大方面试题

主要方向，做GPS定位，大数据展示图和VR

1. js中如何检测一个变量是一个String类型？请写函数实现。

Typeof(obj) ===’string’

Typeof obj === ‘string’

Obj.constructor === String

1. 减低页面加载时间的方法

2.1重复的HTTP请求应尽量减少（减少调用其他页面、文件的数量 背景图片加载缩减为一个）

2.2压缩js和css代码

2.3文件头部放置css样式定义（对于用户端的慢速网络或者网页比较庞大的情况下比较有利，可在网页逐步呈现时保持格式信息，不影响网速美感）

2.4文件末尾放置javascript脚本（网页加载是从上到下加载，如果放到头部，可能导致载入速度过慢或无法载入，放置到末尾，防止影响到主体网页加载速度）

2.5css 、 javascript改由外部调用，浏览器会对css和js进行缓存

2.6尽可能减少DCOM元素，<>减少数量

2.7避免使用css脚本，css动态参数改变，使用js实现

2.8添加文件过期或缓存头，服务器负载降低

2.9CDN网络加速（将图片视频扩散到CDN网络所到之处，用户网络访问时就近下载文件，网络提速目的，减轻自己网站负载）

2.10服务器启用zgip压缩功能（传输文件压缩后输出到客户端解压，网络传输数据会减少，会增加服务器负载要求服务器性能

2.11ajax采取缓存调用

2.12ajax采取get方法调用，js原生post会发生两次请求，get一次请求，get请求HTTP请求减少50%

2.13良好的编程习惯，开发维护习惯，避免脚本重复出现

2.14减缩iframe使用，无必要不要使用

2.15合理使用flush

2.16避免采用301、302的转向

2.17优化图片文件

2.18页面庞大到一定地步可采取分页

2.19使用多域名负载网页内的多个文件，图片

2.20配置ETags

1. 清除字符串中的全部空格用什么方法？

方法1：使用replace正则匹配方法

var str=" a b c d ";

console.log(str);

// 去除所有字符串的所有空格

console.log(str.replace(/\s\*/g,""));

// 去除字符串的两头空格

console.log(str.replace(/^\s\*|\s\*$/g,""))

// 去除字符串前空格

console.log(str.replace(/^\s\*/g,""))

// 去除字符串的后空格

console.log(str.replace(/(\s\*$)/g,""))

方法2：使用trim（）方法，局限，无法去除中间空格，jquery一样

// 只能去除中间空格

console.log(str.trim());

1. 前端数据的存储方法？存储大小？

localStorage 本地存储，H5新方法，IE8以上都兼容

生命周期，持续化的本地存储，除非主动删除数据，否则数据永远不会过期

存储信息在同一域中共享

本页操作了localStorage时（增，删，改），本页面不会触发storage事件，但别的页面会触发storage事件

大小：5M，和浏览器厂商有关

在非IE下的浏览器可在本地打开，IE需在服务器下打开

localStorage本质上是对字符串的读取，存储内容过多会消耗内存空间，导致页面变卡

localStorage受同源策略的影响

设置 localStorage.setItem(‘key’,’value’)

获取 localStorage.getItem(‘key’)

删除 localStorage.remove(‘key’)

一次清除所有缓存 localStorage.clear()

sessionStorage 和localStorage差不多，本地存储，回话本地存储

用于本地存储会话中的数据，在同一个会话中的页面才能访问并且回话结束数据随之销毁。关闭窗口，sessionStorage被销毁，新窗口中打开同源的另一个页面，sessionStorage也没有。

Cookie是纯文本无可执行代码，存储数据，访问某个网站的时候，通过cookie向访问者电脑上存储数据，或者某些网站辨别用户身份，进行session跟踪而存储在本地终端上的数据（加密）

不同浏览器存放cookie位置不一样不可以通用

Cookie的存储形式是以域名进行区分的，不同域下存储的cookie是独立的

可以设置cookie生效的区域，可以操作cookie是当前域及其当前域下的所有子域

一个域名下存放的cookie的个数是有限制的，不同浏览器下存放个数不一样，一般为20个

每个cookie存放的内容大小也是有限制的，不同浏览器下存放大小不一样，一般为4kb

Cookie可以设置过期时间，默认回话结束的时候，当时间到期自动销毁

三种前端存储的同异：

相同：在本地存储数据（浏览器端）

不同：localStorage在相同协议、相同主机名、相同端口下读取、修改到同一份localStorage数据。

sessionStorage比localStorage苛刻一点，除协议，主机，端口，还要求在同一窗口内（浏览器的标签页）

localStorage 永久存储，除非主动删除

sessionStorage 当回话结束自动销毁（页面关闭）

Cookie数据在每一次发送http请求时同时发送给服务器，而local和session不会。

1. Eval是做什么用？

Eval() 函数会计算某个字符串，并执行javaScript代码，只接受原始字符串为参数，

1. Npm执行命令？说出三个

Npm随nodejs安装的包管理工具，可以解决NodeJS代码部署上的许多问题

允许客户从npm服务器下载别人编写的第三方包到本地使用

允许客户从npm服务器下载别人编写的命令行程序到本地会用

允许客户将自己编写的第三方包或命令行程序上传到npm服务器供别人下载使用

Npm -v 检测是否安装成功

Npm confiig list/ls 显示配置项信息

Npm config list/ls -l 更详细

Npm -h 帮助显示信息，建议多查看

1. 程序的输出结果？

Let [foo,[[bar],baz]]=[a,[[2],3]] bar=2

Let [,,third]=[‘foo’,’bar’,’baz’] Third=（报错，Third无法识别）

Let [x,y,...z]=[‘a’] z=（输出空数组[]）

1. Mvvm框架的理解，mvvm的典型框架，与MVC框架的区别？

Mvc把信息展现逻辑和用户 交互分离的计算机用户界面开发模式；model包含应用数据和业务逻辑；controller负责将用户输入转换为命令传递给model和view

Model管理应用数据，model数据发生改变时，通知监听者（view），监听者做出相应变化

View当前model的视觉展示，view会观察model变化，当模型改变时会被通知，允许view更新自己，一般会使用模板引擎渲染model

Controler是models和view之间的调解人，当model改变时更新view和操作view是更新model

Mvc好处：易于维护

模型视图的解耦，意味着可以对业务逻辑进行更好的单元测试

代码可以更好的重用

模块化的开发可是分工更加明确，一部分人专注业务逻辑，一部分人专注用户界面

Mvvm（model view viewModel），为mvc衍生物，mvvm模式支持事件驱动的UI开发平台

比较成熟的框架：KnockoutJS、kendo mvvm和knockbackJS

Model关注数据信息，不关心任何行为，不格式化数据或影响数据在浏览器中的展示，格式化数据是view层的任务，业务逻辑封装在viewmodle中，和model进行交互。

View只应用中直接和用户交互的部分，Mvvm的主动式的view包含数据绑定，事件和需要理解的model和viewmodel行为，行为和属性对应，view响应viewmodel事件，view不负责控制状态。

Viewmodel 专门用于数据转化的controller，将model信息转换为view中信息，view专递命令给model。viewModel位于UI层后，暴露数据给view，view层的数据和行为的源

Mvvm使并行开发更加容易，前端开发人员和后端开发人员互不影响

抽象化的view，减少了代码中业务逻辑

Viewmodel比事件驱动更容易测试

Viewmodel的测试不关心ui的自动化和交互

缺点：简单ui，mvvm有点太重

声明时数据绑定不利于调试，命令式代码可设置断点

不挑剔的应用中数据可绑定和创建大量薄本

大应用中，获取大量概要前很难设计视图-模型层

1. 如果实现Pc端的网页聊天需要使用什么样的技术，列出两种实现方法？

基于ajax技术实现，基于comet（pushlet）技术实现，基于xmpp协议实现，基于flash的XMlsoket和远程共享对象实现

1. 基于ajx技术实现：ajax（异步的javascript和xml，asynchronous javascript和xml），作用是实现页面与服务器端的无刷新交互。页面上每隔一段时间从服务器获取数据更新页面显示。缺点实时性不高。
2. 基于coment技术实现：conmet是一种新的web应用架构，应用中服务端会以异步的形式向客户端推送数据，不需要客户端显示的发出请求，comet架构适合事件驱动的web应用，交互性和实时性较高的应用。Pushlet是一种comet实现（开源comet框架）servlet机制下数据从服务器的java对象直接推送到客户端页面，可使sever端以周期性更新client的web页面。
3. 基于XMPP协议实现：XMPP（可扩展的消息处理现场协议）基于XML协议，转为及时通信设计的通信协议，用于及时消息及在线现场探测，促进服务器间的准及时操作。
4. Canvas绘制一条直线，写出关键步骤

var mycanvas=document.getElementById("mycanvas");

var ctx=mycanvas.getContext("2d");

ctx.moveTo(20,20); //起始点位置

ctx.lineTo(50,100); //结束点位置

ctx.stroke(); //绘制

1. $(document).ready()方法和window.onload区别？

页面加载完成执行的函数，.ready()是DOM结构绘制完毕就执行，不必等到加载完毕，多个ready。.load是页面所有元素都加载完毕，包含图片等所有元素，只能执行一次。.read()执行时间早于load()函数。

执行时间，执行个数，简化写法

1. 简述qq一键登录的实现思想及其用到的技术？
2. 考试系统需要实现的内容及其功能？
3. Js和css选择器？

js

常见选择器：标签选择器，类选择器，id选择器，通配符选择器

层选择器

过滤选择器

属性过滤器

Css

.class 选择器

#id选择器

\*选择器

Element选择器

Element，element选择器

：link选择区

：after选择器等

1. Css的伪元素？

Css伪元素向某些选择器设置特殊效果

：first-line伪元素向文本首行设置特殊样式，只能块级元素

：first-letterr 向文本的首字母设置特殊样式，只能块级元素

：before向元素的内容前插入新元素

：after向元素内容后插入新元素

1. Js改变属性值？

setAttribute设置属性

getAttribute获取属性

Jquery获取或设置

Attr（）

1. 原生js的ajax的实现步骤及其传输什么对象？

17.1创建XMLHTTPRequest对象

17.2使用open方法设置和服务器的交互信息

17.3发送数据，和服务器交互

17.4注册事件

17.5更新界面

GET请求：

// 步骤一:创建异步对象

var ajax=new XLMHttpRequest();

// 步骤二:设置请求的url参数,参数一为请求类型,参数二为请求的url,可带参数,动态传递参数starName到服务端

ajax.open('get','getStar.php?getStartName='+name);

// 步骤三:发送请求

ajax.send();

// 步骤四:注册事件oreadystatechange状态改变就会调用

ajax.onreadystatechange=function(){

if(ajax.readyState==4&&ajax.status=200){

// 步骤五:如果可进入此判断,说明数据完美返回,并且请求页面存在

console.log(xml.responseText);

}

}

POST请求：

与get不同处，必须添加请求头在open方法前，send（）传输数据

Ajax。setRequestHeader（‘Content-type’,’application/x-www-form-urlencode’）

Ajax.open(‘post’,’post.php’);

Ajax.send(‘name=g&age=18’);

使用XMlHttpRequest对象传输xml或者json数据。

1. H5新增标签

Video 视屏播放

Audio 音频

Canvas 位图区域

Source 为video和audio新增数据源

Track video和audio指定字母

Svg 矢量图

Code 代码段

Figure 文档有关图例

Time 时间有关

Mark 高量引用文字

Progress 进度条

Menu 菜单

Embed 嵌入外部资源

Header nav footer

1. ES6的新属性
2. 返回数字问题？

500：服务器可能挂了，后台实体没有序列化，数据库数据出现冗余

405：前后端的http请求不一样

404：服务器找不到请求网页

415：请求格式不受请求页面支持

401：请求要求身份验证

200：客户端请求成功

1. 如何将字符串转化为json对象？

Json字符串转换为json对象

Var obj=eval(‘(‘+str+’)’);

Var obj=str.parseJSON();

Var obj=JSON.parse(str);

如果obj本身就是一个JSON对象，eval（）转换后为还是JSON对象，parseJSON（）函数处理后会抛出语法错误。

JSON对象转换为JSON字符串

Var string=Obj.toJSONString();

Var string=JSON.stringify(obj); //全局方法

判断是否为JSON字符串，如果JOSN.parse()能够转化成功；并且转换后的类型为Object且不为null，那么就为JSON格式的字符串。

1. ajax请求头中包含哪些信息？

设置数据格式：

发送json格式的数据

xhr.setRequestHeader(“content-type”,”application/json;charset=utf-8”)

发送表单数据：

Xhr.setRequestHeader(“content-type”,”application/x-www-form-urlencode;charset=utf-8”);

发送纯文本

Xhr.setRequestHeader(“content-type”,”text-plain;charset=utf-8”);

发送html文本

Xhr.setRequestHeader(“content-type”,”text/html;charset=utf-8”)

不带字符编码写法

Xhr.setRequestHeader(“content-type”,”application/json”)

值对大小写不敏感

Xhr.setRequestHeader(“content-type”,”APPlication/JSON,charset=utf-8”)

1. Px/Rem/em的用法区别？

Px像素，相对于长度单位，像素px是相对于显示器屏幕分辨率而言的

Px的特点：IE无法调整使用px作为单位的字体大小；国外大部分网站能够调整的原因是使用了em或rem作为字体单位；firfox可调整px，em和rem,IE内核占多数。

Em相对于长度单位，相对于当前对象内的文本的字体尺寸。

Em特点：em值不固定；em值会继承父级元素的字体大小。

Rem是css3新增的相对单位，root em，根em。Rem相对于HTML根元素，集相对大小和绝对大小有点于一体，除IE8以下版本都兼容

选择什么样的字体单位由项目决定，用户群都使用最新版的浏览器，使用rem，兼容性使用px或者和rem配合使用。

Px与rem的选择？

对于只需要适配少部分手机设备，且分辨率对页面影响不大，使用px

适配各种移动设备，使用rem，如适配ipone和ipad分辨率差别较大的设备。