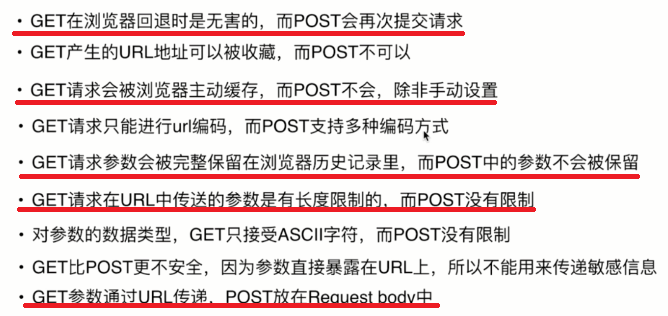
**前端常见面试题**

*自我介绍*

*HTML/HTTP*

1. POST和GET的区别



1. http1.0,1.1区别:多个keep-alive
2. 什么是https,与http有什么区别
3. 一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什么？

（1）查找浏览器缓存https 无法请求http的内容,包括ajax和iframe http可以请求https的ajax,包括ajax和iframe

（2）DNS解析、查找该域名对应的IP地址,找到之后进行三次握手、重定向（301）、发出第二个GET请求

（3）进行HTTP协议会话

（4）客户端发送报头(请求报头)

（5）服务器回馈报头(响应报头),结束开始四次挥手

（6）html文档开始下载

（7）文档树建立，根据标记请求所需指定MIME类型的文件

（8）文件显示

1. 跨域(postMessage,h5)
2. 前端如何存储数据 LocalStorage和sessionStorage,indexedDB的用法以及使用区别,可以聊到pwa(Service Worker)

*CSS*

1. 盒子模型

* 标准模型:width=content,IE模式(怪异模式):width=content+padding+border
* css如何切换这两种模式:box-sizing:content-box(默认值,标准模式),可以切换为border-box(怪异模式)
* 是否了解边距重叠,如何解释(俩div都margin,最后不是俩的和):答出BFC(块级格式化上下文)即可
* 触发BFC的方式（一下任意一条就可以）:
  + float的值不为none
  + overflow的值不为visible
  + display的值为table-cell、tabble-caption和inline-block之一
  + position的值不为static或则releative中的任何一个

2.两边300px中间自适应的三栏布局有几种写法:(五种:1.float,2.position,3.flex,4.table,5.grid)

3.css如何获取直接子节点:>,less中如何获取父级:&

4.如何自适应，不同分辨率如何处理:

1.媒体查询media

2.使用js根据不同分辨率计算出html根节点的font-size,然后使用rem

3.px,em,rem区别

1. 什么是重绘和重排(回流)

*JS*

1. Js都有哪些类型,基本类型,引用类型

基本类型： typeof(Number,String,Boolean,Undefined,Null,Symbol（ES6 新定义）

引用类型:object,array等

1. JS中BOM和DOM的区别与联系:

javacsript是通过访问BOM对象来访问、控制、修改客户端(浏览器)，由于BOM的window包含了document，可以说，BOM包含了DOM(对象)，浏览器提供出来给予访问的是BOM对象，从BOM对象再访问到DOM对象，从而js可以操作浏览器以及浏览器读取到的文档

1. 什么是事件委托

简单来说就是把 target 到根节点 div 通过 node.parentNode 遍历一遍，然后找到对应的委托元素节点，如果符合就缓存起来用于之后的操作，可以通过 jQuery.event.handlers 方法我们可以获取类似这种的一组数据结构

$(document).on( "click" , "button",function(e){}

就是当点击document时查看子元素是否是button,是就执行方法

1. js对象转数组:Array.prototype.slice.call(obj)
2. Es6循环数组的几种方法:map,foreach区别
3. 什么是同源策略:协议,域名和端口必须一致,否则无法操作
4. 前后端如何通信:
   1. AJAX(同源)
   2. WebSocket(不限制)
   3. CORS(新的,不限制) 浏览器发现跨域时自动在请求头添加Access-Control-Allow-Origin
   4. Fetch
5. 跨域通信的几种方式



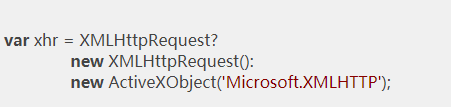
https网站如何请求http,ajax或者iframe:

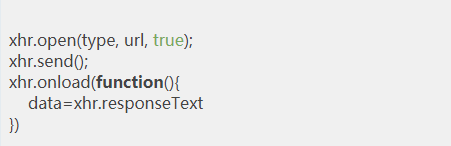
答: 1.https 无法请求http的内容,包括ajax和iframe

1. http可以请求https的ajax,包括ajax和iframe

结论:可以升级,不可以降级

1. 如何创建ajax





1. var,const和let区别（for （let i=0;i<10;i++）{let i=10;console.log(i)}）
2. 箭头函数(箭头函数中的this和普通函数中的this对比)
3. For in, for of 的区别
4. Promise(宏任务微任务)

setTimeout(()=>{

console.log('setTimeout')

})

let p1 = new Promise((resolve)=>{

console.log('Promise1')

resolve('Promise2')

})

p1.then((res)=>{

console.log(res)

})

console.log(1)

输出结果：

　　　　Promise1

　　　　1

　　　　Promise2

　　　　setTimeout

setTimeout:宏任务，Promise.then:微任务;**在当前的微任务没有执行完成时，是不会执行下一个宏任务的。**

**我们把宿主发起的任务称为宏观任务，把 JavaScript 引擎发起的任务称为微观任务。在当前的微任务没有执行完成时，是不会执行下一个宏任务的。参考:**<https://www.cnblogs.com/jiasm/p/9482443.html>

1. 什么是闭包:

一个function嵌套一个function,内部的fn调用外部fn的变量或者参数,导致外部fn变量不能销毁,这一种情况叫闭包

<script>

function f1(){

var n=999;

function f2(){

n++;

alert(n);

}

return f2;

}

var result=f1();

result(); // 1000

result(); // 1001

</script>

1. 判断某值是否在数组中[].includes()
2. 变量提升:

1行 a()

2行 var a=10

3行 function a(){console.log(a)}

4行 a()第一行和第四行输出什么

答案：

第一行：输出function a()函数体

第四行：报错语法错误

因为变量声明var提升，答出这里就可以

但是函数体提升优先级更高

1. this指向

const obj={

a:10,

c:this.a,

getA:function(){

return this.a

}

}

console.log(obj.c); //undefined

console.log(obj.getA()); //10

const test = obj.getA; //undefined

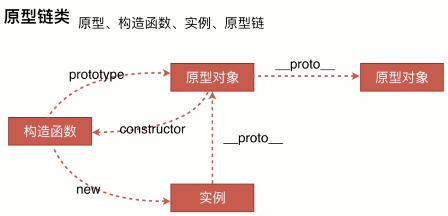
console.log(test()); //undefined

1. 安全类:

CSRF:跨站请求伪造,解决方式:token

XSS:跨域脚本攻击,解决方式:不允许网页插入脚本

1. 原型,构造函数,实例,原型链的关系:



1. instanceof原理:

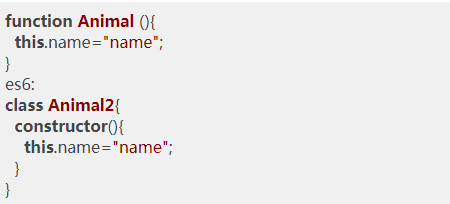
实际上就是判断 实例对象.\_\_proto\_\_(可以无限\_\_proto\_\_) 与构造函数.prototype是不是引用的同一个地址

function Student() { }

var studentA = new Student();

studentA instanceof Student

1. 类的声明方式有几种:



1. 如何实现继承:



1. 装饰器是什么,有什么作用,在哪里用到过

***数据结构***

1.排序(快速排序,冒泡排序)

2.堆栈,队列,链表

3.递归

***Webpack***

1. 什么是webpack(一个模块打包工具,自身只能翻译import这样的语句)
2. 什么是Chunk:webpack打包生成了几个js文件,就是几个Chunk
3. npm和npx的区别
4. 是否知道babel,原理是什么(AST)
5. babel/polyfill作用
6. source-map是什么
7. Webpack如何打包多入口

    entry: {

        main: './src/index.js',

        // lesson:3-6 打包多个文件,key随便起

        // sub: './src/index.js'

},

1. dev-server的作用,以及如何开启热更新
2. tree-shaking是什么,如可开启
3. 如何设置打包后根据文件内容不同变换hash值
4. Webpack-module-loader,如何编写自己的loader或者plugin
5. Create-react-app/vue-cli生成的项目中webpack配置被隐藏,如何开启(npm run eject),vue不可以

***框架***

vue

1. 生命周期
2. 父组件与子组件传值 emit,vuex,inject/provide
3. 路由之间跳转

Vue

* <router-link :to="{name:'home'}">
* <router-link :to="{name:'home', params: {id:1}}">
* this.$router.push

1. 用过哪种ui库,有什么坑吗
2. 懒加载的概念原理以及如何实现

原理:页面渲染时先加载首屏需要的js文件,剩下的需要时再ajax加载,缺点,ajax有延迟,可能会出现白屏,所以得加loading,改进:打包时把所有页面文件以js标签的形式写入到html中,但是js标签上加上rel="prefetch",优先加载首屏需要的js,当网络空闲时再加载其他js,这样就不会有ajax了 注意:当前页面用preload,非当前页面用prefetch

1. 状态管理有哪些方式
2. Vue:v-show和v-if指令的共同点和不同点
3. 双向绑定原理
4. 循环时”为什么使用key(虚拟dom)
5. 如何让CSS只在当前组件中起作用:<style scoped>
6. <keep-alive></keep-alive>的作用是什么?
7. slot是什么
8. 权限管理如何做

react

1.生命周期

2.父组件与子组件传值

3.路由之间跳转

* import { BrowserRouter as Router, Route, Link } from "react-router-dom"
* <Link to="/">Home</Link>
* this.props.history.push

1. 用过哪种ui库,有什么坑吗

5.懒加载的概念原理以及如何实现

6.状态管理有哪些方式(redux,mobx)

7.循环时”为什么使用key(虚拟dom)

8.如何改变state的值,setState是否是同步的

9.React 中 refs 的作用是什么？

10.component和pureComponent的区别,什么是无状态组件

1. 权限管理如何做
2. 如何搭建的框架:

* Create-react-app
* 使用的sass还是less,
* Less如何配置 (eject 然后再配webpack)
* 国际化如何做
* Ui库如何使用
* 权限管理如何做
* 状态管理如何去做
* 代码检查(eslint)

***学习能力***

除了前端还会什么语言(Typescript)

从事编程的原因，以判断是否有兴趣

查看自学了什么技术

面试问题:如何快速学会一套新框架 答:从ui，路由，状态管理三方面学习

遇到问题去哪查资料(google,stack over flow)

***技术广度***

都有哪些服务器

Linux是否了解

App开发几种常见模式

如何判断一个请求是不是ajax请求 headers 带 x-request-with

如何写一个Css ie hack,chorme hack等:媒体查询写一个只有当前浏览器的css样式

跨域代码是否可以获取所有的headers属性,如何获取:header 后端添加Access-Control-Expose-Headers:content-type, cache, ...

vue.use()注册组件与vue.component()区别:

Vue.use(plugin)的初始化方法{install:function(){}}里面可以一次性注册多个组件。

Vue.component只能一个一个注册。(这是一个全局注册),组件中的components:{HelloWorld}属于局部注册

install可以做更多的事情

Vue.extend:需要new然后使用

框架区别:vue没有(大厂)好用的富文本编辑器,react,没有keep-alive