Frontend Interview keypoints

HTML

1.说一下http和https

https的SSL加密是在传输层实现的。

(1)http和https的基本概念

http: 超文本传输协议，是互联网上应用最为广泛的一种网络协议，是一个客户端和服务器端请求和应答的标准（TCP），用于从WWW服务器传输超文本到本地浏览器的传输协议，它可以使浏览器更加高效，使网络传输减少。

https: 是以安全为目标的HTTP通道，简单讲是HTTP的安全版，即HTTP下加入SSL层，HTTPS的安全基础是SSL，因此加密的详细内容就需要SSL。

https协议的主要作用是：建立一个信息安全通道，来确保数组的传输，确保网站的真实性。

(2)http和https的区别?

http传输的数据都是未加密的，也就是明文的，网景公司设置了SSL协议来对http协议传输的数据进行加密处理，简单来说https协议是由http和ssl协议构建的可进行加密传输和身份认证的网络协议，比http协议的安全性更高。 主要的区别如下：

Https协议需要ca证书，费用较高。

http是超文本传输协议，信息是明文传输，https则是具有安全性的ssl加密传输协议。

使用不同的链接方式，端口也不同，一般而言，http协议的端口为80，https的端口为443

http的连接很简单，是无状态的；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。

(3)https协议的工作原理

客户端在使用HTTPS方式与Web服务器通信时有以下几个步骤，如图所示。

客户使用https url访问服务器，则要求web 服务器建立ssl链接。

web服务器接收到客户端的请求之后，会将网站的证书（证书中包含了公钥），返回或者说传输给客户端。

客户端和web服务器端开始协商SSL链接的安全等级，也就是加密等级。

客户端浏览器通过双方协商一致的安全等级，建立会话密钥，然后通过网站的公钥来加密会话密钥，并传送给网站。

web服务器通过自己的私钥解密出会话密钥。

web服务器通过会话密钥加密与客户端之间的通信。

(4)https协议的优点

使用HTTPS协议可认证用户和服务器，确保数据发送到正确的客户机和服务器；

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，要比http协议安全，可防止数据在传输过程中不被窃取、改变，确保数据的完整性。

HTTPS是现行架构下最安全的解决方案，虽然不是绝对安全，但它大幅增加了中间人攻击的成本。

谷歌曾在2014年8月份调整搜索引擎算法，并称“比起同等HTTP网站，采用HTTPS加密的网站在搜索结果中的排名将会更高”。

(5)https协议的缺点

https握手阶段比较费时，会使页面加载时间延长50%，增加10%~20%的耗电。

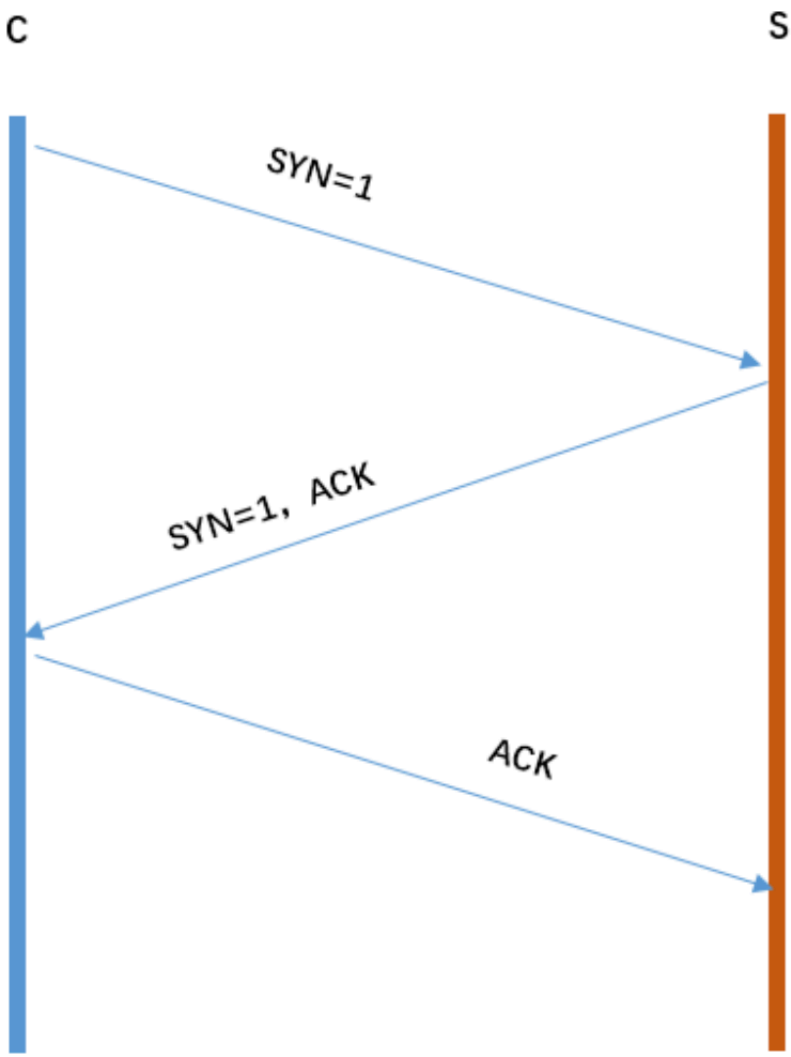
https缓存不如http高效，会增加数据开销。

SSL证书也需要钱，功能越强大的证书费用越高。

SSL证书需要绑定IP，不能再同一个ip上绑定多个域名，ipv4资源支持不了这种消耗。

2. tcp三次握手，一句话概括

客户端和服务端都需要直到各自可收发，因此需要三次握手。 简化三次握手：



从图片可以得到三次握手可以简化为：C发起请求连接S确认，也发起连接C确认我们再看看每次握手的作用：第一次握手：S只可以确认 自己可以接受C发送的报文段第二次握手：C可以确认 S收到了自己发送的报文段，并且可以确认 自己可以接受S发送的报文段第三次握手：S可以确认 C收到了自己发送的报文段

3. TCP和UDP的区别

（1）TCP是面向连接的，udp是无连接的即发送数据前不需要先建立链接。 （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。 并且因为tcp可靠，面向连接，不会丢失数据因此适合大数据量的交换。

（3）TCP是面向字节流，UDP面向报文，并且网络出现拥塞不会使得发送速率降低（因此会出现丢包，对实时的应用比如IP电话和视频会议等）。

（4）TCP只能是1对1的，UDP支持1对1,1对多。

（5）TCP的首部较大为20字节，而UDP只有8字节。

（6）TCP是面向连接的可靠性传输，而UDP是不可靠的。

4. WebSocket的实现和应用

(1)什么是WebSocket? WebSocket是HTML5中的协议，支持持久连续，http协议不支持持久性连接。Http1.0和HTTP1.1都不支持持久性的链接，HTTP1.1中的keep-alive，将多个http请求合并为1个

(2)WebSocket是什么样的协议，具体有什么优点?

HTTP的生命周期通过Request来界定，也就是Request一个Response，那么在Http1.0协议中，这次Http请求就结束了。在Http1.1中进行了改进，是的有一个connection：Keep-alive，也就是说，在一个Http连接中，可以发送多个Request，接收多个Response。但是必须记住，在Http中一个Request只能对应有一个Response，而且这个Response是被动的，不能主动发起。

WebSocket是基于Http协议的，或者说借用了Http协议来完成一部分握手，在握手阶段与Http是相同的。我们来看一个websocket握手协议的实现，基本是2个属性，upgrade，connection。

基本请求如下：

```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: x3JJHMbDL1EzLkh9GBhXDw==
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
Origin: http://example.com
```

多了下面2个属性：

```
Upgrade:webSocket
Connection:Upgrade
```

5. HTTP请求的方式，HEAD方式

head：类似于get请求，只不过返回的响应中没有具体的内容，用户获取报头 options：允许客户端查看服务器的性能，比如说服务器支持的请求方式等等。

6. 一个图片url访问后直接下载怎样实现?

请求的返回头里面，用于浏览器解析的重要参数就是OSS的API文档里面的返回http头，决定用户下载行为的参数。 下载的情况下：

x-oss-object-type	x-oss-request-id	x-oss-storage-class
Normal	598D5ED34F29D01FE2925F41	Standard

7. 说一下web Quality（无障碍）

能够被残障人士使用的网站才能称得上一个易用的（易访问的）网站。

残障人士指的是那些带有残疾或者身体不健康的用户。 使用alt属性。

有时候浏览器会无法显示图像。具体的原因有：

用户关闭了图像显示

浏览器是不支持图形显示的迷你浏览器

浏览器是语音浏览器（供盲人和弱视人群使用） 如果您使用了alt 属性，那么浏览器至少可以显示或读出有关图像的描述。

8. 几个很实用的BOM属性对象方法?

什么是Bom? Bom是浏览器对象。有哪些常用的Bom属性呢?

(1)location对象

```
location.href-- 返回或设置当前文档的URL

location.search -- 返回URL中的查询字符串部分。
例如 http://www.dreamdu.com/dreamdu.php?id=5&name=dreamdu
返回包括(?)后面的内容?id=5&name=dreamdu

location.hash -- 返回URL#后面的内容，如果没有#，返回空

location.host -- 返回URL中的域名部分，例如www.dreamdu.com

location.hostname -- 返回URL中的主域名部分，例如dreamdu.com

location.pathname -- 返回URL的域名后的部分。
例如 http://www.dreamdu.com/xhtml/ 返回/xhtml/

location.port -- 返回URL中的端口部分。
例如 http://www.dreamdu.com:8080/xhtml/ 返回8080

location.protocol -- 返回URL中的协议部分。
例如 http://www.dreamdu.com:8080/xhtml/ 返回(//)前面的内容http:

location.assign -- 设置当前文档的URL

location.replace() -- 设置当前文档的URL，并且在history对象的地址列表中移除这个URL

location.replace(url);

location.reload() -- 重载当前页面
```

(2)history对象

```
history.go() -- 前进或后退指定的页面数 history.go(num);
history.back() -- 后退一页
history.forward() -- 前进一页
```

(3)Navigator对象

```
navigator.userAgent -- 返回用户代理头的字符串表示(就是包括浏览器版本信息等的字符串)
navigator.cookieEnabled -- 返回浏览器是否支持(启用)cookie
```

9. 说一下HTML5 drag api

dragstart：事件主体是被拖放元素，在开始拖放被拖放元素时触发。

darg：事件主体是被拖放元素，在正在拖放被拖放元素时触发。

dragenter：事件主体是目标元素，在被拖放元素进入某元素时触发。

dragover：事件主体是目标元素，在被拖放在某元素内移动时触发。

dragleave：事件主体是目标元素，在被拖放元素移出目标元素是触发。

drop：事件主体是目标元素，在目标元素完全接受被拖放元素时触发。

dragend：事件主体是被拖放元素，在整个拖放操作结束时触发。

10. 说一下http2.0

首先补充一下，http和https的区别，相比于http,https是基于ssl加密的http协议 简要概括：http2.0是基于1999年发布的http1.0之后的首次更新。 提升访问速度（可以对于，请求资源所需时间更少，访问速度更快，相比http1.0）

允许多路复用：多路复用允许同时通过单一的HTTP/2连接发送多重请求-响应信息。改善了：在http1.1中，浏览器客户端在同一时间，针对同一域名下的请求有一定数量限制（连接数量），超过限制会被阻塞。

二进制分帧：HTTP2.0会将所有的传输信息分割为更小的信息或者帧，并对他们进行二进制编码

首部压缩

服务器端推送

http2.0的特性如下：

1、内容安全，应为http2.0是基于https的，天然具有安全特性，通过http2.0的特性可以避免单纯使用https的性能下降

2、二进制格式，http1.X的解析是基于文本的，http2.0将所有的传输信息分割为更小的消息和帧，并对他们采用二进制格式编码，基于二进制可以让协议有更多的扩展性，比如引入了帧来传输数据和指令

3、多路复用，这个功能相当于是长连接的增强，每个request请求可以随机的混杂在一起，接收方可以根据request的id将request再归属到各自不同的服务端请求里面，另外多路复用中也支持了流的优先级，允许客户端告诉服务器那些内容是更优先级的资源，可以优先传输，

11. 补充400和401、403状态码

(1)400状态码：请求无效 产生原因：

前端提交数据的字段名称和字段类型与后台的实体没有保持一致

前端提交到后台的数据应该是json字符串类型，但是前端没有将对象JSON.stringify转化成字符串。

解决方法：

对照字段的名称，保持一致性

将obj对象通过JSON.stringify实现序列化

(2)401状态码：当前请求需要用户验证

(3)403状态码：服务器已经得到请求，但是拒绝执行

12. fetch发送2次请求的原因

fetch发送post请求的时候，总是发送2次，第一次状态码是204，第二次才成功？ 原因很简单，因为你用fetch的post请求的时候，导致fetch 第一次发送了一个Options请求，询问服务器是否支持修改的请求头，如果服务器支持，则在第二次中发送真正的请求。

13. Cookie、sessionStorage、localStorage的区别

共同点：都是保存在浏览器端，并且是同源的

Cookie：cookie数据始终在同源的http请求中携带（即使不需要），即cookie在浏览器和服务器间来回传递。而sessionStorage和localStorage不会自动把数据发给服务器，仅在本地保存。cookie数据还有路径（path）的概念，可以限制cookie只属于某个路径下,存储的大小很小只有4K左右。 （key：可以在浏览器和服务器端来回传递，存储容量小，只有大约4K左右）

sessionStorage：仅在当前浏览器窗口关闭前有效，自然也就不可能持久保持，localStorage：始终有效，窗口或浏览器关闭也一直保存，因此用作持久数据；cookie只在设置的cookie过期时间之前一直有效，即使窗口或浏览器关闭。（key：本身就是一个回话过程，关闭浏览器后消失，session为一个回话，当页面不同即使是同一页面打开两次，也被视为同一次回话） localStorage：localStorage 在所有同源窗口中都是共享的；cookie也是在所有同源窗口中都是共享的。（key：同源窗口都会共享，并且不会失效，不管窗口或者浏览器关闭与否都会始终生效）

补充说明一下cookie的作用： 保存用户登录状态。例如将用户id存储于一个cookie内，这样当用户下次访问该页面时就不需要重新登录了，现在很多论坛和社区都提供这样的功能。 cookie还可以设置过期时间，当超过时间期限后，cookie就会自动消失。因此，系统往往可以提示用户保持登录状态的时间：常见选项有一个月、三个 月、一年等。

跟踪用户行为。例如一个天气预报网站，能够根据用户选择的地区显示当地的天气情况。如果每次都需要选择所在地是烦琐的，当利用了cookie后就会显得很人性化了，系统能够记住上一次访问的地区，当下次再打开该页面时，它就会自动显示上次用户所在地区的天气情况。因为一切都是在后 台完成，所以这样的页面就像为某个用户所定制的一样，使用起来非常方便定制页面。如果网站提供了换肤或更换布局的功能，那么可以使用cookie来记录用户的选项，例如：背景色、分辨率等。当用户下次访问时，仍然可以保存上一次访问的界面风格。

14. 说一下web worker

在HTML页面中，如果在执行脚本时，页面的状态是不可相应的，直到脚本执行完成后，页面才变成可相应。web worker是运行在后台的js，独立于其他脚本，不会影响页面你的性能。并且通过postMessage将结果回传到主线程。这样在进行复杂操作的时候，就不会阻塞主线程了。

如何创建web worker：

```
检测浏览器对于web worker的支持性

创建web worker文件（js，回传函数等）

创建web worker对象
```

15. 对HTML语义化标签的理解

HTML5语义化标签是指正确的标签包含了正确的内容，结构良好，便于阅读，比如nav表示导航条，类似的还有article、header、footer等等标签。

16. iframe是什么？有什么缺点?

定义：iframe元素会创建包含另一个文档的内联框架

提示：可以将提示文字放在

```
<iframe></iframe>
```

之间，来提示某些不支持iframe的浏览器

缺点：

```
1. 会阻塞主页面的 onload 事件

2. 搜索引擎无法解读这种页面，不利于SEO

3. iframe和主页面共享连接池，而浏览器对相同区域有限制所以会影响性能。
```

17. Doctype作用?严格模式与混杂模式如何区分？它们有何意义?

Doctype声明于文档最前面，告诉浏览器以何种方式来渲染页面，这里有两种模式，严格模式和混杂模式。

严格模式的排版和JS 运作模式是 以该浏览器支持的最高标准运行。

混杂模式，向后兼容，模拟老式浏览器，防止浏览器无法兼容页面。

18. Cookie如何防范XSS攻击

XSS（跨站脚本攻击）是指攻击者在返回的HTML中嵌入javascript脚本，为了减轻这些攻击，需要在HTTP头部配上，set-cookie： httponly-这个属性可以防止XSS,它会禁止javascript脚本来访问cookie。

secure - 这个属性告诉浏览器仅在请求为https的时候发送cookie。

结果应该是这样的：Set-Cookie=.....

19. Cookie和session的区别

HTTP是一个无状态协议，因此Cookie的最大的作用就是存储sessionId用来唯一标识用户

20. 一句话概括RESTFUL

就是用URL定位资源，用HTTP描述操作

21. click在ios上有300ms延迟，原因及如何解决?

(1)粗暴型，禁用缩放

(2)利用FastClick，其原理是：

检测到touchend事件后，立刻出发模拟click事件，并且把浏览器300毫秒之后真正出发的事件给阻断掉

22. addEventListener参数

addEventListener(event, function, useCapture)

其中，event指定事件名；function指定要事件触发时执行的函数；useCapture指定事件是否在捕获或冒泡阶段执行。

23. 介绍知道的http返回的状态码

状态码	HTTP	解释
100	Continue	继续。客户端应继续其请求
101	Switching Protocols	切换协议。服务器根据客户端的请求切换协议。只能切换到更高级的协议，例如，切换到HTTP的新版本协议
200	OK	请求成功。一般用于GET与POST请求
201	Created	已创建。成功请求并创建了新的资源
202	Accepted	已接受。已经接受请求，但未处理完成
305	Use Proxy	使用代理。所请求的资源必须通过代理访问
400	Bad Request	客户端请求的语法错误，服务器无法理解
401	Unauthorized	请求要求用户的身份认证
402	Payment Required	保留，将来使用
403	Forbidden	服务器理解请求客户端的请求，但是拒绝执行此请求
404	Not Found	服务器无法根据客户端的请求找到资源（网页）。通过此代码，网站设计人员可设置"您所请求的资源无法找到"的个性页面
501	Not Implemented	服务器不支持请求的功能，无法完成请求
502	Bad Gateway	作为网关或者代理工作的服务器尝试执行请求时，从远程服务器接收到了一个无效的响应

24. http常用请求头

协议头	说明
Accept	可接受的响应内容类型（Content-Types)。
Accept-Charset	可接受的字符集
Accept-Encoding	可接受的响应内容的编码方式。
Accept-Language	可接受的响应内容语言列表。
Authorization	用于表示HTTP协议中需要认证资源的认证信息
Cache-Control	用来指定当前的请求/回复中的，是否使用缓存机制。
Cookie	由之前服务器通过Set-Cookie（见下文）设置的一个HTTP协议Cookie
Content-Length	以8进制表示的请求体的长度
Origin	发起一个针对跨域资源共享的请求（该请求要求服务器在响应中加入一个Access-Control-Allow-Origin的消息头，表示访问控制所允许的来源）。
Via	告诉服务器，这个请求是由哪些代理发出的。
Warning	一个一般性的警告，表示在实体内容体中可能存在错误。

25. 强，协商缓存

缓存分为两种：强缓存和协商缓存，根据响应的header内容来决定。

	获取资源形式	状态码	发送请求到服务器
强缓存	从缓存取	200（from cache）	否，直接从缓存取
协商缓存	从缓存取	304（not modified	是，通过服务器来告知缓存是否可用

强缓存相关字段有expires，cache-control。如果cache-control与expires同时存在的话，cache-control的优先级高于expires。

协商缓存相关字段有Last-Modified/If-Modified-Since，Etag/If-None-Match

因为服务器上的资源不是一直固定不变的，大多数情况下它会更新，这个时候如果我们还访问本地缓存，那么对用户来说，那就相当于资源没有更新，用户看到的还是旧的资源；所以我们希望服务器上的资源更新了

浏览器就请求新的资源，没有更新就使用本地的缓存，以最大程度的减少因网络请求而产生的资源浪费。

26. 前端优化

降低请求量：合并资源，减少HTTP 请求数，minify / gzip 压缩，webP，lazyLoad。

加快请求速度：预解析DNS，减少域名数，并行加载，CDN 分发。

缓存：HTTP 协议缓存请求，离线缓存 manifest，离线数据缓存localStorage。

渲染：JS/CSS优化，加载顺序，服务端渲染，pipeline。

27. GET和POST的区别

get参数通过url传递，post放在request body中。 get请求在url中传递的参数是有长度限制的，而post没有。

get比post更不安全，因为参数直接暴露在url中，所以不能用来传递敏感信息。

get请求只能进行url编码，而post支持多种编码方式

get请求会浏览器主动cache，而post支持多种编码方式。

get请求参数会被完整保留在浏览历史记录里，而post中的参数不会被保留。

GET和POST本质上就是TCP链接，并无差别。但是由于HTTP的规定和浏览器/服务器的限制，导致他们在应用过程中体现出一些不同。

GET产生一个TCP数据包；POST产生两个TCP数据包。

28. 301和302的区别

301 Moved Permanently 被请求的资源已永久移动到新位置，并且将来任何对此资源的引用都应该使用本响应返回的若干个URI之一。如果可能，拥有链接编辑功能的客户端应当自动把请求的地址修改为从服务器反馈回来的地址。除非额外指定，否则这个响应也是可缓存的。

302 Found 请求的资源现在临时从不同的URI响应请求。由于这样的重定向是临时的，客户端应当继续向原有地址发送以后的请求。只有在Cache-Control或Expires中进行了指定的情况下，这个响应才是可缓存的。

字面上的区别就是301是永久重定向，而302是临时重定向。

301比较常用的场景是使用域名跳转。302用来做临时跳转 比如未登陆的用户访问用户中心重定向到登录页面。

29. HTTP支持的方法

GET, POST, HEAD, OPTIONS, PUT, DELETE, TRACE, CONNECT

30. 如何画一个三角形

三角形原理：边框的均分原理

```
    div {
    width:0px;
    height:0px;
    border-top:10px solid red;
    border-right:10px solid transparent;
    border-bottom:10px solid transparent;
    border-left:10px solid transparent;
    }
```

31. HTML5新增的元素

首先html5为了更好的实践web语义化，增加了header，footer，nav,aside,section等语义化标签，在表单方面，为了增强表单，为input增加了color，emial,data ,range等类型，在存储方面，提供了sessionStorage，localStorage,和离线存储，通过这些存储方式方便数据在客户端的存储和获取，在多媒体方面规定了音频和视频元素audio和vedio，另外还有地理定位，canvas画布，拖放，多线程编程的web worker和websocket协议

32. 在地址栏里输入一个URL,到这个页面呈现出来，中间会发生什么?

这是一个必考的面试问题。

输入url后，首先需要找到这个url域名的服务器ip,为了寻找这个ip，浏览器首先会寻找缓存，查看缓存中是否有记录，缓存的查找记录为：浏览器缓存-》系统缓存-》路由器缓存，缓存中没有则查找系统的hosts文件中是否有记录，如果没有则查询DNS服务器，得到服务器的ip地址后，浏览器根据这个ip以及相应的端口号，构造一个http请求，这个请求报文会包括这次请求的信息，主要是请求方法，请求说明和请求附带的数据，并将这个http请求封装在一个tcp包中，这个tcp包会依次经过传输层，网络层，数据链路层，物理层到达服务器，服务器解析这个请求来作出响应，返回相应的html给浏览器。

因为html是一个树形结构，浏览器根据这个html来构建DOM树，在dom树的构建过程中如果遇到JS脚本和外部JS连接，则会停止构建DOM树来执行和下载相应的代码，这会造成阻塞，这就是为什么推荐JS代码应该放在html代码的后面，之后根据外部央视，内部央视，内联样式构建一个CSS对象模型树CSSOM树，构建完成后和DOM树合并为渲染树，这里主要做的是排除非视觉节点，比如script，meta标签和排除display为none的节点，之后进行布局，布局主要是确定各个元素的位置和尺寸，之后是渲染页面，因为html文件中会含有图片，视频，音频等资源，在解析DOM的过程中，遇到这些都会进行并行下载，浏览器对每个域的并行下载数量有一定的限制，一般是4-6个，当然在这些所有的请求中我们还需要关注的就是缓存，缓存一般通过Cache-Control、Last-Modify、Expires等首部字段控制。

Cache-Control和Expires的区别在于Cache-Control使用相对时间，Expires使用的是基于服务器 端的绝对时间，因为存在时差问题，一般采用Cache-Control，在请求这些有设置了缓存的数据时，会先 查看是否过期，如果没有过期则直接使用本地缓存，过期则请求并在服务器校验文件是否修改，如果上一次 响应设置了

ETag值会在这次请求的时候作为If-None-Match的值交给服务器校验，如果一致，继续校验 Last-Modified，没有设置ETag则直接验证Last-Modified，再决定是否返回304。

```
DNS解析

TCP连接

发送HTTP请求

服务器处理请求并返回HTTP报文

浏览器解析渲染页面

连接结束
```

33. cache-control的值有哪些

cache-control是一个通用消息头字段被用于HTTP请求和响应中，通过指定指令来实现缓存机制，这个缓存指令是单向的，常见的取值有private、no-cache、max-age、must-revalidate等，默认为private。

34. 浏览器在生成页面的时候，会生成那两颗树?

构造两棵树，DOM树和CSSOM规则树。

当浏览器接收到服务器相应来的HTML文档后，会遍历文档节点，生成DOM树。

CSSOM规则树由浏览器解析CSS文件生成。

35.csrf和xss的网络攻击及防范

CSRF：跨站请求伪造，可以理解为攻击者盗用了用户的身份，以用户的名义发送了恶意请求，比如用户登录了一个网站后，立刻在另一个ｔａｂ页面访问量攻击者用来制造攻击的网站，这个网站要求访问刚刚登陆的网站，并发送了一个恶意请求，这时候CSRF就产生了，比如这个制造攻击的网站使用一张图片，但是这种图片的链接却是可以修改数据库的，这时候攻击者就可以以用户的名义操作这个数据库，防御方式的话：使用验证码，检查https头部的refer，使用token

XSS：跨站脚本攻击，是说攻击者通过注入恶意的脚本，在用户浏览网页的时候进行攻击，比如获取cookie，或者其他用户身份信息，可以分为存储型和反射型，存储型是攻击者输入一些数据并且存储到了数据库中，其他浏览者看到的时候进行攻击，反射型的话不存储在数据库中，往往表现为将攻击代码放在url地址的请求参数中，防御的话为cookie设置httpOnly属性，对用户的输入进行检查，进行特殊字符过滤

36. 怎么看网站的性能如何

检测页面加载时间一般有两种方式，一种是被动去测：就是在被检测的页面置入脚本或探针，当用户访问网

页时，探针自动采集数据并传回数据库进行分析，另一种主动监测的方式，即主动的搭建分布式受控环境，模拟用户发起页面访问请求，主动采集性能数据并分析，在检测的精准度上，专业的第三方工具效果更佳，比如说性能极客

37. 介绍HTTP协议(特征)

HTTP是一个基于TCP/IP通信协议来传递数据（HTML 文件, 图片文件, 查询结果等）HTTP是一个属于应用层的面向对象的协议，由于其简捷、快速的方式，适用于分布式超媒体信息系统。它于1990年提出，经过几年的使用与发展，得到不断地完善和扩展。目前在WWW中使用的是HTTP/1.0的第六版，HTTP/1.1的规范化工作正在进行之中，而且HTTP-NG(Next Generation of HTTP)的建议已经提出。HTTP协议工作于客户端-服务端架构为上。浏览器作为HTTP客户端通过URL向HTTP服务端即WEB服务器发送所有请求。Web服务器根据接收到的请求后，向客户端发送响应信息。

38. HTML5和CSS3用的多吗？你了解它们的新属性吗？有在项目中用过吗?

html5：

1）标签增删

8个语义元素 header section footer aside nav main article figure

内容元素mark高亮 progress进度

新的表单控件calander date time email url search

新的input类型 color date datetime datetime-local email

移除过时标签big font frame frameset

2）canvas绘图，支持内联SVG。支持MathML

3）多媒体audio video source embed track

4）本地离线存储，把需要离线存储在本地的文件列在一个manifest配置文件

5）web存储。localStorage、SessionStorage

css3：

CSS3边框如border-radius，box-shadow等；

CSS3背景如background-size，background-origin等；

CSS3 2D，3D转换如transform等；CSS3动画如animation等。