性：杯子在不同的地方、温度等环境下是否都可以正常使用

**兼容性**：杯子是否能够容纳果汁、白水、酒精、汽油等

易用性：杯子是否烫手、是否有防滑措施、是否方便饮用

用户文档：使用手册是否对杯子的用法、限制、使用条件等有详细描述

**疲劳测试**：将杯子盛上水(案例一)放24小时检查泄漏时间和情况;盛上汽油(案例二)放24小时检查泄漏时间和情况等

**压力测试**：用根针并在针上面不断加重量，看压强多大时会穿透

1. 问：黑盒测试的测试用例常见设计方法都有哪些?请分别以具体的例子来说明这些方法在测试用例设计工作中的应用。

2)**边界值分析法**：是对等价类划分方法的补充。测试工作经验告诉我,大量的错误是发生在输入或输出范围的边界上,而不是发生在输入输出范围的内部.因此针对各种边界情况设计测试用例,可以查出更多的错误.

使用边界值分析方法设计测试用例,首先应确定边界情况.通常输入和输出等价类的边界,就是应着重测试的边界情况.应当选取正好等于,刚刚大于或刚刚小于边界的值作为测试数据,而不是选取等价类中的典型值或任意值作为测试数据.

3)**错误猜测法**：基于经验和直觉推测程序中所有可能存在的各种错误, 从而有针对性的设计测试用例的方法.

错误推测方法的基本思想: 列举出程序中所有可能有的错误和容易发生错误的特殊情况,根据他们选择测试用例. 例如, 在单元测试时曾列出的许多在模块中常见的错误. 以前产品测试中曾经发现的错误等, 这些就是经验的总结. 还有, 输入数据和输出数据为0的情况. 输入表格为空格或输入表格只有一行. 这些都是容易发生错误的情况. 可选择这些情况下的例子作为测试用例.

5)**正交表分析法**：可能因为大量的参数的组合而引起测试用例数量上的激增，同时，这些测试用例并没有明显的优先级上的差距，而测试人员又无法完成这么多数量的测试，就可以通过正交表来进行缩减一些用例，从而达到尽量少的用例覆盖尽量大的范围的可能性。

6)**场景分析方法**：指根据用户场景来模拟用户的操作步骤，这个比较类似因果图，但是可能执行的深度和可行性更好。

1. 问：详细的描述一个测试活动完整的过程。(供参考，本答案主要是**瀑布模型**的做法)

答：项目经理通过和客户的交流，完成需求文档，由开发人员和测试人员共同完成需求文档的评审，评审的内容包括：需求描述不清楚的地方和可能有明显冲突或者无法实现的功能的地方。项目经理通过综合开发人员，测试人员以及客户的意见，完成项目计划。然后SQA进入项目，开始进行统计和跟踪

开发人员根据需求文档完成需求分析文档，测试人员进行评审，评审的主要内容包括是否有遗漏或双方理解不同的地方。测试人员完成测试计划文档，测试计划包括的内容上面有描述。

测试人员根据修改好的需求分析文档开始写测试用例，同时开发人员完成概要设计文档，详细设计文档。此两份文档成为测试人员撰写测试用例的补充材料。

测试用例完成后，测试和开发需要进行评审。

测试人员搭建环境

开发人员提交第一个版本，可能存在未完成功能，需要说明。测试人员进行测试，发现BUG后提交给BugZilla。

开发提交第二个版本，包括Bug Fix以及增加了部分功能，测试人员进行测试。

重复上面的工作，一般是3-4个版本后BUG数量减少，达到出货的要求

1. **问：****你对测试最大的兴趣在哪里?为什么?**

回答这个面试题，没有固定统一的答案，但可能是许多企业都会问到的。提供以下答案供考：

答：　最大的兴趣，感觉这是一个有挑战性的工作;

测试是一个经验行业，工作越久越能感觉到做好测试的难度和乐趣

通过自己的工作，能使软件产品越来越完善，从中体会到乐趣

回答此类问题注意以下几个方面：

尽可能的切合招聘企业的技术路线来表达你的兴趣，例如该企业是数据库应用的企业，那么表示你的兴趣在数据库的测试，并且希望通过测试提升自己的数据库掌握能力。

表明你做测试的目的是为了提升能力，也是为了更好的做好测试;提升能力不是为了以后转开发或其他的，除非用人企业有这样的安排。

不要过多的表达你的兴趣在招聘企业的范畴这外。比如招聘企业是做财务软件的，可是你表现出来的是对游戏软件的兴趣;或招聘是做JAVA开发的，而你的兴趣是在C类语言程序的开发。

1. **问：****你自认为测试的优势在哪里?**

答：该面试也没有固定不变的答案，但可参考以下几点，并结合自身特点：

有韧性、有耐心、做事有条理性、喜欢面对挑战、有信心做好每一件事情、较强的沟通能力、从以前的经理处都得到了很好的评价表明我做的很好

1. **问：简述你在以前的工作中做过哪些事情，比较熟悉什么**。参考答案如下。

答：　我过去的主要工作是系统测试和自动化测试。在**系统测试**中，主要是对BOSS系统的业务逻辑功能，以及软交换系统的Class 5特性进行测试。**性能测试**中，主要是进行的压力测试，在各个不同数量请求的情况下，获取系统响应时间以及系统资源消耗情况。**自动化测试**主要是通过自己写脚本以及一些第三方工具的结合来测试软交换的特性测试。

在测试中，我感觉对用户需求的完全准确的理解非常重要。另外，就是对BUG的管理，要以需求为依据，并不是所有BUG均需要修改。

测试工作需要耐心和细致，因为在新版本中，虽然多数原来发现的BUG得到了修复，但原来正确的功能也可能变得不正确。因此要注重迭代测试和回归测试。

问：你认为做好测试计划工作的关键是什么?

答：　明确测试的目标，增强测试计划的实用性

编写软件测试计划得重要目的就是使测试过程能够发现更多的软件缺陷，因此软件测试计划的价值取决于它对帮助管理测试项目，并且找出软件潜在的缺陷。因此，软件测试计划中的测试范围必须高度覆盖功能需求，测试方法必须切实可行，测试工具并且具有较高的实用性，便于使用，生成的测试结果直观、准确

**坚持“5W”规则，明确内容与过程**

“5W”规则指的是“What(做什么)”、“Why(为什么做)”、“When(何时做)”、“Where(在哪里)”、“How(如何做)”。利用“5W”规则创建软件测试计划，可以帮助测试团队理解测试的目的(Why)，明确测试的范围和内容(What)，确定测试的开始和结束日期(When)，指出测试的方法和工具(How)，给出测试文档和软件的存放位置(Where)。

采用评审和更新机制，保证测试计划满足实际需求

测试计划写作完成后，如果没有经过评审，直接发送给测试团队，测试计划内容的可能不准确或遗漏测试内容，或者软件需求变更引起测试范围的增减，而测试计划的内容没有及时更新，误导测试执行人员。

分别创建测试计划与测试详细规格、测试用例

应把详细的测试技术指标包含到独立创建的测试详细规格文档，把用于指导测试小组执行测试过程的测试用例放到独立创建的测试用例文档或测试用例管理数据库中。测试计划和测试详细规格、测试用例之间是战略和战术的关系，测试计划主要从宏观上规划测试活动的范围、方法和资源配置，而测试详细规格、测试用例是完成测试任务的具体战术。

49、

问：您认为做好测试用例设计工作的关键是什么?

答：白盒测试用例设计的关键是以较少的用例覆盖尽可能多的内部程序逻辑结果

黑盒法用例设计的关键同样也是以较少的用例覆盖模块输出和输入接口。不可能做到完全测试，以最少的用例在合理的时间内发现最多的问题

50、

问：你的测试职业发展目标是什么?

答：测试经验越多，测试能力越高。所以我的职业发展是需要时间累积的，一步步向着高级测试工程师奔去。而且我也有初步的职业规划，前3年累积测试经验，不断的更新自己改正自己，做好测试任务。

1. 问：**你对测试最大的兴趣在哪里?为什么?**

答：　**最大的兴趣就是测试有难度，有挑战性**!

我觉得做测试整个过程中有2点让我觉得很有难度(对我来说，有难度的东西我就非常感兴趣)，**第一是测试用例的设计**，因为测试的精华就在测试用例的设计上了，**要在版本出来之前，把用例写好，用什么测试方法写**?(也就是测试计划或测试策略)，如果你刚测试一个新任务时，你得花一定的时间去消化业务需求和技术基础，业务需求很好理解(多和产品经理和开发人员沟通就能达到目的)，而技术基础可就没那么简单了，这需要你自觉的学习能力，比如说网站吧，最基本的技术知识你要知道网站内部是怎么运作的的，后台是怎么响应用户请求的?**测试环境如何搭建**?这些都需要最早的学好。至少在开始测试之前能做好基本的准备，可能会遇到什么难题?需求细节是不是没有确定好?这些问题都能在设计用例的时候发现。

第二是发现**BUG的时候了**，这应该是测试人员最基本的任务了，一般按测试用例开始测试就能发现大部分的bug，**还有一部分bug需要测试的过程中更了解所测版本的情况获得更多信息，补充测试用例，测试出bug**。还有如何发现bug?这就需要在测试用例有效的情况下，通过细心和耐心去发现bug了，每个用例都有可能发现bug，每个地方都有可能出错，所以测试过程中思维要清晰(测试过程数据流及结果都得看仔细了，bug都在里面发现的)。如何描述bug也很有讲究，bug在什么情况下会产生，如果条件变化一点点，就不会有这个bug，以哪些最少的操作步骤就能重现这个bug，这个bug产生的规律是什么?如果你够厉害的话，可以帮开发人员初步定位问题。

1. **问**：**您所熟悉的软件测试类型都有哪些?请试着分别比较这些不同的测试类型的区别与联系(如功能测试、性能测试……)**

答：测试类型有：功能测试，性能测试，界面测试。

**功能测试**在测试工作中占的比例最大，功能测试也叫**黑盒测试**。是把测试对象看作一个黑盒子。利用黑盒测试法进行动态测试时，需要测试软件产品的功能，不需测试软件产品的内部结构和处理过程。采用黑盒技术设计测试用例的方法有：等价类划分、边界值分析、错误推测、因果图和综合策略。

**性能测试**是通过自动化的测试工具模拟多种**正常**、**峰值**以及**异常负载条件**来对系统的各项性能指标进行测试。**负载测试和压力测试**都属于性能测试，两者可以结合进行。通过**负载测试**，确定在各种工作负载下系统的性能，**目标是测试当负载逐渐增加时，系统各项性能指标的变化情况**。**压力测试**是通过确定一个系统的**瓶颈或者不能接收的性能点**，**来获得系统能提供的最大服务级别的测试**。

界面测试，界面是软件与用户交互的最直接的层，界面的好坏决定用户对软件的第一印象。而且设计良好的界面能够引导用户自己完成相应的操作，起到向导的作用。同时界面如同人的面孔，具有吸引用户的直接优势。设计合理的界面能给用户带来轻松愉悦的感受和成功的感觉，相反由于界面设计的失败，让用户有挫败感，再实用强大的功能都可能在用户的畏惧与放弃中付诸东流。

区别在于，**功能测试**关注产品的所有功能上，要考虑到每个细节功能，每个可能存在的功能问题。**性能测试Performance testing主要关注于产品整体的多用户并发下的稳定性和健壮性**。界面测试更关注于用户体验上，用户使用该产品的时候是否易用，是否易懂，是否规范(快捷键之类的)，是否美观(能否吸引用户的注意力)，是否安全(尽量在前台避免用户无意输入无效的数据，当然考虑到体验性，不能太粗鲁的弹出警告)?做某个性能测试的时候，首先它可能是个功能点，首先要保证它的功能是没问题的，然后再考虑该功能点的性能测试

1. 问：请试着比较一下黑盒测试、白盒测试、单元测试、集成测试、系统测试、验收测试的区别与联系。

答：　**黑盒测试**：已知产品的功能设计规格，可以进行测试证明每个实现了的功能是否符合要求。

**白盒测试**：已知产品的内部工作过程，可以通过测试证明每种内部操作是否符合设计规格要求，所有内部成分是否以经过检查。

**软件的黑盒测试**意味着测试要在软件的接口处进行。这种方法是把测试对象看做一个黑盒子，测试人员完全不考虑程序内部的逻辑结构和内部特性，只依据程序的需求规格说明书，检查程序的功能是否符合它的功能说明。因此黑盒测试又叫功能测试或数据驱动测试。黑盒测试主要是为了发现以下几类错误：

1、**是否有不正确或遗漏的功能?**

**2、在接口上，输入是否能正确的接受?能否输出正确的结果?**

**3、是否有数据结构错误或外部信息(例如数据文件)访问错误?**

**4、性能上是否能够满足要求?**

**5、是否有初始化或终止性错误**?

软件的白盒测试是对软件的过程性细节做细致的检查。这种方法是把测试对象看做一个打开的盒子，它允许测试人员利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序所有逻辑路径进行测试。通过在不同点检查程序状态，确定实际状态是否与预期的状态一致。因此白盒测试又称为结构测试或逻辑驱动测试。白盒测试主要是想对程序模块进行如下检查：

1、对程序模块的所有独立的执行路径至少测试一遍。

2、对所有的逻辑判定，取“真”与取“假”的两种情况都能至少测一遍。

3、在循环的边界和运行的界限内执行循环体。

4、测试内部数据结构的有效性，等等。

单元测试(模块测试)是开发者编写的一小段代码，用于检验被测代码的一个很小的、很明确的功能是否正确。通常而言，一个单元测试是用于判断某个特定条件(或者场景)下某个特定函数的行为。

**单元测试是由程序员自己来完成，最终受益的也是程序员自己**。可以这么说，程序员有责任编写功能代码，同时也就有责任为自己的代码编写单元测试。执行单元测试，就是为了证明这段代码的行为和我们期望的一致。

**集成测试(也叫组装测试，联合测试)是单元测试的逻辑扩展**。它的最简单的形式是：两个已经测试过的单元组合成一个组件，并且测试它们之间的接口。从这一层意义上讲，组件是指多个单元的集成聚合。在现实方案中，许多单元组合成组件，而这些组件又聚合成程序的更大部分。方法是测试片段的组合，并最终扩展进程，将您的模块与其他组的模块一起测试。最后，将构成进程的所有模块一起测试。

**系统测试**是将经过测试的子系统装配成一个完整系统来测试。它是检验系统是否确实能提供系统方案说明书中指定功能的有效方法。(常见的联调测试)

**系统测试的目的是对最终软件系统进行全面的测试**，确保最终软件系统满足产品需求并且遵循系统设计。

验收测试是部署软件之前的最后一个测试操作。验收测试的目的是确保软件准备就绪，并且可以让最终用户将其用于执行软件的既定功能和任务。

验收测试是向未来的用户表明系统能够像预定要求那样工作。经集成测试后，已经按照设计把所有的模块组装成一个完整的软件系统，接口错误也已经基本排除了，接着就应该进一步验证软件的有效性，这就是验收测试的任务，即软件的功能性能如同用户所合理期待的那样。

1. 问：你在测试中发现了一个bug，但是开发经理认为这不是一个bug，你应该怎样解决？

首先，将问题提交到缺陷管理库里面进行备案。

然后，要获取判断的依据和标准：

根据需求说明书、产品说明、设计文档等，确认实际结果是否与计划有不一致的地方，提供缺陷是否确认的直接依据；

如果没有文档依据，可以根据类似软件的一般特性来说明是否存在不一致的地方，来确认是否是缺陷；

根据用户的一般使用习惯，来确认是否是缺陷；

与设计人员、开发人员和客户代表等相关人员探讨，确认是否是缺陷；

合理的论述，向测试经理说明自己的判断的理由，注意客观、严谨，不参杂个人情绪。

等待测试经理做出最终决定，如果仍然存在争议，可以通过公司政策所提供的渠道，向上级反映，并有上级做出决定

3、缺陷方面（有问到）

1. **描述一个你印象最深刻的bug**

在做上传视频的测试时，发现华为荣耀V10上传手机自带相机专业模式录制的视频会闪退。而ios上传同个视频提示合成失败。

我将手机自带相机录制的专业模式和普通模式录制的同样时长的视频发到电脑上，用格式工厂软件查看视频的不同之处，之后发现视频编码是不同的。

我继续网上查阅了视频编码方面的知识，发现mp4视频有几种编码，而继续测试验证发现我们的app上传的视频只支持mp4视频中的H.264编码格式。于是提交了视频上传不支持非H.264格式的视频。并补充完善了相关用例。

（因为在公司没有查日志权限，所以其实应该先查日志

1. 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。

但是在没有需求分析文档的前提下，来设计测试用例，可以考查一个测试人员的基本功，比如考虑问题是否全面，设计测试用例的方法是否合理等。

**一般是根据自己的日常经验和测试的思维来设计测试用例**。在设计测试用例时一般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**。

1. **压力测试（Stress Testing）**

压力测试的主要任务就是获取**系统正确运行的极限**，检查系统在瞬间峰值负荷下正确执行的能力。例如，对服务器做压力测试时就可以**增加并发操作的用户数量**；或者**不停地向服务器发送请求**；或**一次性向服务器发送特别大的数据**等。看看服务器保持正常运行所能达到的最大状态。人们通常使用测试工具来完成压力测试，如模拟上万个用户从终端同时登录，这是压力测试中常常使用的方法。

1. 您在以往的测试工作中都曾经具体从事过哪些工作？其中最擅长哪部分工作？  
   参考答案：（根据项目经验不同，灵活回答即可）

我曾经做过web测试，后台测试，客户端软件，其中包括功能测试，性能测试，用户体验测试。最擅长的是功能测试

您所熟悉的软件测试类型都有哪些？请试着分别比较这些不同的测试类型的区别与联系（如功能测试、性能测试……）

参考答案：

测试类型有：功能测试，性能测试，界面测试。

**功能测试**在测试工作中占的比例最大，功能测试也叫黑盒测试。是把测试对象看作一个黑盒子。利用黑盒测试法进行动态测试时，需要测试软件产品的功能，不需测试软件产品的内部结构和处理过程。采用黑盒技术设计测试用例的方法有：等价类划分、边界值分析、错误推测、因果图和综合策略。

**性能测试**是通过自动化的测试工具模拟多种正常、峰值以及异常负载条件来对系统的各项性能指标进行测试。负载测试和压力测试都属于性能测试，两者可以结合进行。**通过负载测试，确定在各种工作负载下系统的性能，目标是测试当负载逐渐增加时，系统各项性能指标的变化情况**。**压力测试是通过确定一个系统的瓶颈或者不能接收的性能点，来获得系统能提供的最大服务级别的测试**。

　　界面测试，界面是软件与用户交互的最直接的层，界面的好坏决定用户对软件的第一印象。而且设计良好的界面能够引导用户自己完成相应的操作，起到向导的作用。同时界面如同人的面孔，具有吸引用户的直接优势。设计合理的界面能给用户带来轻松愉悦的感受和成功的感觉，相反由于界面设计的失败，让用户有挫败感，再实用强大的功能都可能在用户的畏惧与放弃中付诸东流。

　　区别在于，功能测试关注产品的所有功能上，要考虑到每个细节功能，每个可能存在的功能问题。性能测试主要关注于产品整体的多用户并发下的稳定性和健壮性。界面测试更关注于用户体验上，用户使用该产品的时候是否易用，是否易懂，是否规范（快捷键之类的），是否美观（能否吸引用户的注意力），是否安全（尽量在前台避免用户无意输入无效的数据，当然考虑到体验性，不能太粗鲁的弹出警告）？做某个性能测试的时候，首先它可能是个功能点，首先要保证它的功能是没问题的，然后再考虑该功能点的性能测试

1. 请以您以往的实际工作为例，详细的描述一次测试用例设计的完整的过程。

就说最近的这次网站功能的测试吧

　　首先：得到**相关文档**（需求文档和设计文档），**理解需求和设计设计思想**后，想好测试策略（测试计划简单点就OK了），**考虑到测试环境，测试用例，测试时间等问题**。

　　第二步：**设计测试用例**，测试策略是：**把网站部分的功能点测试完，然后在进行系统测试**（另外个模块呢有另一个测试人员负责，可以进行联调测试），**网站模块的测试基本是功能测试和界面测试**（**用户并发的可能性很小，所以不考虑**）：**这次的网站的输入数据呢是使用数据库中的某张表记录**，如果表中某一数据记录中新加进来的（还没有被处理的，有个标志位），网站启动后会立刻去刷那张表，得到多条数据，然后在进行处理。处理过程中，会经历3个步骤，网站才算完成了它的任务。有3个步骤呢，就可以分别对　　**这3个步骤进行测试用例的设计,尽量覆盖到各种输入情况（包括数据库中的数据，用户的输入等），得出了差不多50个用例**。界面测试，也就是用户看的到的地方，包括发送的邮件和用户填写资料的页面展示。

　　第三步：搭建测试环境（为什么这个时候考虑测试环境呢？因为我对网站环境已经很熟了，只有有机器能空于下来做该功能测试就可以做了），因为网站本身的环境搭建和其他的系统有点不同，它需要的测试环境比较麻烦，需要web服务器（Apache,tomcat），不过这次需求呢，网站部分只用到了tomcat，所以只要有tomcat即可

　　第四步：执行测试

1. 在网站流量逐渐加大的情况下，开始考虑做性能测试了，首先要写好性能测试计划，**根据运营数据得出流量最大的页面**（如果是第一次的话，一般是首页，下载页，个人帐户页流量最大，而且以某种百分比），

Web服务器指标指标：

**Avg Rps**: 平均每秒钟响应次数＝总请求时间 / 秒数；

**Successful Rounds**：成功的请求；

**Failed Rounds** ：失败的请求；

**Successful Hits** ：成功的点击次数；

**Failed Hits** ：失败的点击次数；

**Hits Per Second** ：每秒点击次数；

Successful Hits Per Second ：每秒成功的点击次数；

Failed Hits Per Second ：每秒失败的点击次数；

**Attempted Connections** ：尝试链接数；

1. 软件的构造号与版本号之间的区别？BVT(BuildVerificationTest)

参考答案：版本控制命名格式: 主版本号.子版本号[.修正版本号[.编译版本号 ]]

Major.Minor [.Revision[.Build]]

应根据下面的约定使用这些部分：

Major ：具有相同名称但不同主版本号的程序集不可互换。例如，这适用于对产品的大量重写，这些重写使得无法实现向后兼容性。

Minor ：如果两个程序集的名称和主版本号相同，而次版本号不同，这指示显著增强，但照顾到了向后兼容性。例如，这适用于产品的修正版或完全向后兼容的新版本。

Build ：内部版本号的不同表示对相同源所作的重新编译。这适合于更改处理器、平台或编译器的情况。

Revision ：名称、主版本号和次版本号都相同但修订号不同的程序集应是完全可互换的。这适用于修复以前发布的程序集中的安全漏洞。

**BVT(BuildVerificationTest)**：

作为Build的一部分，主要是通过对基本功能、特别是关键功能的测试，保证新增代码没有导致功能失效，保证版本的持续稳定。实现BVT方式是有以下几种：

1、**测试人员手工验证关键功能实现的正确性**。特点：这是传统开发方法中，通常采用的方式。无需维护测试脚本的成本，在测试人力资源充足，测试人员熟悉业务、并对系统操作熟练情况下效率很高，比较灵活快速。缺点：人力成本较高；对测试人员能力有一定要求；测试人员面对重复的工作，容易产生疲倦懈怠，从而影响测试质量。

2、**借助基于GUI的自动化功能测试工具来完成**，将各基本功能操作录制成测试脚本，每次回放测试脚本验证功能实现的正确性。特点：**能够模拟用户操作完成自动的测试，从UI入口到业务实现，每一层的代码实现都经过验证**；节约人力成本；降低测试人员重复劳动的工作量，机器不会疲倦；缺点：**对于UI变动比较频繁的系统来说**，这种方式的维护成本很高，实施起来非常困难。另外，在项目周期较短且后续无延续性或继承的情况下，也不推荐使用此方式。

3、**由开发人员通过自动化测试工具完成业务层的BVT测试**。特点：通过对业务层关键功能的持续集成测试，保证系统功能的持续稳定。**可以结合DailyBuild**，做为Build的一部分，自动实现并输入BVT报告。缺点：仅对业务规则实现的正确性进行了测试，对表现层无法测试到，对于诸如：前台页面控件各种事件响应、页面元素变化等方面的问题无法保证。

1. 软件的安全性应从哪几个方面 去测试？

软件安全性测试包括程序、数据库安全性测试。根据系统安全指标不同测试策略也不同。

用户认证安全的测试要考虑问题：

**明确区分系统中不同用户权限**

**用户登陆密码是否是可见、可复制**

**是否可以通过绝对途径登陆系统**（拷贝用户登陆后的链接直接进入系统）

**用户退出系统后是否删除了所有鉴权标记，是否可以使用后退键而不通过输入口令进入系统**

1. 手机APP测试

主要包括功能、性能测试、稳定性、兼容性、用户测试。

性能测试： CPU占用/内存占用 /耗电测试 /流量消耗测试 /安装包大小 /加载时间测试 /核心功能相应时间 (①启动时间检测：检测App在终端上首次启动时间。②内存、CPU耗用检测：检测App在终端上运行时不同时段占用内存、CPU情况。③流量耗用检测：检测App在终端上运行时的网络流量消耗情况。④电池温度检测：检测App在终端上运行时，对终端的电池温度等性能指标的影响情况 )

兼容性测试： 屏幕分辨率 /网络状态，状态切换 /android版本 /安装卸载升级等 /权限设置 /与其他APP兼容性 (①安装卸载测试：测试App在指定终端上是否可正常安装、正常卸载，准确定位错误原因。②遍历测试：自动识别App可执行的功能，在一定时间内遍历App的不同功能界面，通过截图记录操作路径 并输出日志、定位异常现象。③运行稳定性测试：类似Monkey的随机性压力测试，测试App运行期的稳定性。④UI适配测试：测试App的UI与目标终端的屏幕是否适配，记录是否存在渲染失败、错位、黑边框、黑白屏等现象。)

稳定性测试包括：服务器异常时稳定性 /外部事件影响（电话，短信等） /内存是否有溢出或者泄漏 /多线程问题 。

1. **性能测试常用指标**：

从外部看，主要有

1、吞吐量：每秒钟系统能够处理的请求数，任务数

2、响应时间：服务处理一个请求或一个任务的耗时

3、错误率：一批请求中结果出错的请求所占比例

从服务器的角度看，性能测试关注CPU，内存，服务器负载，网络，磁盘ＩＯ

对登录功能做性能测试

单用户登陆的响应界面是否符合预期

单用户登陆时后台请求数量是否过多

高并发场景下用户登录的响应界面是否符合预期

高并发场景下服务端的监控指标是否符合预期

高集合点并发场景下是否存在资源死锁和不合理的资源等待

长时间大量用户连续登录和登出，服务器端是否存在内存泄漏

1. **测试实战面试题**

请设计一个关于ATM自动取款机的测试用例。

1）功能

a)ATM所识别卡的类型；

b)密码验证(身份登陆、是否为掩码、输入错误密码时是否提示，连续三次错误吞卡等)；

c)取款功能：

i、金额多少的限制，单次最大最小提取金额、每天最大提取金额等）；

Ii、取款币种的不同，如人民币、美元、欧元等。

d)是否提示客户操作完成后，打印相关操作信息；

e)查询功能是否正常；

f)转账功能是否正常；

g)是否提示客户操作完成后，取回客户卡；

2）性能

a)是否有自动吞卡：非法客户\密码错误客户\规定时间内未完成相关操作功能的客户。（如果有，有无报警功能(保密报警)）

b)平均无故障时间，平均故障修复时间，输入密码后验证时间，出钞票时间，查询余额等待时间。

3）易用性

a)ATM各个操作功能(硬件)是否正常、易懂；

b)ATM的界面显示是否友好；

c)ATM是否支持英文操作；

d)ATM是否存在异常（断电、黑客入侵）有自动保护（报警）功能；

如何测试一个 纸杯？

功能度：用水杯装水看漏不漏；水能不能被喝到

安全性：杯子有没有毒或细菌

可靠性：杯子从不同高度落下的损坏程度

可移植性：杯子在不同的地方、温度等环境下是否都可以正常使用

兼容性：杯子是否能够容纳果汁、白水、酒精、汽油等

易用性：杯子是否烫手、是否有防滑措施、是否方便饮用

用户文档：使用手册是否对杯子的用法、限制、使用条件等有详细描述

疲劳测试：将杯子盛上水（案例一）放 24 小时检查泄漏时间和情况；盛上汽油（案例二）

放 24 小时检查泄漏时间和情况等

压力测试：用根针并在针上面不断加重量，看压强多大时会穿透

我手上这支笔，请你根据这支笔设计测试用例

首先我要测它的外观、颜色是否符合要求、所占的空间是多大、是否环保、接下来测它的质量、这支笔是否能够写字流畅、写出的自得颜色是否符合要求、能使用多长时间等

测试手机开机键

功能测试：按下开机键，屏幕能否亮起

性能测试：按下开机键，屏幕能否在规定时间内亮起

压力测试：连续多次按下开机键，观察屏幕是否能一直亮起，到多久时间失灵

健壮性测试：给定一个中了病毒的手机或者是淘汰许久的老机子，安歇开机键观察屏幕能否亮起

可靠性测试：连续按下开机键有限次数，比如1万次，记录屏幕未亮起的次数

可用性测试：开机键按下费不费力，开机键的形状设计是否贴合手指，开机键的位置设计是否方便

***如何回答登录功能怎么进行测试？***

首先，进行界面测试。

查看界面上的所有元素是否齐全；

没有输入内容时，是否有相应的提示语；

验证码是否能够显示；

移动鼠标，【登陆】按钮默认不能点击；

【忘记密码】是否有个小问号“？”（其他都有）；

第二，进行功能测试。

输入正确的用户名、密码、验证码，点【登陆】能登陆；

输入正确的用户名、错误的密码、正确的验证码，提示用户名或密码错误；

输入错误的用户名、正确的验证码，提示用户名或密码错误；

输入正确的用户名、密码，错误的验证码，提示验证码错误；

输入不符合规则的手机号或者邮箱应该提示错误；

页面长时间不登陆和操作，验证码会不会过期；

点【记住密码】，登录后退出，再次登陆是不是可以不输入密码；

点【忘记密码】能够跳转到密码设置页面（至于是什么不用管，就是能不能跳转）

只点击验证码图案，验证码能不能刷新；

页面刷新，验证码图案能不能刷新；

输入栏是否设置快速删除按钮；

用户名和密码是否大小写敏感；

用户名和密码前后有空格的处理；

登陆成功，是否有记住密码功能；

登陆失败后，不能记录密码的功能；

新用户第一次登陆成功，是否有修改密码提示；

用户登录过程中log中是否有个人信息明文打印；

是否支持第三方登陆；

刷新页面时是否会刷新验证码；

输入密码的时候，大写键盘开启的时候要有提示信息 ；

不同级别的用户，比如管理员用户和普通用户，登录系统后的权限是否正确；

第三、业务安全测试。

有没有登陆错误次数的限制；

每次登陆错误之后有没有限制再次登陆的时间间隔；

是否支持一个账号多地登陆；

不同机型登陆，异地登陆是否有提醒 ；

不登录的情况下，在浏览器中直接输入登录后的URL地址，验证是否会重新定向到用户登录界面；

第四、**兼容性测试**。

在**相同浏览器的不同版本上打开登录页面**，效果是否一致；在**不同浏览器上**打开登录页面，效果是否一致；在**不同操作系统的不同浏览器**打开登录页面，效果是否一致；在不同的**屏幕分辨率下打开登录页面**，效果是否一致；

第五、代码安全性测试。

用户输入登录信息登陆时，个人信息是不是会显示在浏览器地址栏；

用户登陆的时候，通过抓包工具抓数据，密码是否加密；

查看页面源代码，验证码是否直接显示在代码中；

密码在后台储存时是否加密；

是否可以使用登录的API发送登录请求，并绕开验证码校验；

用户名和密码的输入框中分别输入典型的“SQL注入攻击”字符串，验证系统的返回页面；

用户名和密码的输入框中分别输入典型的“XSS跨站脚本攻击”字符串，验证系统行为是否被篡改；

第六、页面性能测试。

**单用户登录的响应时间是否小于3秒**；

通过工具向登录页发起大量请求，查看页面响应时间的变化；

**通过工具对登陆功能进行并发测试；通过工具向登录页发起大量请求，查看页面何时崩溃**；

通过工具向登录页发起大量请求，查看页面崩溃后有没有良好的提示信息；

通过工具向登录页发起大量请求，查看页面崩溃后多长时间能够恢复服务；

弱网，不同网速时登陆的时间，网络切换和网络延迟时登陆界面是否正常；

最后、易用性测试。

页面是否美观；

功能是否都可以使用；

页面速度快不快；

页面元素加载是否耗费网络流量；

能不能第三方登陆；

为什么不使用手机验证码登陆；

输入框能否可以以Tab键切换。

**如何回答京东购物车功能怎么进行测试？**

1.功能测试

a）、未登录时：

将商品加入购物车，页面跳转到登录页面，登录成功后购物车数量增加。

b）、登录后：

所有链接是否跳转正确；

商品是否可以成功加入购物车；

没有限购要求的商品，添加数量能不能超过库存数；

购物车商品总数是否有限制；

商品总数统计是否正确；

全选功能是否可用；

删除功能是否可用；

删除功能是否有提示；

价格总计是否正确；

商品文字太长时是否显示完整；

购物车中下架的商品是否有标识，是否还能支付；

新加入购物车商品排序（添加购物车中存在的店铺的商品和购物车中不存在的店铺的商品）；

是否支持快TAB、ENTER等快捷键；

商品删除后商品总数是否减少；

收藏功能是否可用；

账号退出后，购物车添加的内容是否还在；

购物车结算功能是否可用。

限购商品按照规则购买完成后，还能不能再次添加购物车并购买；

2.兼容性测试

BS架构：不同浏览器测试，比如：IE，火狐，谷歌，360这些。

APP：在主流的不同类型，不同分辨率，不同操作系统的手机上测试，华为，vivo，oppo等

3.用户体验测试

删除商品是否有提示；

是否支持快捷键功能；

是否有回到顶部的功能；

商品过多时结算按钮是否可以浮动显示；

购物车有多个商品时，能不能只对单个商品结算；

界面布局、排版是否合理；

文字是否显示清晰；

不同卖家的商品是否区分明显。

4.性能测试

打开购物车页面要多长时间

支付流程测试

功能测试。

用等价类和边界值，判断支付的金额

如果没有登陆能否支付，支付成功后是否可以正常跳转；

支付方式是否支持扫码支付，第三方平台支付（支付包，云网等），语音支付，指纹支付；

支付时是否需要身份验证，支付后有无手机短信提示，是否可以找他人代付；

用边界值法有无支付额度限制，余额不足时有无提示，支付时是否是动态加密支付；

待支付状态：订单是否可以正常支付；是否可以取消；有相同订单是否可以支付两次；

是否可以扫码支付，输入错误的密码会怎样显示，有无错误次数限制；

若支持扫码支付，二维码是否支持支付包和微信扫码，若两人同时扫描怎么处理；

有无最小支付金额限制，无意义的支付金额0，重复支付如何处理；

如果支付包含优惠金额，该怎么处理优惠额度；

性能测试

弱网，无网时是否可以支付；

退款到账时间，耗电量的多少；

带负载情况下的响应时间和吞吐率，在某个时间段内同时访问系统的用户数量 ；

压力测试

多人同时付款；

界面测试；

支付界面有无错别字，排版是否合理，颜色搭配是否合理；

兼容性测试

是否可以跨平台，不同电脑机型下显示有无区别；

安全性测试；

若支付不成功是否原路退款，若支付成功，有无支付信息提示；

用fiddler抓包尝试修改价格，对订单金额有无效验；

直接输入需要权限的页面地址可用访问；

接口测试

第三方平台支付

对于有系统大量并发访问，你会如何做测试，有什么建议

参考回答：

如何做高并发系统的测试，一般而言，整体的测试策略是：先针对部分系统进行性能测试及压力测试，得到各部分的峰值处理性能，再模拟整体流程测试，重点测试整体业务流程以及业务预期负荷，着重测试以下几点：

1、不同省份，不同运营商CDN节点性能，可采用典型压力测试方案

2、核心机房BGP网络带宽，此部分重点在于测试各运行商的BGP网络可靠性，实际速率，一般采用smokeping,lxChariot等工具

3、各类硬件设备性能，一般采用专业的网络设备测试工具

4、各类服务器并发性能，分布式处理能力，可采用压力测试方案工具

5、业务系统性能，采用业务系统压力测试方案

6、数据库处理性能，这部分需要结合业务系统进行测试，以获取核心业务场景下的数据库的TPS/QPS，

7、如果有支付功能，需要进行支付渠道接口及分流测试，此部分相对而言可能是最大的瓶颈所在，此外还涉及备份方案，容灾方案，业务降级方案的测试。

**请对这个系统做出测试用例：一个系统，多个摄像头，抓拍车牌，识别车牌，上传网上，网上展示**

参考回答：

功能：

1.每个摄像头都能抓拍车牌；

2.每个摄像头抓拍到的车牌能正常交给系统处理；

3.系统能够正确识别车牌；

4.系统能够将识别出的车牌上传；

5.上传至网络的车牌能够正常展示出来；

一、功能测试

1.使用正常的车牌，保持车牌静止，检查每个摄像头是否能抓拍车牌；

2.使用类似非车牌的写有字的纸板，检查每个摄像头是否抓拍；

3.使用正常的车牌，保持车牌较高速移动，检查每个摄像头是否能抓拍车牌；

4.在多种情况下检查每个摄像头抓拍到的车牌能否正常交给系统处理，如临时断电、断网后能否正常将数据交给系统；

5.使用抓拍到的正常的车牌，交由系统处理，检查系统能否识别车牌；

6.使用非车牌的其他图片，交由系统处理，检查系统能否识别；

7.在多种情况下检查系统能否将正常识别出的车牌进行上传，如临时断电、断网后未上传数据是否能继续上传；

8.构造非车牌的其他内容的数据，检查系统能否将异常内容进行上传；

9.检查上传至网络的车牌能否正常展示出来；

10.上传非车牌的其他内容的数据，检查能否正常显示出来。

二、性能测试

1.同时向一个摄像头展示多个静止的车牌，检查摄像头能否抓拍到多个车牌；

2.同时向一个摄像头展示多个较高速运动的车牌，检查摄像头能否抓拍到多个车牌；

3.抓拍后，检查系统识别车牌的时间是否在需求要求的时间内；

4.模拟大量抓拍照片同时交由系统处理，检查一定压力下系统能否正常识别车牌；

5.模拟大量车牌同时上传，检查一定压力下能否上传成功。

三、安全性测试

1.检查是否能够通过给车牌加装饰物等方法，使摄像头无法抓拍或抓拍后系统无法正常识别车牌。

请你说一说PC网络故障，以及如何排除障碍

参考回答：

(1)首先是排除接触故障，即确保你的网线是可以正常使用的。然后禁用网卡后再启用，排除偶然故障。打开网络和共享中心窗口，单击窗口左上侧“更改适配器设置”右击其中的“本地连接“或”无线网络连接”，单击快捷菜单中的“禁用”命令，即可禁用所选网络。接下来重启网络，只需右击后单击启用即可。

(2)**使用ipconfig查看计算机的上网参数**

1、单击“开始|所有程序|附件|命令提示符“，打开命令提示符窗口

2、输入ipconfig，按Enter确认，可以看到机器的配置信息，输入ipconfig/all,可以看到IP地址和网卡物理地址等相关网络详细信息。

(3)**使用ping命令测试网络的连通性，定位故障范围**

在命令提示符窗口中输入”ping 127.0.0.1“，数据显示本机分别发送和接受了4个数据包，丢包率为零，可以判断本机网络协议工作正常，如显示”请求超时“，则表明本机网卡的安装或TCP/IP协议有问题，接下来就应该检查网卡和TCP/IP协议，卸载后重装即可。

(4)ping本机IP

在确认127.0.0.1地址能被ping通的情况下，继续使用ping命令测试本机的IP地址能否被ping通，如不能，说明本机的网卡驱动程序不正确，或者网卡与网线之间连接有故障，也有可能是本地的路由表面收到了破坏，此时应检查本机网卡的状态是否为已连接，网络参数是否设置正确，如果正确可是不能ping通，就应该重新安装网卡驱动程序。丢失率为零，可以判断网卡安装配置没有问题，工作正常。

(5)ping网关

网关地址能被ping通的话，表明本机网络连接以及正常，如果命令不成功，可能是网关设备自身存在问题，也可能是本机上网参数设置有误，检查网络参数。

**如何对淘宝搜索框进行测试**

参考回答：

一, 功能测试

输入关键字，查看: 返回结果是否准确，返回的文本长度需限制

1.1输入可查到结果的正常关键字、词、语句，检索到的内容、链接正确性；

1.2输入不可查到结果的关键字、词、语句；

1.3输入一些特殊的内容，如空、特殊符、标点符、极限值等，可引入等价类划分的方法等；

结果显示：标题，卖家，销售量，单行/多行，是否有图片

结果排序：价格 销量 评价 综合

4.返回结果庞大时，限制第一页的现实量，需支持翻页

多选项搜索：关键字 品牌 产地 价格区间 是否天猫 是否全国购

是否支持模糊搜索，支持通配符的查询

7, 网速慢的情况下的搜索

搜索结果为空的情况

未登录情况和登录情况下的搜索（登录情况下 存储用户搜索的关键字/搜索习惯）

二.性能测试：

1压力测试：在不同发用户数压力下的表现（评价指标如响应时间等）

2负载测试：看极限能承载多大的用户量同时正常使用

3稳定性测试：常规压力下能保持多久持续稳定运行

4内存测试：有无内存泄漏现象

5大数据量测试：如模拟从庞大的海量数据中搜索结果、或搜索出海量的结果后列示出来，看表现如何等等。

三. 易用性：交互界面的设计是否便于、易于使用

1依据不同的查询结果会有相关的人性化提示，查不到时告知？查到时统计条数并告知？有疑似输入条件错误时提示可能正确的输入项等等处理；

2查询出的结果罗列有序，如按点击率或其他排序规则，确保每次查询出的结果位置按规则列示方便定位，显示字体、字号、色彩便于识别等等

3标题查询、全文检索、模糊查询、容错查询、多关键字组织查询（空格间格开）等实用的检索方式是否正常？

4输入搜索条件的控件风格设计、位置摆放是否醒目便于使用者注意到，有否快照等快捷查看方式等人性化设计？

**四. 兼容性**

1**WINDOWS/LINUX/UNIX等各类操作系统下及各版本条件下的应用**

**2IE/FIREFOX/GOOGLE/360/QQ等各类浏览器下及各版本条件下**、各种显示分辨率条件下的应用

3SQL/ORACLE/DB2/MYSQL等各类数据库存储情况下的兼容性测试

4简体中文、繁体中文、英文等各类语种软件平台下的兼容性测试

**5IPHONE/IPAD、安卓等各类移动应用平台下的兼容性测试**

6与各相关的监控程序的兼容性测试，如输入法、杀毒、监控、防火墙等工具同时使用

五. 安全性

1被删除、加密、授权的数据，不允许被SQL注入等攻击方式查出来的，是否有安全控制设计；

2录入一些数据库查询的保留字符，如单引号、%等等，造成查询SQL拼接出的语句产生漏洞，如可以查出所有数据等等，这方面要有一些黑客攻击的思想并引入一些工具和技术，如爬网等。

1. **自动化测试执行时间，容错处理机制**
2. **Robot Framework 框架的优缺点**
3. **自动化测试下一步的有哪些可以提高的地方**
4. **Java的多态机制**
5. **平日工作是如何设计测试用例，讲解下目前所测产品的一个模块的测试用例设计策略**
6. **在整个项目周期中，测试要做哪些事情**