

1. 你做的页面在哪些浏览器测试过？这些浏览器的内核分别是什么？

- IE 内核 -ms
- Firefox 内核 -moz
- Safari:webkit 内核 -webkit
- Opera:内核 -o
- Chrome: webkit 内核 -webkit

2. 每个 HTML 文件里开头都有个很重要的东西，Doctype，知道这是干什么的吗？

答案：<!DOCTYPE> 声明位于文档中的最前面的位置，处于 <html> 标签之前。此标签可告知浏览器文档使用哪种 HTML 或 XHTML 规范还是 HTML5。（重点：告诉浏览器按照**何种规范**解析页面）

3. HTML 和 XHTML 的区别？

XHTML 是基于 XML 语法规则的 HTML 语法，比 HTML 语法要严格，比如说标签必须成对，例如<br/>

4. HTML 和 XML 的区别

HTML 是超文本标记语言，主要用于做网页

XML 是可扩展的标记语言，特点是自定义标签，主要作为一种数据格式做数据交互，或者做配置文件

XML 语法：必须有根标签，标签区分大小写且必须成对出现，属性必须加双引号

5. Quirks（怪异）模式是什么？它和 Standards（标准）模式有什么区别

有两种情况浏览器的盒子模型会按照怪异模式执行

1 在 IE6 下不写<!DOCTYPE>

2 CSS3 设置 box-sizing:border-box

盒子模型的区别是：

怪异模式：设置高度宽度=内容+padding+border

标准模式：设置高度=内容，padding 和 border 会增加高度宽度

6. div+css 的布局较 table 布局有什么优点？

1. 表格的样式一般是用 html 属性设置（高度，宽度，边框线）样式，div+css 主要是用 css 控制样式（优点改变样式只要改 css 就可以了）
2. Div 标签少，页面加载速度更快、结构化清晰、页面显示简洁
3. DIV 和 CSS 表现（样式）与结构（标签）相分离
4. 易于优化（seo）搜索引擎更友好，排名更容易靠前
5. Table 必须完全加载完毕才能显示，而 div 可以一边加载一边显示

7. img 的 alt 与 title 有何异同？b: strong 与 em 的异同？

title 是图片或者链接鼠标悬停的提示

alt 只是图片的替代文字，图片不显示的时候可以显示

8. 你能描述一下渐进增强和优雅降级之间的不同吗？

都是针对于 css 在浏览器高低版本中运行兼容性的问题

渐进增强 progressive enhancement: 针对**低版本**浏览器进行构建页面，保证最基本的功能，然后再针对**高级浏览器**进行效果、交互等改进和追加功能达到更好的用户体验（先低后高）

优雅降级 graceful degradation: 一开始就构建完整的功能，然后再针对低版本浏览器进行兼容（先高后低）

9. 为什么利用多个域名来存储网站资源会更有效？

- CDN 缓存更方便
- 突破浏览器并发限制，浏览器对同一个域名的并发请求个数有限制
- 节约 cookie 带宽，同一个域名 cookie 数量和大小也有限制
- 节约主域名的连接数，优化页面响应速度

10. 请谈一下你对网页标准和标准制定机构重要性的理解。

（无标准答案）网页标准和标准制定机构都是为了让 web 发展的更‘健康’，开发者遵循统一的标准，降低开发难度，开发成本，SEO 也会更好做，也不会因为滥用代码导致各种 BUG、安全问题，最终提高网站易用性。

11. html5 的 web storage 和 cookie 的区别

1. web storage 存储容量大 cookie 存储受大小和条数限制
2. cookie 每次请求都会自动存储在请求报头中发送给服务器（浪费带宽），web storage 不会

12. web storage 的 localStorage 和 sessionStorage 的区别？

localStorage 永久性存储当前域名的数据

sessionStorage 是会话级，浏览器关闭就清空

13. GET 请求和 POST 请求的区别？

1. 发送数据方式：get 把数据追加 url 后面，post 把数据放到请求正文中
2. 数据量：get 小，post 几乎无限制
3. 安全性：get 不安全，post 安全
4. 中文问题：get 不能直接发中文，post 可以

14. 简述一下 src 与 href 的区别？

src 的标签：img frame iframe script

href 的标签：a link

<script src=”外部 JS 文件”>当浏览器解析到该元素时，会暂停其他资源的下载和处理，直到将该资源加载、编译、执行完毕，将会暂停网页中的元素加载，这也是为什么将 js 脚本放在底部而不是头部，如果希望<script>引入外部 js 不会产生阻塞，需要加 defer 属性或者 async

<link href=”common.css” rel=”stylesheet”/>当浏览器解析到该元素时，就会并

行下载资源并且不会停止对当前文档的处理

15. 知道的网页制作会用到的图片格式有哪些？

png-8, png-24, jpeg, jpg, gif, svg 后期新技术标准 Webp 谷歌出品 Apng safari 支持

16. 在 css/js 代码上线之后开发人员经常会优化性能，从用户刷新网页开始，一次 js 请求一般情况下有哪些地方会有缓存处理？

dns 缓存，cdn 缓存，浏览器缓存，服务器缓存

17. 一个页面上有大量的图片（大型电商网站），加载很慢，你有哪些方法优化这些图片的加载，给用户更好的体验

懒加载（延迟加载）：一开始图片不加载，利用判断滚动条位置，动态通过 ajax 加载图片

预加载：提前下载图片，在图片使用前就下载

如果图片为 css 图片，可以使用 CSSsprite(雪碧图,减少请求次数),SVGsprite, Iconfont（字体图标）、Base64 图片等技术。

图片过大可以先做成缩略图，然后连接到大图

18. 闭包的理解？

内层函数使用了外层函数的变量，会导致外层函数即使调用结束，局部变量也不会在内存中释放，会造成内存泄露，最常用的地方就是事件回调函数里面使用外层作用域的数据

19. 你如何理解 HTML 结构的语义化？

语义化标签：header footer nav section article aside h1 img table div title

优点：代码可读性好，有利于 SEO 搜索引擎优化，符合 W3C 标准

20. 有哪项方式可以对一个 DOM 设置它的 CSS 样式?

行内样式

内联样式

外部样式表文件

21. CSS 都有哪些选择器?

标签选择器 (用 HTML 标签申明) `body h1`

id 选择器 (用 DOM 的 ID 申明) `#p1`

类选择器 (用一个样式类名申明) `.title1`

属性选择器 (用 DOM 的属性申明, 属于 CSS2, IE6 不支持, 不常用, 不知道就算了) `a[href]`

除了前 3 种基本选择器, 还有一些扩展选择器, 包括

后代选择器 (利用空格间隔, 比如 `div .a{ }`)

父子选择器 `div>a`

群组选择器 (利用逗号间隔, 比如 `p,div,#a{ }`) `h1,h2,h3`

伪选择器 `a:hover`

CSS 选择器的优先级是怎样定义的?

标签选择器: 权重 1

类选择器: 权重 10

id 选择器: 权重 100

行内样式: 权重 1000

`!important` 优先级高于一切

例如 `div.test1 .span li` 权重 22

同样的权重, 后面定义的优先于前面定义的

22. CSS 中可以通过哪些属性定义, 使得一个 DOM 元素不显示在浏览器可视范围内?

设置 `display` 属性为 `none`, 或者设置 `visibility` 属性为 `hidden`

区别: display 不占位 visibility 占位

技巧性:

设置宽高为 0, 设置透明度为 0, 设置 z-index 位置在-1000

23. 超链接访问过后 hover 样式就不出现的问题是什么? 如何解决?

定义链接的伪类必须按照以下顺序: link,visited,hover,active

24. 什么是 Css Hack? ie6,7,8 的 hack (浏览器兼容) 分别是什么?

答案: 针对不同的浏览器写不同的 CSS code 的过程, 就是 CSS hack。

25. 行内元素和块级元素的具体区别是什么? 行内元素的 padding 和 margin 可设置吗?

行内元素: 多个元素可以在一行排列, 不能设置宽度高度 常见 a span

块级元素: 一个元素独占一行 常见 p div h1 h2 ul

行内元素的 padding 和 margin 可设置, 但是只能设置水平位置的 padding 和 margin 垂直的不能设置

inline-block 元素: 可以设置宽度高度, 且多个元素可以在一行排列, 例如 <input> 、<img> 、<button> 、<textarea> 、<label>

26. 什么是外边距重叠? 重叠的结果是什么?

在 CSS 当中, 相邻的两个盒子 (可能是兄弟关系也可能是祖先关系) 的外边距可以结合成一个单独的外边距。

一般相邻 div 垂直 margin 会重叠, 而水平 margin 不重叠

27. rgba()和 opacity 的透明效果有什么不同?

rgba()设置颜色透明度, opacity 设置元素透明度

rgba()样式不能不继承

28. css 中可以让文字在垂直和水平方向上重叠的两个属性是什么?

垂直方向: line-height

水平方向: letter-spacing

## 29. px 和 em, rem 的区别

px 和 em 都是长度单位, 区别是, px 的值是固定的, 指定是多少就是多少, 计算比较容易。em 得值不是固定的, em 是以父元素的字号为基准, 并且 em 会继承父级元素的字体大小。

浏览器的默认字体高都是 16px。所以未经调整的浏览器都符合: 1em=16px。那么 12px=0.75em, 10px=0.625em。

rem 同 em, 但是不能被继承, 默认是以 html 元素的字号为基准

## 30. 描述一个 "reset" 的 CSS 文件并如何使用它。知道 normalize.css 吗? 你了解他们的不同之处?

统一不同浏览器的元素的默认样式, normalize.css 是一个开源提供的写好的 reset.css 文件

## 31. Sass、LESS 是什么? 大家为什么要使用他们?

由于 css 没有变量的概念, 可以借助 sass 或者 less 的工具来模拟变量, 便于编写维护 css 代码

## 32. 知道 css 有个 content 属性吗? 有什么作用?

知道。css 的 content 属性专门应用在 before/after 伪元素上, 用于来插入生成内容。清除浮动的一种写法就是用 content 实现的

## 33. JavaScript 是一门什么样的语言, 它有哪些特点?

是脚本语言, 是解释性而不是编译性, 弱类型语言, 变量不需要明确定义数据类型

## 34. JavaScript 的数据类型都有什么?

基本数据类型: String,boolean,Number,Undefined, Null

引用数据类型: Object(Array,Date,RegExp) Function

判断数据类型 typeof, 判断是哪个类 instanceof

35. 已知 ID 的 Input 输入框, 希望获取这个输入框的输入值, 怎么做?

文本框对象.value 属性

36. 希望获取到页面中所有的 checkbox 怎么做?

document.getElementsByName('checkbox 的 name')

37. 注册事件的三种方式

直接在 DOM 里绑定事件: <div onclick=" test()" ></div>

在 JS 里通过 onclick 绑定: xxx.onclick = test

通过事件添加进行绑定: addEventListener(xxx, 'click' , test)

IE 下 attachEvent

JS 的事件流的两种情况:

事件冒泡: 从事件源开始触发到最外层的父元素

事件捕获 (IE 不存在): 从最外层的父元素触发到事件源

38. 什么是 Ajax 和优缺点?

Ajax 是通过 javascript 发出异步请求, 和服务端进行数据交互

优点: 提高用户体验, 实现数据局部刷新

缺点: 不支持 SEO 搜索引擎优化, 没有历史记录要实现 ajax 下的前后退功能成本较大, 跨域问题限制

39. 什么是 JSON, 相对于 XML 有什么优点?

JSON 是一种轻量级的数据交换格式, JSON 数据的书写格式是: 名称/值对

特殊字符需要转义, 例如 单引号 \u0027

相对于 xml 的优点: 存储相同的数据占据的空间更少



#### 40. 变量声明提升？

在函数内部声明的变量，默认在执行的时候，声明位置会自动提升到函数最前面

#### 41. 方法通过 prototype 定义，而不是直接定义到类中有什么优点？

在类中定义方法，每次创建对象都会重复声明存储方法，而在原型中定义方法，不管创建多少对象，方法只声明存储一次，提高效率

#### 42. call 和 apply 的区别和作用？

call 和 apply 的唯一区别就是传递参数一个用参数列表 (call)，一个用参数数组 (apply)

功能都是改变当前函数调用的上下文对象

#### 43. 直接在对象的原型上添加方法是否安全？尤其是在 Object 对象上

如果是团队协作，尽量不要再 JS 对象中添加方法，会造成全局污染

#### 44. 函数声明与函数表达式的区别？

```
//m1(); //正确
//函数声明，会自动位置提升
function m1() {
    alert('m1');
}
// m2(); //找不到函数
//函数表达式，不会自动前置
var m2 = function() {
    alert('m2');
}
m2(); //正确
```

45. this 在各种情况下代表的含义?

函数级调用, this 代表全局 windows 对象

对象级调用, this 代表当前对象

事件回调函数中, this 代表当前事件源对象

46. 原生 JS 的 window.onload 与 JQuery 的 \$(document).ready(function(){})) 有什么不同?

window.onload 在网页的所有 dom 加载完毕且所有的外部资源加载也完毕才会触发

\$(document).ready(function(){})) 网页所有 dom 加载完毕就触发

47. Doctype 作用? 严格模式与混杂模式如何区分? 它们有何意义?

<!DOCTYPE> 声明位于文档中的最前面, 处于 <html> 标签之前。告知浏览器的解析器, 用什么文档类型 规范来解析这个文档

§ 严格模式: 以浏览器最高 (最新) 标准执行

混杂模式: 以兼容低版本浏览器的标准执行, 不写 DOCTYPE, 或者 DTD 声明为 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional> 都会触发混杂模式

48. link 和 @import 的区别是?

§ link 属于 HTML 标签, 而 @import 是 CSS 提供的

而 @import 是 CSS 提供的语法

页面被加载的时, link 会同时被加载, 而 @import 引用的 CSS 会等到页面被加载完

49. HTML5 新增的特性?

语义化标签, 绘画 canvas 和 SVG 图形, 媒体的 video 和 audio 元素, 本地离线存储, 新增表单元素, 地理定位, 拖放功能, 应用缓存, WebWorker 多线程, 消息推送

50. iframe 的缺点？

iframe 会阻塞主页面的 Onload 事件

iframe 和主画面用的是同一个连接池，会影响主页面的并发

51. 如何实现浏览器内多个标签页之间的通信？

调用 localStorage、cookies 等本地存储方式

52. websocket 如何兼容低浏览器做推送？

用 ajax 轮询代替，每隔一段毫秒发送一次请求

53. position 的值，relative 和 absolute 定位原点是？

position 的四个取值：

static:默认，正常文档输出流

absolute: 如果父元素设置了 position，相对于父元素定位，否则相对于网页原点定位

fixed:相对于视口定位（相对于浏览器窗口定位）

relative: 相对于本身的原始位置

54. 解释下浮动和它的工作原理？清除浮动的技巧

浮动可以使块级元素横向排列，可以使文字产生文字环绕，是脱离了正常的文档输出流

清除浮动：

下面的元素 clear: both;

父元素：overflow:hidden

after 动态设置 content 的 clear: both;

55. 如果需要手动写动画，你认为最小时间间隔是多久，为什么？（阿里）

多数显示器默认频率是 60Hz，即 1 秒刷新 60 次，所以理论上最小间隔为  $1/60 * 1000ms = 16.7ms$

56. display:inline-block 怎么去掉间隙空白？

父元素 font-size:0px

移除元素代码之间的空白

父元素 letter-spacing 设置负数

57. 常见后台开发语言？

Java 适合大型项目

Php 适合中小型项目

NodeJs 适合高并发的中小型项目

58. use strict 的作用？

严格模式，不允许不规范的语法执行

59. new 操作符具体干了什么呢？

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  
属性和方法被加入到 this 引用的对象中

新创建的对象由 this 所引用，并且最后隐式的返回 this

60. Javascript 中，有一个函数，执行时对象查找时，永远不会去查找原型，这个函数是？

hasOwnProperty 方法

61. js 延迟加载的方式有哪些？

<script>标签中加入 defer 和 async 使 js 可以和其他资源并行下载

动态创建 DOM 方式（用得最多）

```
function loadJS() {  
    var js = document.createElement('script');  
    js.type = 'text/javascript';  
    js.src = '2.js';  
    document.getElementsByTagName('head')[0].appendChild(js);  
}
```

}

## 62. 同步和异步的区别？

同步不允许并发执行，在 ajax 请求发出期间，其他操作都是冻结

异步允许同时发出多个 ajax 请求，在请求期间，其他操作照常操作

## 63. 解决跨域请求？

Jsonp 请求，iframe 浮动框架

## 64. 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（做 dom 操作）

## 65. 如何编写高性能的 Javascript？

dom 对象用 getXxxx 得到之后赋给局部变量

进可能让内部作用不要层层查找外层作用域的数据，直接作为参数传入效率比较高

## 66. WEB 应用从服务器主动推送 Data 到客户端有那些方式？

推送消息：原则上来说是服务器可以主动给客户端响应信息，而不需要客户端每次都发出请求，目前的推送手段有三种：

1. 假推送：ajax 轮询操作，就是利用定时器（setInterval）每隔一段时间，主动向服务器发送一次请求，然后得到最新的响应，优点是简单粗暴，缺点是发送请求太频繁影响效率，再就是不能达到毫秒级更新。
2. 长连接：推翻了 http 的基本原则请求发出连接服务器，响应结束立即断开服务器连接，而是让客户端和服务器永久保持连接
3. 服务器发送事件（server-sent event）
4. WebSocket 类似于服务器发送事件

## 67. 常见 http 状态码

200 响应成功

304 浏览器缓存

404 资源找不到

405 没有对应的 get 和 post 的代码

500 服务器程序错误

## 68. 当用户发出请求开始，到响应结束，中间的过程有哪些？

1. 输入 url 地址
2. 解析域名，查找的对应的 IP 地址（包括读取缓存的过程：浏览器换成——系统缓存——路由器缓存）
3. 浏览器向服务器端发送一个请求
4. 服务器端处理执行请求
5. 服务器端产生响应
6. 浏览器显示 html