1. 什么是AJAX ，AJAX原理

什么是AJAX ：

AJAX全称为“Asynchronous JavaScript and XML”（异步JavaScript和XML），是一种创建交互式网页应用的网页开发技术。它使用：  
使用XHTML+CSS来标准化呈现；  
使用XML和XSLT进行数据交换及相关操作；  
使用XMLHttpRequest对象与Web服务器进行异步数据通信；   
使用Javascript操作Document Object Model进行动态显示及交互；   
使用JavaScript绑定和处理所有数据。

优点： 1.不会跳转页面 2.能进行局部刷新 3.提升用户体验 4.**支持异步处理 5. 减少服务器请求和网络负载**

**缺点：1.浏览器不兼容** -AJAX高度依赖Javascript，而不同的浏览器对Javascript支持性不同。这成 了一个问题，尤其是当AJAX必须跨许多浏览器工作的时候。那些不支持Javascript或者不支持Javascript某些选项的浏览器将不能够正常 使用ajax。由于ajax对Javascript的依赖性，它不适用移动应用。你的web浏览器的后退键不能如期运行。

**2. 不安全性** -网页可能很难调试，增加网页的代码量，你的网页更可能遇上严峻的安全威胁。

**3.增加Web服务器的负载** -如果你增加一个自动更新的功能，它每隔几秒向服务发起请求，那么就会增加服务器的负载

AJAX原理 ：

Ajax的工作原理相当于在用户和服务器之间加了—个中间层(AJAX引擎),使用户操作与服务器响应异步化。并不是所有的用户请求都提交给服务器,像—些数据验证和数据处理等都交给Ajax引擎自己来做, 只有确定需要从服务器读取新数据时再由Ajax引擎代为向服务器提交请求。  
  
Ajax其核心有JavaScript、XMLHTTPRequest、DOM对象组成，通过XmlHttpRequest对象来向服务器发异步请求，从服务器获得数据，然后用JavaScript来操作DOM而更新页面。这其中最关键的一步就是从服务器获得请求数据。让我们来了解这几个对象。

1. 使用异步加载获取JS数据至少两种方法？

异步加载又叫非阻塞加载，浏览器在下载执行js的同时，还会继续进行后续页面的处理。

<async>（<async>属性是HTML5中新增的异步支持）

<script type="text/javascript" async="true" src=” ”></script>

### Onload window.Onload =function(){}

1. 什么是JSON，JSON的格式

是一种轻量级的数据交换格式。写的像对象的字符串，本质是字符串

1. 除了AJAX还有什么模板可以与后台交互

JSONP Form表单提交 jquery中封装好的ajax

1. AJAX中的跨域问题：什么是跨域？如何解决跨域问题？

****域名，协议，端口均 有一个不同的就叫跨域****

1. ****JSONP：**** 但是要注意JSONP只支持GET请求
2. ****代理****
3. ****让后台给权限****
4. AJAX的流程是什么？

1.客户端产生js的事件  
2.创建XMLHttpRequest对象  
3.对XMLHttpRequest进行配置  
4.通过AJAX引擎发送异步请求  
5.[服务器端](https://www.baidu.com/s?wd=%E6%9C%8D%E5%8A%A1%E5%99%A8%E7%AB%AF&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)接收请求并且处理请求，返回html或者xml内容  
6.XML调用一个callback()处理响应回来的内容  
7.页面局部刷新

1. AJAX如何调用JSON数据

$.ajax({

url : 地址

success ： 成功函数

erorr : 失败的函数

data : 传送的数据

async : 同步异步默认 true

type : 传送数据类型，默认get

dataType : 后台返回的数据类型 xml json jsonp

})

1. AJAX能够处理哪些格式的文件

json后缀 php xml

1. JSON数据的解析方法
2. JSON.parse() 只能转化标准格式的JSON;key值必须用双引号包裹
3. eval() eval('('+data+')') >1. 会消耗更多的性能 >2. 安全性低，会携带病毒
4. (new Function('','return'+data))()
5. 说明异步请求的get和post方法的区别

get传送方式的特点：

1. 传送的数据在浏览器的缓存机制里面可以看见，所有安全性比较低

2. 传送给后台的数据只能放在地址栏

3. 传给后台的数据大小有限制，最多2000字节

4. 在低版本的ie浏览器里面，传送中文会报错，为了防止报错，必须把中文转码

post传送的特点：

1. 相对get来说，安全性高

2. post的数据是放在send里面发送的

3. 理论上post传送的数据没有大小限制

4. 必须设置请求头

ajax.setRequestHeader('Content-type','application/x-www-form-urlencoded');

1. 在ie低版本浏览器里面，中文汉字不需要在转码
2. 清除浮动的方法

父级div定义 height  结尾处加空div标签 clear:both （清除浮动） 父级div定义 伪类:after 和 zoom 父级div定义 overflow:hidden  父级div 也一起浮动

1. JQuery对象和Dom对象相互转换

jq对象变成原生对象：

1. 通过下标把jq对象转换为原生对象

2. get(下标)

jq对象转成原生对象以后，是不能在使用jq里面的方法

把原生对象转成jq对象：

$(原生元素) = > jq对象

1. 什么是HTML5？新增属性有哪些？

HTML5 是对 HTML 标准的第五次修订

border-radius box-shadow border-image background-size background-origin

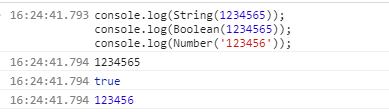
 text-shadow @keyframes

14. zepto与jquery框架有何区别？

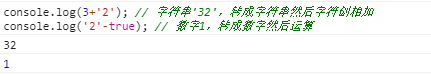
一个pc 一个移动端

15.例举至少 3 种强制类型转换和 2 种隐式类型转换?

String()，Boolean()，Number()；



种隐式类型转换： + -



16.JavaScript 的事件流模型都有什么

事件流分为两个过程：

1. 捕获 ： 当一个元素的事件被触发，浏览器会

沿着window一层一层往下走，一直到找到目标元素为止，

这个过程我们叫做： 捕获过程或者捕获阶段

2. 冒泡 ： 当目标元素的事件执行完，在沿着祖先元素一层一层往上走

，一直到window为止，这个阶段我们称为：冒泡阶段

17.BOM 对象有哪些，列举 window 对象？

1、window对象 ，是JS的最顶层对象，其他的BOM对象都是window对象的属性；

2、document对象，文档对象；

3、location对象，浏览器当前URL信息；

4、navigator对象，浏览器本身信息；

5、screen对象，客户端屏幕信息；

6、history对象，浏览器访问历史信息；

18 .HTTP 状态消息 200 302 304 403 404 500 分别表示什么？

200 ：（OK）请求已成功，请求所希望的响应头或数据体将随此响应返回。出现此状态码是表示正常状态。

302 ：请求的资源临时从不同的 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。

304：如果客户端发送了一个带条件的 GET 请求且该请求已被允许，而文档的内容（自上次访问以来或者根据请求的条件）并没有改变，则服务器应当返回这个状态码。304响应禁止包含消息体，因此始终以消息头后的第一个空行结尾。

403：服务器已经理解请求，但是拒绝执行它。与401响应不同的是，身份验证并不能提供任何帮助，而且这个请求也不应该被重复提交。如果这不是一个 HEAD 请求，而且服务器希望能够讲清楚为何请求不能被执行，那么就应该在实体内描述拒绝的原因。当然服务器也可以返回一个404响应，假如它不希望让客户端获得任何信息。

404：请求失败，请求所希望得到的资源未被在服务器上发现。没有信息能够告诉用户这个状况到底是暂时的还是永久的。假如服务器知道情况的话，应当使用410状态码来告知旧资源因为某些内部的配置机制问题，已经永久的不可用，而且没有任何可以跳转的地址。404这个状态码被广泛应用于当服务器不想揭示到底为何请求被拒绝或者没有其他适合的响应可用的情况下。出现这个错误的最有可能的原因是服务器端没有这个页面。

500：服务器遇到了一个未曾预料的状况，导致了它无法完成对请求的处理。一般来说，这个问题都会在服务器端的源代码出现错误时出现。

19.new 操作符具体干了什么呢?

1. 创建一个空对象
2. 设置原型链
3. 让function中的this指向obj，并执行function的函数体。
4. 判断Func的返回值类型：
5. 事件默认行为的阻止?

cancelBubble : 阻止冒泡,兼容所有浏览器, 但不是w3c的标准

false : 有冒泡机制

true : 没有冒泡机制

stopPropagation() : 阻止冒泡的，是w3c的标准但是 低版本ie浏览器不兼容