**HTTP：全拼HyperText Transfer Protocol**

# HTTP状态码(背)

HTTP状态码，表示Web服务器的响应状态，用来告诉客户端，发生了什么事。

http状态返回代码 1xx（临时响应）表示临时响应并需要请求者继续执行操作的状态代码。

**100 （继续）: 请求者应当继续提出请求。 服务器返回此代码表示已收到请求的第一部分，正在等待其余部分。**

http状态返回代码 2xx （成功）表示成功处理了请求的状态代码。

**200 （成功）: 服务器已成功处理了请求。 通常，这表示服务器提供了请求的网页。**

**http状态返回代码 3xx （重定向）**

表示要完成请求，需要进一步操作。 通常，这些状态代码用来重定向。

**300 （多种选择）: 针对请求，服务器可执行多种操作。 服务器可根据请求者 (user agent) 选择一项操作，或提供操作列表供请求者选择。**

**http状态返回代码 4xx（请求错误）**

这些状态代码表示请求可能出错，妨碍了服务器的处理。

**400 （错误请求）: 服务器不理解请求的语法。**

403 （禁止）: 服务器拒绝请求。

**404 （未找到）: 服务器找不到请求的网页。**

410 （已删除）: 如果请求的资源已永久删除，服务器就会返回此响应。

413 （请求实体过大）: 服务器无法处理请求，因为请求实体过大，超出服务器的处理能力。

414 （请求的 URI 过长）: 请求的URI（通常为网址）过长，服务器无法处理。

http状态返回代码 5xx（服务器错误）

这些状态代码表示服务器在尝试处理请求时发生内部错误。 这些错误可能是服务器本身的错误，而不是请求出错。

**500 （服务器内部错误）: 服务器遇到错误，无法完成请求。**

502 （错误网关）: 服务器作为网关或代理，从上游服务器收到无效响应。

505 （HTTP 版本不受支持）: 服务器不支持请求中所用的 HTTP 协议版本。

一些常见的http状态返回代码为：

【总结\*\*\*】

**1XX- 信息，服务器收到请求，需要请求者继续执行操作**

**2XX- 成功，操作被成功接收并处理**

**3XX- 重定向，需要进一步的操作以完成请求**

**4XX- 客户端错误，请求包含语法错误或无法完成请求**

**5XX- 服务器错误，服务器在处理请求的过程中发生了错误**

# http与https的区别(背)

http超文本传输协议 默认端口是80

http协议的特点：协议简单 通信速度快 灵活 允许传输任意类型的数据

http协议的缺点：通信使用明文，请求和响应不会对通信方进行确认，无法保护数据的完整性

http+ssl协议构成的比http协议更加安全

https的特点：  
1.采用混合加密技术，中间者无法查看明文内容  
2验证身份，保护数据完整性

**协议类型：**

**http协议**：超文本传输协议（协议简单，通信速度快） 端口：80（默认隐藏）

**SSL协议**：安全协议（用来保护参数）

**https协议**：加密的超文本传输协议 端口：443

**http协议+SSL协议=https协议** 也可以认为是http的加密加强版

**http请求中的类型：**

1. **GET** 向特定资源发出请求来获取它
2. **POST** 向指定资源提交数据进行处理请求 可以在火狐后台里面去看，后期可以使用fiddler查看
3. **PUT** 向指定资源位置上传最新内容
4. **Head** 服务端接收到请求头后只返回响应头，而不是发送响应内容。
5. **Delete** 请求服务器删除内容
6. **Trace** 回显服务器收到请求，返回时带的协议，主要测试和诊断
7. **Connect** http/1.1协议中能够将 / 连接改为 | 管道方式的代理服务器
8. **Options** 真正请求之前的一次预请求

# **get与post的区别（背）**

**请求方式一定是根据接口文档来进行的，不是自己乱定义的**是http请求的两种基本方法，主要出现在笔试和面试的时候  
使用fiddler 查看get post post下面有段乱码，就是加密的展示  
桌面上有提交用户基础文档，可以查看什么是接口，里面有post请求

**GET（查询、搜索、读取）：**

1. 是从服务器上获取数据，请求的数据会附在url后，以？分割  
   <https://123.sogou.com/?211149>
2. 具有长度限制，传递的数据量小4kb左右（不同的浏览器有差异）
3. Get方式通过url请求传递用户需要的参数，内容显示在浏览器地址栏中，安全性低，如果用户含有敏感信息密码之类的很容易被获取到  
   4.速度快，通常用于安全性不高的请求

**POST（交互：如订单、支付、登录）：**

1. 请求参数在request body里面 是在请求体里面的
2. 传递数据量大，没有提交限制
3. Post提交时不会将用户信息显示在浏览器的地址栏中，相对较安全

4.用于密码等安全性要求高的场合，提交数据量较大的场合（上传文件，发布文章）  
5.提交数据上限默认8m d tomcat conf  
server.xml 发现设置了maxPostSize最大规模=1024" 这个参数，只允许表单上传1m的数据  
12.Get和post的区别：

get（查询，搜索）post（支付，登录）

1.get和post首先都是属于TCP链接

2.get是通过url传递参数，post参数是放再请求体（body）里面

3.get请求在请求的时候有长度限制，post理论上是没有的

起限制的是服务器的处理能力

4.get请求相对post请求不是很安全，因为他的参数直接暴露在

url里面了，但是如果post请求的参数不加密也是不安全的。

5.对于get而言，浏览器会把http 请求头和请求数据一起发送出去

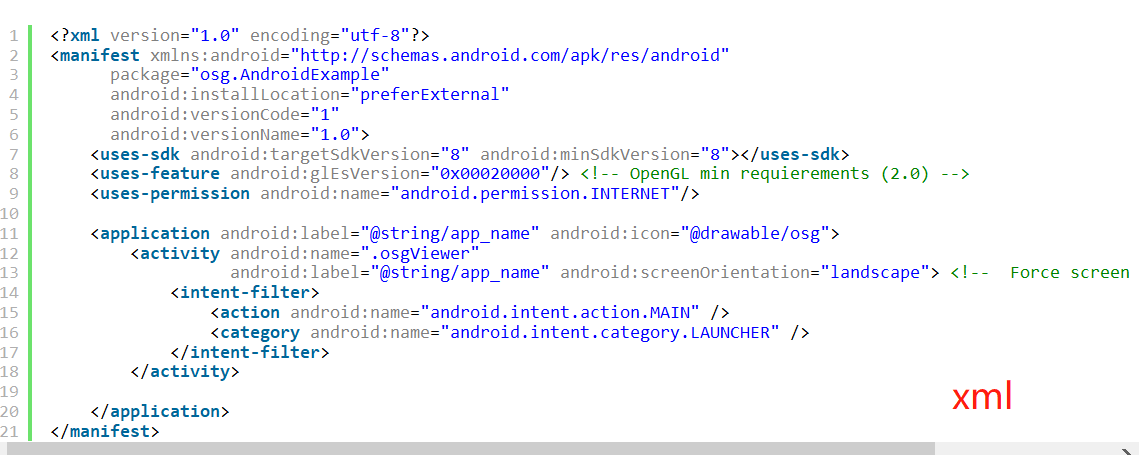
然后返回数据，而post是浏览器先发送请求头服务器响应100 然后

再发送请求数据

# 开发语言类型

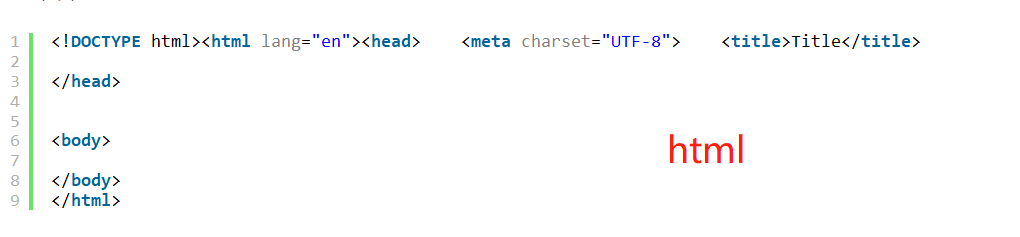
**Xml:**

可扩展标记性语言，是用来传输和存储数据，就像txt文本一样，它仅仅是纯文本而已，一般用于程序之间的数据传输



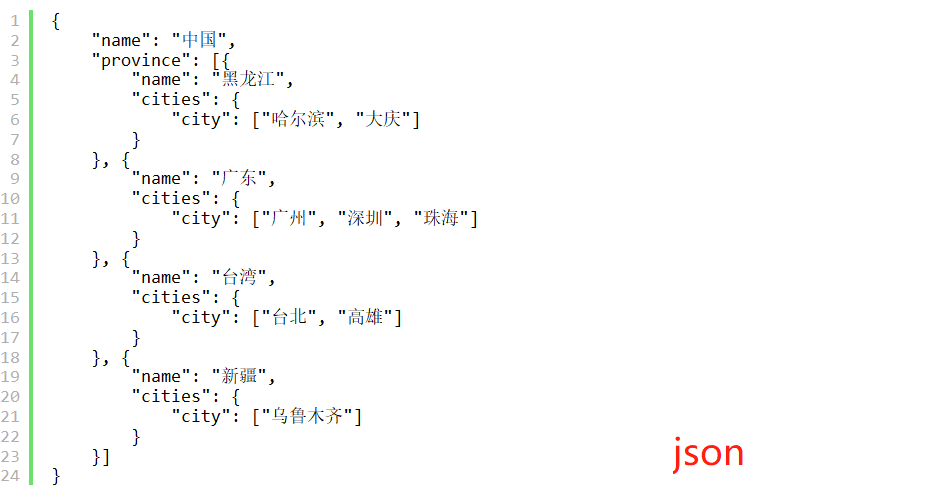
**html:**可以在火狐后台里面看一下

超文本标记语言，它是可以组合各种文字，图片，颜色等将其显示在网站上，以html开头，有body（身体） 和title(标题)



**{json}**

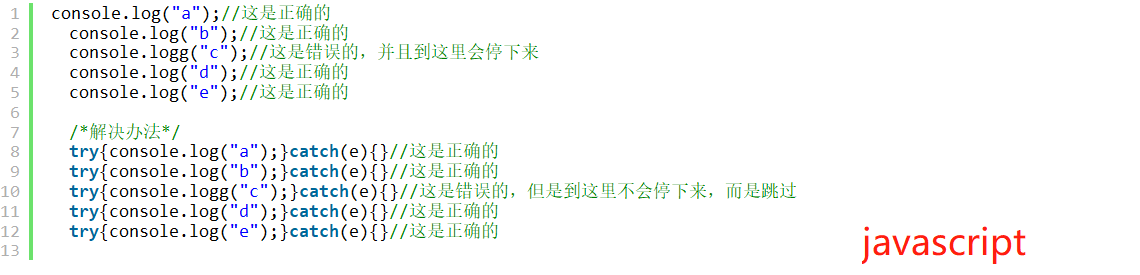
它是一种轻量级的数据交换格式。是以花括号形式展示，并显示为字段名，字段值，键对值的形式



**javascript:**

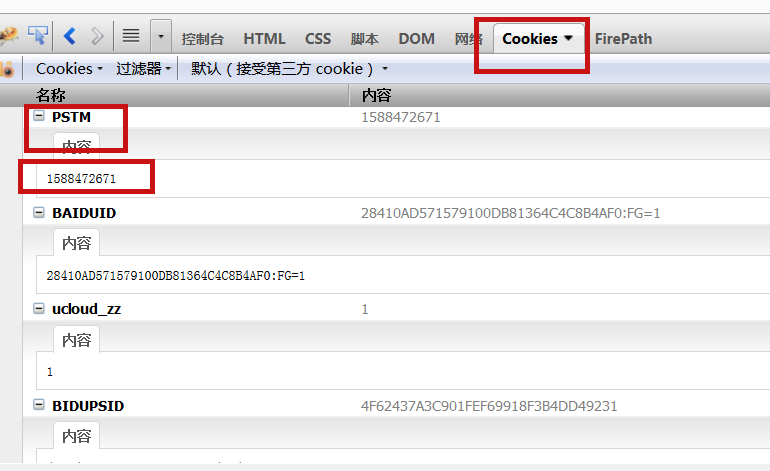
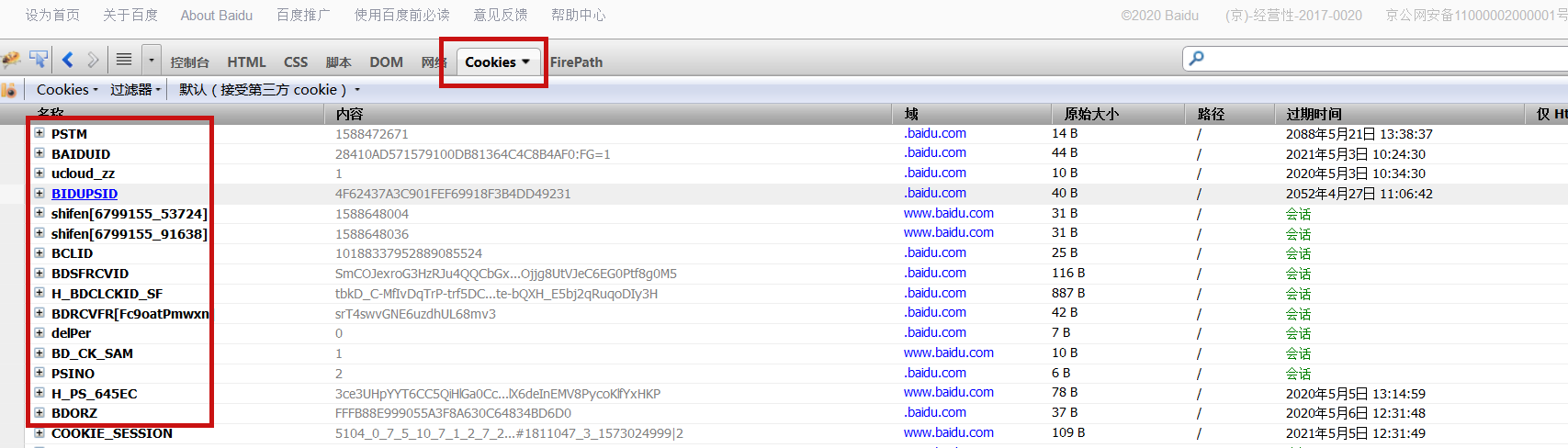
是一种前端开发页面时候用的一种脚本语言

(其源代码在发往客户端运行之前不需经过编译，而是将文本格式的字符代码发送给浏览器由浏览器解释运行)



# 什么是Cookie（通过cookie保存某网址浏览的历史记录）

1. 一个cookie只能包含一个key\_value信息
2. 一个cookie只能识别一个信息
3. 一个web站点可以给一个浏览器发送多个cookie
4. 浏览器一般只允许存放300个cookie
5. 每个站点（网址）最多保存20个cookie
6. 每个cookie的大小限制为不超过4kb 约80kb左右
7. Cookie是存放在浏览器上面的，不安全
8. Cookie有时效限制



**Cookie的作用：（存放在浏览器（客户端）中）**

1. 会话状态管理（浏览器发送请求给服务器，服务器把响应给用户，这个过程就是会话）

如：登录状态、购物车  
因为会保留你的登录状态，手机上的银行app一般不保留这种状态，也是为了安全，掌上生活就是这样  
也有的会保留你的密码，别人可以直接点击登录  
购物车就是你选择的商品会一直保留在你的购物车里面

1. 个性化设置（主题、皮肤）  
   自己选择的类型会根据不同的账号有不同的展示

浏览器行为跟踪（浏览记录的缓存）  
淘宝里面有个浏览记录，就是保留了用户的浏览记录  
大数据：现在是资源共享的时代，你在别的地方注册登录，查看自己喜好的电影美食等，对你规整出一个你自己的个人档案，当你登录其他客户端，大致也会推荐类似的东西  
Cookie的作用是让用户端（客户端）和服务器完成会话

# **session：（会话）**

会话是指我们访问网站的一个周期。  
打开一个站点：在这个站点点击多个超链接查看各个网页，然后关闭浏览器，整个过程称之为一个会话

1. 服务器用来辨别用户的唯一标识
2. 每个浏览器来访问服务器都会创建一个唯一标识（sessionid）

session这个标识不会重复（是由cookie+session=sessionid，也可以存储数据）

1. 基于底层cookie来实现  
     
   客户端浏览器访问网站的时候  
   服务器会向客户浏览器发送一个每个用户特有的会话编号sessionID，让他进入到cookie里  
   服务器同时也把sessionID和对应的用户信息、用户操作记录在服务器上，这些记录就是session  
   客户端浏览器再次访问时，会发送cookie给服务器，其中就包含sessionID  
   客户端浏览器再次访问时，会发送cookie给服务器，其中就包含session  
   服务器从cookie里找到sessionID，再根据sessionID找到以前记录的用户信息就可以知道他之前操控些、访问过哪里

# Cookie和session的区别：

1. cookie的数据是存放在客户端的浏览器上，session数据是存放在服务器上的
2. cookie不是很安全，别人可以分析本地cookie进行一个**cookie欺骗**

cookie欺骗类似于打断点改用户的登录信息

1. session会在一定时间内存放在服务器上，当访问增多的时候，会占用服务器的资源
2. 一般情况下cookie的大小不超过4kb，很多浏览器限制一个web节点最多保存20个 cookie，一个浏览器最多保存300个cookie
3. 一般情况下登录信息的账号和密码等隐私信息或重要信息存放在session里面其他信息保留可以放在cookie里

**Token：（令牌/出入证）一般应用于手机端（app）**

**组成**：（uid）用户，（time）时间戳，（sign）签名

基于token的验证方式，在服务器端不需要存储用户的登录记录；

首先客户端使用用户名和密码进行请求登录，服务器收到请求之后去验证用户名和密码，验证成功后，服务端会签发一个token，再把这个token传给客户端。客户端收到token后，把它存储起来，比如说可以存放在cookie里面客户端。用户每次向服务器请求的时候都要带着服务器签发的token服务器收到请求之后，然后去验证token是否正确，如果正确就返回数据，不正确就提示重新登录。同时服务器也会验证token有效期。  
就好比新职员报道，没有工牌保安不让进，你告诉他第一次上班有公司名字和地址，然后公司给你工牌，以后再来自己带着工牌  
时间戳：一个能表示一份数据在某个特定时间之前已经存在的、 完整的、 可验证的数据,通常是一个字符序列,唯一地标识某一刻的时间

# 无法找到该网页的原因分析：

1.没有网（检查网线，检查网路是否正常）

2.浏览器是否授权使用网络

3.浏览器版本与网页不兼容

4.浏览器有故障

5.浏览器设置了代理

6.Nginx出现问题

7.DNS出现问题

8.网站证书出现问题

9.检查防火墙是否打开

10.网址本身就存在问题

11.电脑设置了相关的host

12.请求中数据包丢失

13.本地资源占用过多

14.网速过慢，连接超时

15.电脑有病毒

16.服务器是否拒绝请求

17.服务器不存在请求的网页（被永久删除或转移到其他网页）

18.服务器不支持当前的协议版本

19.服务器出现拥堵/崩溃

20..停服

21.服务器作为网关或代理，从上游服务器收到无效响应

# OSI七层模型

包含：应用层 表示层 会话层 传输层 网络层 数据链路层 物理层

**TCP/IP模型**

包含：应用层 传输层 网际层 网络接口层

**功能：**

7.应用层：为应用软件提供很多服务，比如文件服务，数据库服务  
应用程序进程，电子邮件，文件传输和终端，就跟我们看到的这个显示层事差不多的

6.表示层：用于处理在两个通信系统中交换信息的表示方法  
封装数据，加密，格式化  
解封装的时候：  
微信聊天会发表情包，手机版本低，展示不出来，解密不出来这个表情，只能看见文字描述

5.会话层：负责维护两个节点之间的传输连接，以便确保点到点传输不中断  
建立了应用程序之间的一个会话，用一个软件实现了屏幕共享，qq好友聊天，你手机里面的qq和她手机里面的qq  
建立了qq和qq之间的会话  
这个会话的建立就是为了防止，qq发出去的信息使用微信收到了，肯定是不行的

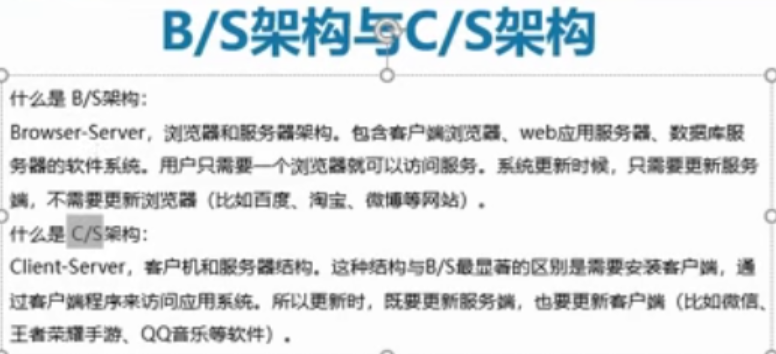
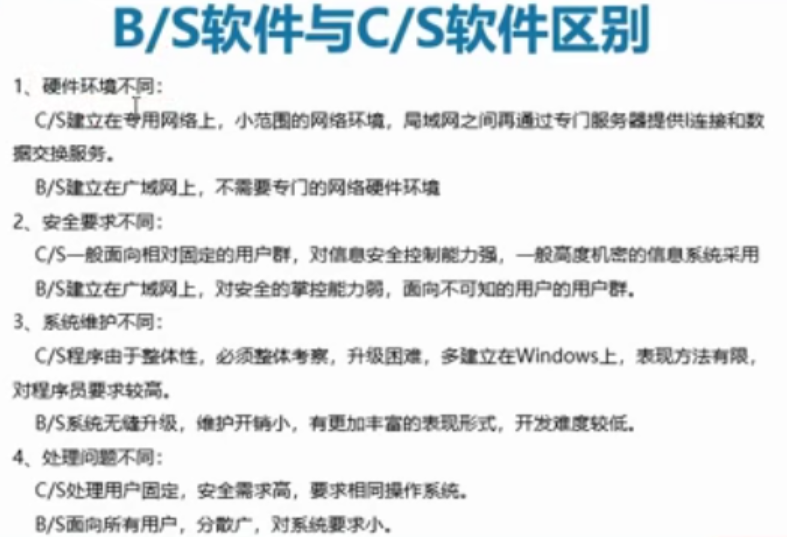
4.传输层：向客户提供可靠地端到端的服务，处理数据包的错误  
传输问题，端到端的问题  
我现在使用教师机来讲解问题，给你们群发信息，你们是不是也有的教师机的资料  
这就实现了端到端的，就是两台设备的连接  
tcp udp 就在这里 一个实现数据的正确和完整，一个不保证

3.网络层：为数据在节点之间传输逻辑链路层，以及实现拥堵控制  
路由和寻址：路由就是我找个一条路，我要这个路这么走才能到达目的地，寻址就是寻找IP地址，  
用来数据传输，支持逻辑寻址和路径选择  
是一个地图，高德，我现在在哪现在要去哪里，不知道路就开始导航  
路就是路由出来的，有不同车型的推荐路线

2.数据链路层：在物理层提供服务的基础上  
我这个数据是以什么样的形式在我底层来进行传输的，下面选的网线那就是网线  
支持错误检测  
我们关注234层最多

1.物理层：网络通信的数据传输介质，负责处理数据传输并监制数据出错率  
电脑连接了一个网线，这个数据经过网线转出去的时候就是通过了物理层，定义了传输介质：网线还是光纤

只要是通过浏览器打开的网页就是b/s架构  
需要安装客户端的就是c/s

# BS/CS的区别

B/S(浏览器/服务器)browser/server B/S就是通过浏览器

C/S(客户机/服务器)client/server C/S则需要客户端程序

比较特殊的还有F/S(文件/服务器)file/server

C/S是Borland最早研发，

B/S是微软研发。

应用C/S架构应用服务器运行数据负荷较轻,数据的储存管理功能较为透明,但是同时维护的成本较高。

应用B/S架构维护和升级都比较方便简单些,但是很明显服务器的数据负荷较高了.

c/s也叫胖客户结构

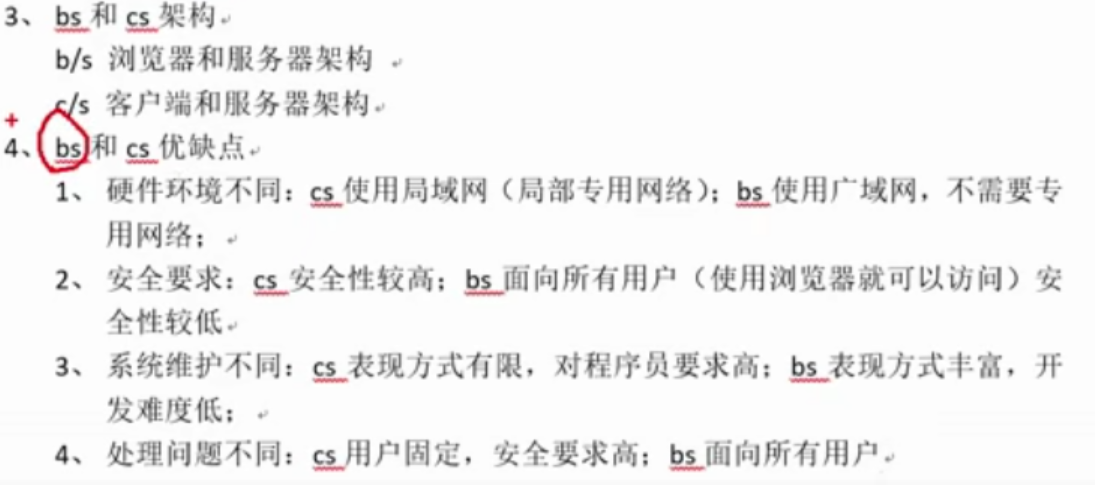
b/s叫痩客户结构

**BS/CS的区别**

bs是浏览器(browser)和服务器(server) cs是静态客户端程序(client)和服务器(server)

区别在于,虽然同样是通过一个程序连接到服务器进行网络通讯,但是bs结构的,客户端运行在浏览器里,比如你看百度,就是通过浏览器.还有一些bs结构的应用,比如中国电信,以及一些电子商务平台.用bs结构的好处是,不必专门开发一个客户端界面,可用asp,php,jsp等比较快速开发web应用的程序开发。

cs结构的,要做一个客户端.网络游戏基本上大多是cs结构,比如你玩传奇,要专门开个传奇程序;玩冒险岛,要专门开个冒险岛...... cs结构的优点是可以定做很多外观,可以做很多安全措施  
,可以补充浏览器没有的功能.缺点是开发速度比较慢,一个功能比较完善的客户端比较难做。



# APP测试与web测试的区别（背）

我首先是简单的回答了下，答案如下：

**A：**相同===>都是采用功能测试

**B：**不相同====>一个在web测试，一个在APP测试。

但是仅仅这样回答，你自己不满意，可能面试官也不会满意，那么我们就升级下这个答案。

**A：相同点**

不管是传统行业的web测试，还是新兴的手机app测试，都离不开测试的基础知识：

**1）同样的设计测试用例方法**：边界值分析法、等价类划分、错误推测法、场景法等

**2）同样的测试方法：**黑盒测试，验证业务功能是否正确符合用户或者设计预期；

**3）都要检查UI：**界面的布局、风格和按钮等是否简洁美观、是否统一等；

**4）页面性能检测：**测试页面载入和翻页的速度、登录时长、内存是否溢出等；

**5）应用的稳定性：**测试应用系统的稳定性等，不会闪退卡死等。

**B：不同点**

相对于web测试，APP测试，除了要考虑基本的功能测试、性能等，还要考虑手机本身固有的属性特征。所以APP测试过程中还需要注意如下几个方面特性：

**1）**手机作为通信工具，来电、去电、接收短信等操作都会对app应用程序产生影响，所以app测试第一个要考虑的属性特征是：**中断测试。**

中断测试有人为中断、新任务中断以及意外中断等几种情况，主要从以下几个方面进行验证：  
a.来电中断：呼叫挂断、被呼叫挂断、通话挂断、通话被挂断  
b.短信中断：接收短信、查看短信  
c.其他中断：蓝牙、闹钟、插拔数据线、手机锁定、手机断电、手机问题（系统死机、重启）

**2）**手机用户对app产品的安装卸载操作：

a.从上一个版本/上两个版本直接升级到最新版本。  
b.全新安装新版本  
c.新版本覆盖旧版本安装  
d.卸载旧版本，安装新版本  
e.卸载新版本，安装新版本

**3）**web自动化测试使用的工具较常用的是QTP，而android手机自动化测试工具比较常用的是monkey、monkeyrunner、appium。

# APP测试关注点（背）

## 1.适配性测试：

1>.手机不同分辨率支持：客户端支持的分辨率等

2>.手机不同版本的支持：2.34.04.4等；在测试计划中：需要安排单独的时间用于android不同系统的兼容性测试，包括2.0以下版本和4.0以上等

3>.手机不同厂家系统的支持：不同厂家会有不同android系统，例如：小米，华为，锤子对市面上主流手机的支持

4>.手机不同尺寸的支持：3.5到5.0屏幕在UI显示有区别，要支持最大到最小。

## 2.网络测试：

1.外网测试主要模拟客户使用网络环境，检验客户单程序在实际网络环境中使用情况及进行业务操作。

2.外网测试主要覆盖到wifi\2G\3G\4G,.net\wap、电信\移动\联通、所有可能的组合进行测试，在进行考虑这些情况下的弱网测试。

***弱网测试***

手机的网络目前主要分为2G、3G、wifi。目前2G的网络相对于比较慢，测试时尤其要注意此块的测试。

1) 无网络时，执行需要网络的操作，给予友好提示，确保程序不出现crash。

2) 内网测试时，要注意选择到外网操作时的异常情况处理。

3) 在网络信号不好时，检查功能状态是否正常，确保不因提交数据失败而造成crash。

4) 在网络信号不好时，检查数据是否会一直处于提交中的状态，有无超时限制。如遇数据交换失败时要给予提示。

5) 在网络信号不好时，执行操作后，在回调没有完成的情况下，退出本页面或者执行其他操作的情况，有无异常情况。此问题也会经常出现程序crash。

## 3、权限测试

1）软件权限：其中包括发送信息，拨打电话，链接网络，访问手机信息，联系人信息等等

2）数据在本地的存储、传输等

3）执行某些操作时导致的输入有效性验证、授权、数据加密等方面

***安全测试***

App安全测试大概划分为以下几类：

1.检测App的用户授权级别，数据泄漏，非法授权访问等；

2.对App的输入有效性校验、认证、授权、敏感数据存储、数据加密等方面进行检测,以期发现潜在的安全问题；

## 4.、安装运行卸载测试

1）验证app能否正确安装运行卸载，以及操作过程和操作前后对系统资源的占有情况

2）安装运行卸载的提示，报告等

3）检查安装路径，文件是否合理，组件是否正确注册等

## 5、UI测试

1）用户界面（菜单、对话框、窗口）等布局，风格是否满足用户需求，文字位置，描述是否正确，界面美观程度，文字图片组合是否合理

2）用户友好性、人性化、便于操作等

## 6、功能测试

1）评审需求，多方面考虑，整理出内在外在以及非功能性的直接间接功能点，对比需求，提取测试点

2）根据常用的一些分析方法，等价类边界值判定表因果图场景法等方法，设计测试用例，对提取的功能点进行覆盖

3）测试各个阶段不断跟踪缺陷，做好用例的更新迭代和不断变更需求所带来的业务或者需求的错误

## 7、性能测试

1）极限测试：各种边界情况下验证app的响应能力

如：低电量、储存满。弱网等情况

2）响应能力测试：验证各种情况下不同操作能否满足用户响应需求

3）压力测试：反复长期操作下，系统该资源的使用情况

## 8、中断测试（干扰）

比如：前后台运行时来电话，短信，下载文件，听音乐看电影等不同情况下的表现

## 9、兼容测试

1）不同网络环境（WiFi、2G、3G、4G等）

2）各种设备品牌机型系统版本等兼容

苹果、安卓（不同品牌，不同安卓系统版本）等

主要测试内部和外部兼容性，包括：再不同平台，不同手机上的运行，如安卓端的首先会考虑安卓的系统版本如Android 5.0 ，Android6.0 ，Android7.0 然后还会考虑不同系统版本下的手机型号测试 如华为，小米，三星，vivo，oppo等安卓机型 然后ios平台的会首先考虑系统版本 ios8，ios9，iOS10 然后会考虑不同手机型号 如 iPhone6，iPhone6 plus，iPhone7，iPhone8，iPhone8plus，iPhoneSE 等手机 看软件再不同系统以及不同手机型号上面是否能够正常运行，页面和功能是否完整正常。

## 10、升级更新测试

每次app版本迭代更新时，配合不同网络环境，及不同更新权限（强制更新，不强制更新），进行下载、安装、更新、启动运行等测试

1）测试升级后的功能是否与需求说明一样

2）测试与升级模块相关的模块的功能是否与需求一致

3）升级安装意外情况的测试（如断电、重启）

4）升级界面的UI测试

## 11、支付测试

1）支付结果的确认，数据库查询

2）请求报文是否加密

3）不同场景的支付

金额足够、金额不足、重复支付、无网支付、弱网支付、同账号多平台一起支付、余额宝微信信用卡等多种支付方式、不同支付方式的组合、密码正确/错误、支付上限等情况

## 手势操作测试

1）手机开锁屏对运行中的App的影响

2）切换网络对运行中的App的影响

3）运行中的App前后台切换的影响

4）多个运行中的App的切换

5）App运行时关机

6）App运行时重启系统

7）App运行时充电

8）App运行时kill掉进程再打开

## 渠道包测试

渠道：就是根据app发布的不同渠道进行打包，比如发布到安卓市场的和发布到应用宝的都会有相应的渠道包，渠道包就是用来做统计的，比如我们项目中用的是[友盟统计](https://www.baidu.com/s?wd=%E5%8F%8B%E7%9B%9F%E7%BB%9F%E8%AE%A1&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1d9myf1rjFbP1bznW-hn16Y0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3ErH6dnHRdPjmvPHbvrHDdnWb4" \t "https://zhidao.baidu.com/question/_blank)，它可以统计用户从哪个平台下载了你们的app，从而更好的掌握用户的操作习惯。如果不需要统计那就打成一样得包就可以了。主要是下载安装看下功能是否正常，还有就是看下开发是否把渠道得名字写的正确。

【汇总】一般都包含 **安装卸载测试**，**渠道包测试**，功能测试，UI测试，**权限测试**，**弱网测试**，**中断测试，兼容性测试**，安全测试，数据更新测试，**推送消息测试，升级更新测试，**用户体验测试，**手势操作，前后台切换，**用户体验以及回归测试。

# APP和web如何测试兼容性测试

第一 什么是兼容性

   兼容性三个字，如果要转换成我们熟悉的语言，可以说为：适应性。

  1)浏览器兼容性：当你测试的网站，换了一个浏览器去打开时，网站是否能够适应这个浏览器，并能正常使用？

  2)APP兼容性：当你测试的APP，换了一个品牌的手机，换了一个像素不同的手机，换了一个手机系统，APP是否适应这个手机，并能正常使用。

3)系统兼容性：当你的安装软件，换了一个操作系统，（xp,win7,win8,win10,mac)，软件是否适应不同的操作系统，并能够正常使用。

第二 如何测试兼容性？

刚刚上面说到了三个兼容性，其中第一和第二是我们平时测试接触最多的，第三种大家可以自行去了解。

**A：浏览器兼容性**

**常用主流浏览器：**IE、Firefox、chrome、360游览器及其他浏览器。

**测试点：**

1.不同的浏览器软件和版本

2.带有不同插件的浏览器

3.不同的浏览器选项下的表现

4.不同的图像分辨率下的测试

5.不同的系统色彩下的测试

6.不同的系统字体大小下的测试

7.不同的网速下的测试

8.文字、超链接、表格、图片。

9.各个功能模块的测试**（最重要）**

**测试方法：**对比测试。

**APP兼容性测试**

**市场主流手机：**苹果机、安卓机以及特殊要求的其他品牌机。具体如要详细到手机的品牌、像素以及手机系统，可以跟一些数据统计网站进行参考。

**测试点:**

1：功能测试

2：业务测试

3：图片、页面跳转、表单、数据加载测试

4：web端数据同步测试

5：不同手机系统下的测试

6：不同像素下的功能测试

7：版本兼容性测试（同时多个版本存在的时候）

**测试方法：**

对比测试

# 小程序测试(背)

小程序测试要点：

运行机制：【冷启动】和【热启动】

冷启动：首次打开小程序或者手动清除之后的再次打开为冷启动

热启动：打开之后在5分钟内再次开发是热启动，超过5分钟微信会自动回收内存

小程序测试除了要考虑测试最基本的

功能，性能，接口，ui测试，以及业务流和数据流的测试外

小程序有本身的一些特点

（1）小程序分为三种类型：开发版，体验版，正式版

项目中一般准备三套环境 开发版针对开发人员去开发，体验版供测试使用

正式环境使用正式版

（2）小程序的一些限制

页面层级跳转不能超过10层

用户本地缓存不能超过10M

小程序代码表不能超过3M

而且小程序需要提交微信审核才能发布

在测试的时候小程序有一些专项测试比如

1.权限测试

（1）获取用户权限测试，比如用户头像，性别，昵称等信息

（2）已授权/未授权进入小程序

（3）同一微信号在不同手机端的权限测试

（4）重新授权测试

（5）小程序内部的一些权限比如VIP和普通用户权限

2.缓存测试

（1）不同小程序之间的数据是否正确

（2）同一微信号用不同手机登录时数据是否正确

（3）同一手机登录不同或者多个微信号数据是否正确

3.页面层级跳转测试

首先页面层级跳转不能超过10层

（1）上下级的进入和返回

（2）不同层级之间进入和返回是否有相应按键

（3）页面层级跳转加载速度测试

（4）列表和菜单的测试

4.兼容性测试

（1）手机操作系统测试

（2）手机不同型号测试

（3）不同微信版本测试

（4）不同屏幕尺寸测试

5.网络测试

6.更新测试

更新的2中机制

（1）未启动时更新（后台静默更新到最新版本）

（2）启动时更新（每次冷启动的时候会检查是否有最新的版本，如果有就异步下载

，新版本会在下次冷启动时候应用）

7.业务流和数据流的测试

# H5测试(背)

**H5app和原生app的区别：**

H5app又称为webapp

原生app又称为native app

现在很多app都是原生结合H5共同开发的。

原生app根据不同的平台需要分别开发。

H5测试除了最基本的功能，性能，接口，UI测试外

还要一些专项测试：

1.针对登录模块的测试

（1）登录app那么进入H5 的时候也应该是登录状态

（2）若app未登录那么应该触发单点登录系统去提示登录

（3）app的cookie或者session过期的时候是否提示登录

2.刷新、翻页返回测试

（1）是否可以正常刷新返回或者返回

（2）点击物理back键是否可以返回

（3）单次或者多次刷新是否正常

（4）翻页之后页面是否可以正常加载

3.屏幕旋转测试

4.网络测试

5.滑动定位焦点是否准确

6.兼容性测试（跟app兼容性是一样的）

7.链接分享测试

（1）链接分享时候的标题是否展示完整

（2）分享之后是否可以打开，用户信息是否正确

（3）链接是否可以多次分享

（4）分享之后的页面样式和格式是否美观

（5）分享之后的点赞或者评论数据是否正确

（6）不同系统不同app之间的链接分享测试