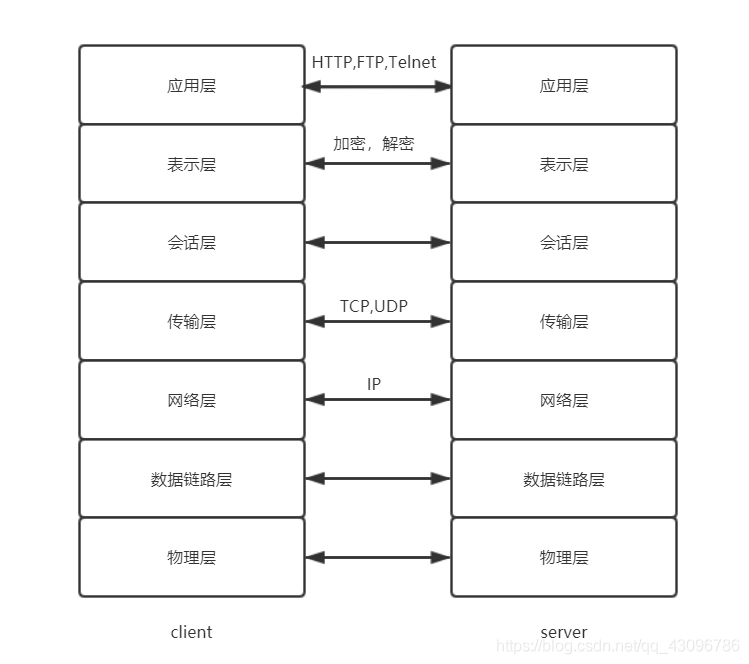
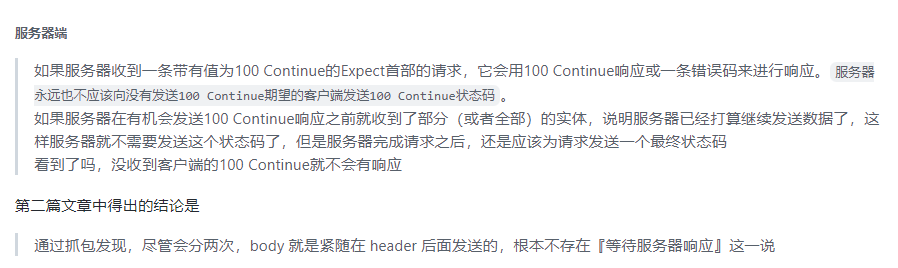
1. **接口测试面试题＜一＞**

[**https://blog.csdn.net/qq\_43096786/article/details/111768556?utm\_medium=distribute.pc\_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-16.control&depth\_1-utm\_source=distribute.pc\_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-16.control**](https://blog.csdn.net/qq_43096786/article/details/111768556?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-16.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-16.control)

**1.HTTP和HTTPS协议区别？**  
答：https协议需要到CA（Certificate Authority，证书颁发机构）申请证书，一般免费证书较少，因而需要一定费用；  
http是超文本传输协议，信息是明文传输，Https协议是由SSL+Http协议构建的可进行加密传输、身份认证的网络协议，比http协议安全；  
http和https使用的是完全不同的连接方式，用的端口也不一样，前者是80，后者是443；

**2.HTTPS在哪一层？**  
答：HTTPS在应用层。  
  
 ** 4.常见的POST提交数据方式**  
答：主要有四种方式：**application**/x-www-**form-urlencoded**、**multipart/form-data**、application/json、text/xml等。

**5.什么是Http协议无状态协议?怎么解决HTTP协议无状态协议?**  
答：无状态是指协议对于事务处理没有记忆能力，服务器不知道客户端是什么状态。即我们给服务器发送 HTTP 请求之后，服务器根据请求，会给我们发送数据过来，但是，发送完，不会记录任何信息。  
HTTP 是一个无状态协议，这意味着每个请求都是独立的，Keep-Alive 没能改变这个结果。缺少状态意味着如果后续处理需要前面的信息，则它必须重传，这样可能导致每次连接传送的数据量增大。另一方面，在服务器不需要先前信息时它的应答就较快。  
HTTP 协议这种特性有优点也有缺点，优点在于解放了服务器，每一次请求“点到为止”不会造成不必要连接占用，缺点在于每次请求会传输大量重复的内容信息。客户端与服务器进行动态交互的 Web 应用程序出现之后，HTTP 无状态的特性严重阻碍了这些应用程序的实现，毕竟交互是需要承前启后的，简单的购物车程序也要知道用户到底在之前选择了什么商品。于是，两种用于保持 HTTP 连接状态的技术就应运而生了，一个是 Cookie，而另一个则是 Session。

**6.cookie和session的区别**  
答：  
cookie数据存放在客户的浏览器上，session数据放在服务器上;  
cookie不是很安全，别人可以分析存放在本地的cookie并进行cookie欺骗，考虑到安全应当使用session;  
session会在一定时间内保存在服务器上。当访问增多，会比较占用你服务器的性能，考虑到减轻服务器性能方面应当使用cookie;  
单个cookie保存的数据不能超过4K，很多浏览器都限制一个站点最多保存20个cookie;  
可以将登陆信息等重要信息存放为session；其他信息需要保存，可以放在cookie.

**7.请求接口中常见的返回状态码**  
答：  
1xx – 信息提示（表示临时的响应。客户端在收到常规响应之前，准备接收一个或多个1xx响应）  
2xx – 成功（表明服务器成功地接受了客户端请求）  
3xx – 重定向（客户端浏览器必须采取更多操作来实现请求。例如，浏览器可能不得不请求服务器上的不同的页面，或通过代理服务器重复该请求）  
4xx – 客户端错误（发送错误，客户端有问题。例如，客户端请求不存在的页面，客户端未提供有效的身份证验证信息）  
5xx – 服务器错误（服务器由于遇到错误而不能完成该请求）

**8.什么是DNS？**  
答：DNS 是域名系统 (Domain Name System)，DNS是用来做域名解析的,它会在你上网输入网址后，把它转换成IP，然后去访问对方服务器；没有它，你想上百度就要记住百度的IP，但有了DNS的处理，你只需要记住对应网站的域名，即网址就可以了。

**9.请问你们公司是如何做接口测试的？**  
答：  
接口测试实际跟一般测试不同就是测试用例的设计部分。  
①获取接口规范。  
②设计接口测试功能用例（主要从用户角度出发看接口能否实现业务需求，用例设计就是黑盒用例那一套）。  
③各种入参验证（正常情况，异常情况包括输入参数个数不对，类型不对，可选/必选，还有考虑参数有互斥或关联的情况）。  
④接口返回值各种验证（符合接口文档需求）  
⑤了解接口实现逻辑，实现逻辑覆盖（语句/条件/分支/判定/…）  
⑥接口能并发执行吗、安全吗，性能满足要求吗?  
⑦采用工具或者自写代码来验证。  
⑧发现问题跟功能测试一样，该报bug报bug，该跟踪状态的跟踪状态。

**10.怎么设计接口测试用例？**  
答：  
通常，设计接口测试用例需要考虑以下几个方面：  
①是否满足前提条件  
有些接口需要满足前提，才可成功获取数据。常见的，需要登录Token  
逆向用例：针对是否满足前置条件（假设为n个条件），设计0~n条用例  
②是否携带默认值参数  
正向用例：带默认值的参数都不填写、不传参，必填参数都填写正确且存在的“常规”值，其他不填写，设计1条用例  
③业务规则、功能需求  
这里根据时间情况，结合接口参数说明，可能需要设计N条正向用例和逆向用例  
④参数是否必填  
逆向用例：针对每个必填参数，都设计1条参数值为空的逆向用例  
⑤参数之间是否存在关联  
有些参数彼此之间存在相互制约的关系  
⑥参数数据类型限制  
逆向用例：针对每个参数都设计1条参数值类型不符的逆向用例  
⑦参数数据类型自身的数据范围值限制  
正向用例：针对所有参数，设计1条每个参数的参数值在数据范围内为最大值的正向用例

**11.你做接口测试，测什么？**  
答：  
可用性测试  
根据约定的协议、方法、格式内容，传输数据到接口经处理后返回期望的结果：

接口功能是否正确实现；  
返回值测试 - 返回值除了内容要正确，类型也要正确，保证调用方能够正确地解析；  
参数值边界值、等价类测试；  
错误和异常处理测试

输入异常值（空值、特殊字符、超过约定长度等），接口能正确处理，且按预期响应；  
输入错误的参数，接口能正确处理，并按预期响应；  
多输入、少输入参数，接口能正确处理，且按预期响应；  
错误传输数据格式（如json格式写成form格式）测试；  
安全性测试，主要指传输数据的安全性：

敏感数据（如密码、秘钥）等是否加密传输；  
返回数据是否含有敏感数据，如用户密码、完整的用户银行账号信息等；  
接口是否对传入的数据做安全校验，如身份ID加token类似校验；  
接口是否防止恶意请求（如大量伪造请求接口致使服务器崩溃）；  
性能测试，如接口的响应时间、并发处理能力、压测处理情况：

并发请求相同的接口（特别为POST请求），接口的处理情况（如插入了相同的记录导致数据出错，引发系统故障）；  
接口响应时长在用户可忍受的范围内；  
对于请求量大的接口做压测，确定最大的瓶颈点是否满足当前业务需要；

**12.平常用什么工具测接口的？**  
答：常用http协议接口测试工具，如：postman、fiddler、jmeter；webService接口用SoapUI、jmeter等。

**13.没有接口文档，如果做接口测试？**  
答：用抓包工具把接口抓取处理，然后针对性进行测试；接口中字段信息不清楚的，找时间集中寻求开发解答。

**14.在手工接口测试或者自动化接口测试的过程中，上下游接口有数据依赖如何处理？**  
答：用一个全局变量来处理依赖的数据，比如登录后返回token，其它接口都需要这个token，那就用全局变量来传token参数。

**15.依赖于第三方数据的接口如何进行测试？**  
答：mock。接着面试官会问你，如果mock的，搭建mock服务，参考这篇 <http://www.51ste.com/share/det-485.html>

**16.接口测试中，依赖登录状态的接口如何测试？**  
答：依赖登录状态的接口的本质上是在每次发送请求时需要带上session或者cookie才能发送成功，在构建POST请求时添加必要的session或者cookie

**17.如何模拟弱网做测试？**  
答：Fiddler和charles都可以模拟弱网测试，平常说的模拟丢包，也是模拟弱网测试。具体可以看 <http://www.51ste.com/share/det-825.html>

**18.你平常做接口测试的过程中发现过哪些bug?**  
答：  
常规错误，接口没实现，没按约定返回结果，边界值处理出错等。  
**输入异常值**（空值、特殊字符、超过约定长度等），**接口抛错**，没做封装处理；  
**输入错误的参数**、多输入、少输入参数，接口可能出现的错误；  
安全性问题，如明文传输、返回结果含有敏感信息，没对用户身份信息做校验，没做恶意请求拦截等；  
**性能问题**，如接口并发插入多条相同操作，响应时间过长，接口压测出现瓶颈等；

**19.当一个接口出现异常时候，你是如何分析异常的？**  
答：先抓包，用fiddler（charles）工具抓包，或者浏览器上F12调试工具；APP上的话，那就用Fiddler做代理，通过手机设置代理去看请求和返回报文；  
查看后端日志，如Linux系统通过xhell连上服务器，查看接口日志，查看是否有报错信息（命令：tail -f 日志文件）

**20.如何分析一个bug是前端还是后端的？**  
答：这种情况很容易判断，先抓包看请求报文，对着接口文档，看请求报文有没问题，有问题就是前端发的数据不对；  
请求报文没问题，那就看返回报文，返回的数据不对，那就是后端开发的问题咯。

4、**公司有用到第三方服务**，出了问题，打电话给第三方，第三方不承认，这时候日志又显示不到错误，应该怎么处理？

1/ 单独调用第三方服务，查看三方接口返回的status，code，msg，看是否能支持正常服务

2/ 在自己的应用中mock第三方服务，来调用自己的接口，看能否正常工作，将结果反馈给第三方

3/  配置日志级别在debug级别时，添加第三方请求过程的日志，查看入参，响应来追踪问题

5、**接口自动化中的关联怎么处理**？

把上一个请求返回的结果传入到下一个请求的参数中，将请求的结果反射到一个类属性（使用setattr（）函数），下一个请求去调用这个类属性

6、自动化测试怎么校验结果？

断言 ，预期结果与实际结果对比

数据库校验，根据测试场景来查询数据库里的数据和请求之前的数据进行比对

7、自动化使用的测试框架是什么？简述自动化框架的设计、维护

1/测试框架：python+unittest+requests+ddt+openpyxl+pymysql+logging

测试框架：

python：入门简单，语法简洁

unittest ：定义一个测试用例类，具体的方法来维护测试用例的生命周期，测试场景行为，测试用例  前置场景，行为，期望结果，实际结果，断言方法，Setup teardown方法

requests：接口调用 ，支持http请求的库，API 简洁，提供不同的http请求方法，支持session，cookies，

 ddt ：数据驱动，ddt 类装饰器，data 测试方法装饰器  unpack解包可迭代的数据类型

普通用户，数据库，配置文件---（基础数据）

openpyxl：  数据管理 excel管理数据，使用openpyxl模块来进行excel数据的读和写（excle，csv,  json,  yaml,  txt都可以管理测试数据）

 pymysql：数据库交互，数据校验

 eval，json：数据格式的转换   Eval将python支持的格式转换成对应的格式

logging：日志处理， 统一日志输出格式，渠道，级别，执行结果的记录，便于定位问题

 jenkins：持续集成

2/框架设计思路：数据驱动+结构分层（可读性，可维护性，可扩展性）

数据驱动：将维护数据与代码分离，接口调用行为一致，针对不同的参数组合驱动不同的测试场景，减少代码冗余

结构分层：数据层+用例层+逻辑层

数据层：测试数据的支撑 data.xls

用例层：用例的执行 test\_register.py  test\_recharge.py

逻辑层：公用的方法的封装与提取 doexcle.py do\_mysql.py http\_requests.py logger.py等模块

3/框架设计步骤：

准备测试数据： EXCEL表准备测试用例---excel数据的读取---参数值的替换

发起请求：请求方法（get/post方法进行封装---URL的拼接（不同---参数转化为字典

拿到请求的返回值：解析返回值code，status，msg信息

断言

  好处：

1、自动化测试用例和手工测试用例的完美结合，减少重复工作

2、配置灵活，可以自主切换测试环境，执行测试用例

3、常用功能进行封装，逻辑清晰，易于维护

4、统一执行入口，管理测试用例集：

run.py模块通过模糊查找来选择需要执行的测试用例

5、持续集成，定时构建，快速反馈

8、具体的在这个项目中自动化怎么应用到实际的，您对自动化结果的分析

完成所有的自动化测试框架的设计和实现后，进行接口测试，然后集成到jenkins，配置定时执行，生成html报表，查看测试通过率，查看接口的功能

每次发版时，进行回归测试，新功能开发未提测前

9、做好测试计划的关键是什么？

明确测试目的，增强测试计划的实用性，保证测试用例的实用性和覆盖率，完全需求文档和规格指标书等，严格界定测试周期，测试轮次，测试覆盖模块等

2.测试的数据你放在哪？ -**数据与脚本分离**  
3.参数化 - **数据驱动模式**, @pytest.mark.parametrize  
4.下个接口请求参数依赖上个接口的返回数据 - 参数关联  
5.依赖于登录的接口如何处理 -**token和session的管理**  
6.依赖第三方的接口如何处理 -mock模拟数据返回  
7.不可逆的操作，如何处理，比如删除一个订单这种接口如何测试 -**造数据**  
8.接口产生的垃圾数据如何清理 - **数据清理**  
9.一个订单的几种状态如何全部测到，如：未处理，处理中，处理失败，处理成功 - **造数据，改数据库订单状态**  
10.python如何连接数据库操作？  
11.其它的就是运行出报告、代码管理（git）、运行策略和持续集成jenkins相关了

首先python里面的基础数据类型有：int、str、 float、list、bool、tuple、dict、set这几种类型，里面没json这种数据类型。

JSON(JavaScript Object Notation, JS 对象简谱) 是一种轻量级的数据交换格式。它基于 ECMAScript (欧洲计算机协会制定的js规范)的一个子集，采用完全独立于编程语言的文本格式来存储和表示数据。简洁和清晰的层次结构使得 JSON 成为理想的数据交换语言。 易于人阅读和编写，同时也易于机器解析和生成，并有效地提升网络传输效率。

由于你的代码是python写的（也有可能是php,java,c，ruby等语言）,但是后端接口是java写的（也有可能是其它语言），**不同的语言数据类型是不一样的**, 所以就导致你提交的数据，别的开发语言无法识别，这就需要规范传输的数据（传输的数据都是一个字符串），大家都遵循一个规范，按一个标准的格式去传输，于是就有就json这种国际化规范的数据类型。

**json本质上还是字符串**，只是按key：value这种键值对的格式来的字符串



# 2.测试的数据你放在哪?

First, depends on what kind of data;

**首先测试的数据是分很多种的**，有登录的账户数据，也有注册的账户数据，还有接口的参数，还有邮箱配置的数据等等等等，**所以这个题不能一概而论给答死了。要不然就是给自己挖坑**。

* Configuration data: log-in, token, pytest config, host name

Pytest.ini, config.ini,

以下两个大忌不能回答：

* 测试的数据是不能写死到代码里面的，这个是原则问题，也是写代码的大忌（你要是回答写在代码里面，估计就是回去等通知了）
* 测试数据放到.py的开头，这种其实很方便，对于少量的，固定不变的数据其实是可以放的，但是面试时候，千万不能这样说，面试官喜欢装逼的方法

测试数据存放总结：

1.对于账号密码，这种管全局的参数，可以用命令行参数，单独抽出来，写的配置文件里（如**ini**）  
2.对于一些一次性消耗的数据，比如注册，每次注册不一样的数，可以用**随机函数生成**  
3.对于一个接口有**多组测试的参数**，可以参数化，数据放yaml,text,json,excel都可以  
4.对于可以**反复使用的数据**，比如订单的各种状态需要造数据的情况，可以**放到数据库**，**每次数据初始化，用完后再清理**  
5.对于邮箱配置的一些参数，可以用ini配置文件  
6.对于**全部是独立的接口项目**，可以用数据驱动方式，用excel/csv管理测试的接口数据  
7.对于少量的静态数据，比如一个接口的测试数据，也就2-3组，可以写到py脚本的开头，十年八年都不会变更的

总之不同的测试数据，可以用不同的文件管理

# 3.什么是数据驱动，如何参数化？

参数化和数据驱动的概念这个肯定要知道的，参数化的思想是代码用例写好了后，不需要改代码，**只需维护测试数据**就可以了，并且根据不同的测试数据生成多个用例  
python里面用unittest框架







# 4.下个接口请求参数依赖上个接口的返回数据

这个很容易，不同的接口封装成不同的函数或方法，需要的数据return出来，用一个中间变量a去接受，后面的接口传a就可以了  
参考这篇[【python接口自动化26-参数关联和JSESSIONID(上个接口返回数据作为下个接口请求参数)】](https://www.cnblogs.com/yoyoketang/p/9671952.html)

# 5.依赖于登录的接口如何处理

登录接口依赖**token**的，可以先登录后，**token存到一个yaml或者json**，或者**ini**的配置文件里面，后面所有的请求去拿这个数据就可以全局使用了  
参考之前分享的一篇[python接口自动化24-有token的接口项目使用unittest框架设计](https://www.cnblogs.com/yoyoketang/p/9264773.html)

如果是cookies的参数，可以用session自动关联

s=requests.session()

后面请求用s.get()和s.post()就可以自动关联cookies了

# 6.依赖第三方的接口如何处理

这个需要自己去搭建一个mock服务，模拟接口返回数据，参考【python笔记25-mock-server之moco】(<https://www.cnblogs.com/yoyoketang/p/9348552.html>)

moco是一个开源的框架，在github上可以下载到https://github.com/dreamhead/moco

moco服务搭建需要自己能够熟练掌握，面试会问你具体如何搭建 ，如何模拟返回的数据，是用的什么格式，如何请求的

# 7.不可逆的操作，如何处理，比如删除一个订单这种接口如何测试

此题考的是**造数据的能力**，接口的请求数据，很多都是需要依赖前面一个状态的  
比如工作流这种，流向不同的人状态不一样，操作权限不一样，测试的时候，每种状态都要测到,就需要自己会造数据了。  
**平常手工测试造数据，直接在数据库改字段状态**。那么自动化也是一样，造数据可以用python连数据库了，做增删改查的操作  
测试用例前置操作，setUp做数据准备  
后置操作，tearDown做数据清理

# 8.接口产生的垃圾数据如何清理

跟上面一样，造数据和数据清理，需用python连数据库了，做增删改查的操作  
测试用例前置操作，setUp做数据准备

# 9.一个订单的几种状态如何全部测到，如：未处理，处理中，处理失败，处理成功

跟上面一样，也是考察造数据，修改数据的状态

# 10.python如何连接数据库操作？

这个就是详细的考察你是如何用python连数据库的，并且最好能现场写代码那种（有的笔试题就是python连数据库）  
具体问你用到哪个模块，查询的数据是什么类型？如何删除数据？如何新增数据？如何修改数据？  
PyMySQL 是在 Python3.x 版本中用于连接 MySQL 服务器的一个库，Python2中则使用mysqldb。  
详情参考教程http://www.runoob.com/python3/python3-mysql.html



其它的就是运行出报告、代码管理（git）、运行策略和持续集成jenkins相关了，这个所以的自动化但是一样的，后面会单独讲一篇jenkins持续集成相关