



本面试题仅供观看，不可商用
如商用后果自负

欢迎大家为本群做推广
加群暗号：我是我



JavaScript 部分

要点：数据类型、面向对象、继承、闭包、插件、作用域、跨域、原型链、模块化、自定义事件、异步装载回调、模板引擎、Nodejs 等。

js 的几种数据类型：

number,string,boolean,object,undefined

js 的常见内置对象类：Date,Array,Math、Number、

Web 明教

Boolean、String、Array、RegExp、Function...

通常可以做一些小练习来判断 TA 的水平，js 虽然很

灵活，但是具体的代码和实现方式能体现出一个人的全局观，随着代码规模的增长，复杂度增加，如何合理划分模块实现功能和接口的能力比较重要。（下面例题）

```
["1", "2", "3"].map(parseInt)
```

```
[typeof null, null instanceof  
Object][ [3,2,1].reduce(Math.pow),  
[].reduce(Math.pow)] ]
```

```
var val = 'smtg';
```

```
console.log('Value is ' + (val === 'smtg') ?
```

```
'Something' : 'Nothing');
```

1.创建一个对象

```
function Person(name, age)  
{ this.name = name;  
  this.age = age;  
  this.sing = function()  
{ alert(this.name) } }
```

【web 明教】天下武功，唯快不破

Web 明教

}

2.谈谈 This 对象的理解。

this 是 js 的一个关键字,随着函数使用场合不同,**this** 的值会发生变化。

但是总有一个原则,那就是 **this** 指的是调用函数的那个对象。

this 一般情况下: 是全局对象 **Global**。 作为方法调用,那么 **this** 就是指这个对象

3.事件、IE 与火狐的事件机制有什么区别? 如何阻止冒泡?

1. 我们在网页中的某个操作 (有的操作对应多个事件)。例如: 当我们点击一个按钮就会产生一个事件。是可以被 **JavaScript** 侦测到的行为。
2. 事件处理机制: **IE** 是事件冒泡、火狐是 事件捕获;
3. **ev.stopPropagation();**
- 4.什么是闭包 (**closure**), 为什么要用?

待完善

执行 `say667()` 后, `say667()` 闭包内部变量会存在, 而 闭包内部函数的内部变量不会存在. 使得

Javascript

的垃圾回收机制 GC 不会收回 `say667()` 所占用的资源, 因为 `say667()` 的内部函数的执行需要依赖 `say667()` 中的变量。这是对闭包作用的非常直白的描述.

```
function say667() {  
    // Local variable that ends up within  
closure  
    var num = 666;  
    var sayAlert = function() { alert(num); }  
    num++;  
    return sayAlert;  
}
```

```
var sayAlert = say667();
```

```
sayAlert();//执行结果应该弹出的
```

667 5. 如何判断一个对象是否属于某个类?

使用 instanceof （待完善）

【web 明教】天下武功，唯快不破

Web 明教

```
if(a instanceof  
    Person){ alert('yes');  
}
```

6.new 操作符具体干了什么呢？

- 1、创建一个空对象，并且 **this** 变量引用该对象，同时还继承了该函数的原型。
- 2、属性和方法被加入到 **this** 引用的对象中。
- 3、新创建的对象由 **this** 所引用，并且最后隐式的返回 **this** 。

```
var obj  = {};  
obj.__proto__ = Base.prototype;  
Base.call(obj);
```

7.JSON 的了解

JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式。它是基于 JavaScript 的一个子集。数据格式简单，易于读写，占用带宽小

```
{'age':'12', 'name':'back'}
```

8.js 延迟加载的方式有哪些

Web 明教

defer 和 async、动态创建 DOM 方式（用得最多）、按需异步载入 js

9.ajax 是什么?ajax 的交互模型?同步和异步的区别?如何解决跨域问题?

待完善

1. 通过异步模式，提升了用户体验
2. 优化了浏览器和服务器之间的传输，减少不必要的数据往返，减少了带宽占用
3. Ajax 在客户端运行，承担了一部分本来由服务器承担的工作，减少了大用户量下的服务器负载。

2. Ajax 的最大的特点是什么。

Ajax 可以实现动态不刷新（局部刷新）

readyState 属性 状态 有 5 个可取值：0=未初始化，1=正在加载 2=以加载，3=交互中，4=完成
ajax 的缺点

Web 明教

1、ajax 不支持浏览器 back 按钮。

2、安全问题 AJAX 暴露了与服务器交互的细节。

3、对搜索引擎的支持比较弱。

4、破坏了程序的异常机制。

5、不容易调试。

跨域：jsonp、iframe、window.name、
window.postMessage、服务器上设置代理页面

10.模块化怎么做？

立即执行函数,不暴露私有成员

```
var module1 = (function(){  
    var _count = 0;  
    var m1 =  
        function(){ //...  
    };  
    var m2 = function(){
```

Web 明教

```
        //...  
    };  
    return {  
        m1 : m1,  
        m2 : m2  
    };  
})();
```

11.对 **Node** 的优点和缺点提出了自己的看法:

* (优点) 因为 **Node** 是基于事件驱动和无阻塞的, 所以非常适合处理并发请求,

因此构建在 **Node** 上的代理服务器相比其他技术实现 (如 **Ruby**) 的服务器表现要好得多。

此外, 与 **Node** 代理服务器交互的客户端代码是由 **javascript** 语言编写的,

因此客户端和服务端都用同一种语言编写, 这是非常美妙的事情。

* (缺点) **Node** 是一个相对新的开源项目, 所以不太稳定, 它总是一直在变,

而且缺少足够多的第三方库支持。看起来, 就像是 **Ruby/Rails** 当年的样子。

12. 异步加载的方式

(1) defer, 只支持 IE

(2) async:

(3) 创建 script, 插入到 DOM 中, 加载完毕后 callback

document.write 和 innerHTML 的区别

document.write 只能重绘整个页面

innerHTML 可以重绘页面的一部分

13. 告诉我答案是多少?

```
(function(x){  
    delete x;  
    alert(x);  
})(1+5);
```

函数参数无法 delete 删除, delete 只能删除通过 for

【web 明教】天下武功，唯快不破

Web 明教

in 访问的属性。

当然，删除失败也不会报错，所以代码运行会弹出“1”。

14.JS 中的 call()和 apply()方法的区别？

例子中用 add 来替换 sub，add.call(sub,3,1)
== add(3,1) ，所以运行结果为：alert(4);

注意：js 中的函数其实是对象，函数名是对
Function 对象的引用。

```
function add(a,b)
{
    alert(a+b);
}
```

```
function sub(a,b)
{
    alert(a-b);
}
```

```
add.call(sub,3,1);
```

15.Jquery 与 jQuery UI 有啥区别？

Web 明教

*jQuery 是一个 js 库,主要提供的功能是选择器,属性修改和事件绑定等等。

*jQuery UI 则是在 jQuery 的基础上,利用 jQuery 的扩展性,设计的插件。

提供了一些常用的界面元素,诸如对话框、拖动行为、改变大小行为等等

16.jquery 中如何将数组转化为 json 字符串,然后再转化回来?

jQuery 中没有提供这个功能,所以你需要先编写两个 jQuery 的扩展:

```
$.fn.stringifyArray = function(array)
{
    return JSON.stringify(array)
}
```

```
$.fn.parseArray = function(array)
{
    return JSON.parse(array)
}
```

【web 明教】天下武功，唯快不破

Web 明教

然后调用：

```
$.("").stringifyArray(array)
```

17.JavaScript 中的作用域与变量声明提升？

其他部分

（HTTP、正则、优化、重构、响应式、移动端、团队协作、SEO、UED、职业生涯）

*基于 **Class** 的选择性的性能相对于 **Id** 选择器开销很大，因为需遍历所有 **DOM** 元素。

*频繁操作的 **DOM**，先缓存起来再操作。

用 **Jquery** 的链式调用更好。

比如：`var str=$("a").attr("href");`

```
*for (var i = size; i < arr.length; i++) {}
```

for 循环每一次循环都查找了数组 (**arr**) 的 **.length** 属性，在开始循环的时候设置一个变量来存储这个数字，可以让循环跑得更快：

```
for (var i = size, length = arr.length; i < length; i++) {}
```

前端开发的优化问题（看雅虎 **14** 条性能优化原则）。

(1) 减少 http 请求次数: CSS Sprites, JS、CSS 源码压缩、图片大小控制合适; 网页 Gzip, CDN 托管, data 缓存, 图片服务器。

(2) 前端模板 JS+数据, 减少由于 HTML 标签导致的带宽浪费, 前端用变量保存 AJAX 请求结果, 每次操作本地变量, 不用请求, 减少请求次数

(3) 用 innerHTML 代替 DOM 操作, 减少 DOM 操作次数, 优化 javascript 性能。

(4) 当需要设置的样式很多时设置 className 而不是直接操作 style。

(5) 少用全局变量、缓存 DOM 节点查找的结果。减少 IO 读取操作。

(6) 避免使用 CSS Expression (css 表达式) 又称 Dynamic properties(动态属性)。

(7) 图片预加载, 将样式表放在顶部, 将脚本放在底部 加上时间戳。

【web 明教】天下武功，唯快不破

Web 明教

(8) 避免在页面的主体布局中使用 **table**, **table** 要等其中的内容完全下载之后才会显示出来, 显示比 **div+css** 布局慢。

http 状态码有那些? 分别代表是什么意思?

100-199 用于指定客户端应相应的某些动作。

200-299 用于表示请求成功。

300-399 用于已经移动的文件并且常被包含在定位头信息中指定新的地址信息。

400-499 用于指出客户端的错误。**400** 1、语义有误, 当前请求无法被服务器理解。**401** 当前请求需要用户验证 **403** 服务器已经理解请求, 但是拒绝执行它。

500-599 用于支持服务器错误。**503** - 服务不可用

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? (流程说的越详细越好)

你所知道的页面性能优化方法有那些?

除了前端以外还了解什么其它技术么? 你最最厉害的

【web 明教】天下武功，唯快不破

Web 明教

技能是什么？

AMD (Modules/Asynchronous-Definition)、CMD
(Common Module Definition) 规范区别？

18.谈谈你认为怎样做能是项目做的更好？

19.你对前端界面工程师这个职位是怎么样理解的？
它的前景会怎么样？

20.加班的看法

加班就像借钱，原则应当是-----救急不救穷

21.平时如何管理你的项目，如何设计突发大规模并发架构？

先期团队必须确定好全局样式（`globe.css`），编码模式(`utf-8`) 等

编写习惯必须一致（例如都是采用继承式的写法，单样式都写成一行）；

【web 明教】天下武功，唯快不破

Web 明教

标注样式编写人，各模块都及时标注（标注关键样式调用的地方）；

页面进行标注（例如 页面 模块 开始和结束）；

CSS 跟 HTML 分文件夹并行存放，命名都得统一（例如 `style.css`）

JS 分文件夹存放 命名以该 JS 功能为准英文翻译；

图片采用整合的 `images.png` `png8` 格式文件使用尽量整合在一起使用方便将来的管理
那些操作会造成内存泄漏？

内存泄漏指任何对象在您不再拥有或需要它之后仍然存在。

垃圾回收器定期扫描对象，并计算引用了每个对象的其他对象的数量。如果一个对象的引用数量为 0（没有其他对象引用过该对象），或对该对象的惟一引用是循环的，那么该对象的内存即可回收。

`setTimeout` 的第一个参数使用字符串而非函数的

【web 明教】天下武功，唯快不破

Web 明教

话，会引发内存泄漏。

闭包、控制台日志、循环（在两个对象彼此引用且彼此保留时，就会产生一个循环）

23.你说你热爱前端，那么应该 **WEB** 行业的发展很关注吧？说说最近最流行的一些东西吧？

Node.js、Mongodb、npmM、MVVM、MEAN

24.你了解我们公司吗？说说你的认识？

因为我想去阿里，所以我针对阿里的说
最羡慕就是在双十一购物节，**350.19** 亿元，每分钟支付 **79** 万笔。海量数据，居然无一漏单、无一故障。太厉害了。

25.移动端（比如：**Android IOS**）怎么做好用户体验？

作为一名前端工程师，无论工作年头长短都应该必须掌握的知识点有：

1、DOM 结构 —— 两个节点之间可能存在哪些关系以及如何在节点之间任意移动。

Web 明教

2、DOM 操作 —— 如何添加、移除、移动、复制、创建和查找节点等。

3、事件 —— 如何使用事件，以及 IE 和标准 DOM 事件模型之间存在的差别。

4、XMLHttpRequest —— 这是什么、怎样完整地执行一次 GET 请求、怎样检测错误。

5、严格模式与混杂模式 —— 如何触发这两种模式，区分它们有何意义。

6、盒模型 —— 外边距、内边距和边框之间的关系，及 IE8 以下版本的浏览器中的盒模型

7、块级元素与行内元素 —— 怎么用 CSS 控制它们、以及如何合理的使用它们

8、浮动元素 —— 怎么使用它们、它们有什么问题以及怎么解决这些问题。

9、HTML 与 XHTML —— 二者有什么区别，你觉

Web 明教

得应该使用哪一个并说出理由。

10、JSON —— 作用、用途、设计结构。

Web 明教

Web 明教