**\*\*前端开发知识点\*\***

**HTML&CSS：**

对Web标准的理解、浏览器内核差异、兼容性、hack、CSS基本功：布局、盒子模型、选择器优先级、HTML5、CSS3、Flexbox

**JavaScript：**

数据类型、运算、对象、Function、继承、闭包、作用域、原型链、事件、RegExp、JSON、Ajax、DOM、BOM、内存泄漏、跨域、异步装载、模板引擎、前端MVC、路由、模块化、canvas、ECMAScript

**其他：**

移动端、响应式、自动化构建、HTTP、离线存储、WEB安全、优化、重构、团队协作、可维护、易用性、SEO、UED、架构、职业生涯、快速学习能力

一、HTML/CSS:

1.让一个input背景颜色变成红色.

style="border:0px;background-color:pink"

2.请画出网页中盒子模型的结构.

（1）有两种， IE 盒子模型、W3C 盒子模型；

（2）盒模型： 内容(content)、填充(padding)、边界(margin)、 边框(border)；

（3）区 别： IE的content部分把 border 和 padding计算了进去;

3.写出CSS优先级顺序.

\* 优先级就近原则，同权重情况下样式定义最近者为准;

\* 载入样式以最后载入的定位为准;

优先级为:

同权重: 内联样式表（标签内部）> 嵌入样式表（当前文件中）> 外部样式表（外部文件中）。

!important > id > class > tag

important 比 内联优先级高

4.什么是HTML5？哪些浏览器支持HTML5?

5.DOCTYPE有何作用？如何定义HTML的DOCTYPE?

用来告知浏览器的解析器用什么文档标准解析这个文档

<!DOCTYPE HTML>

6.HTML5新增的一些常用标签

HTML5 现在已经不是 SGML 的子集，主要是关于图像，位置，存储，多任务等功能的增加。绘画 canvas;用于媒介回放的 video 和 audio 元素;本地离线存储 localStorage 长期存储数据，浏览器关闭后数据不丢失; sessionStorage 的数据在浏览器关闭后自动删除;语意化更好的内容元素，比如 article、footer、header、nav、section;表单控件，calendar、date、time、email、url、search; 新的技术webworker, websocket, Geolocation;

7.如何从HTML5本地存储中添加、移除数据？与cookie有何区别？

8.HTML5的Application Cache与老的缓存机制有什么不同？

9.什么是CSS3 Media Query？

10.CSS3关于制作动画的3个属性是什么？

11.如何检测一个变量的类型?

1.格式：typeof 变量

用法：if( typeof 变量 == "类型标识") { ... }

2.变量.constructor（构造函数法）

用法：if(变量.constructor == "类型标识符") { ... }

12.请写出CSS的几种引入方式.

1.行内样式

2.内嵌样式<style></style>

3.链接样式link

4.导入样式@import

优先级：就近原则

行内>内嵌>链接>导入

13.请写出两种css清除浮动方法.

1.伪元素清除浮动

.clearfix:after{

content:'';

display:block;

height:0;

visibility:hidden;

overflow:hidden;

clear:both;

}

2.bfc

overflow:hidden;

3.添加空盒子

.clear{

clear:both;

}

14.行内元素和块级元素分别有哪些？二者区别是什么？

1.行内元素有：a b span img input select strong（强调的语气）

2.块级元素有：div ul ol li dl dt dd h1 h2 h3 h4…p

3.常见的空元素：<br> <hr> <img> <input> <link> <meta>

4.鲜为人知的是：<area> <base> <col> <command> <embed> <keygen> <param> <source> <track> <wbr>

5.区别：

块级元素独占一行，默认情况下，其宽度自动填满其父元素宽度，行内不会独占一行，相邻行内元素会排列在同一行里，直到一行排不下，才会换行，其宽度随元素的内容而变化

15.如何获取select中option的value的值.

document.getElementById("select").value

16.canvas怎么使用

1.先在body里写一个<canvas id=‘canvqas’ width=‘400’ heihht=‘400’></canvas>标签

2.然后js获取canvas，var canvas = document.getElementById('canvas')

3.获取上下文环境 var context = canvas.getContext('2d')

context.fillStyle='#000'

context.fullRect(50,50,100,100)

### 17. display:none和visibility:hidden的区别？

display:none 隐藏对应的元素，在文档布局中不再给它分配空间，它各边的元素会合拢，就当他从来不存在。visibility:hidden 隐藏对应的元素，但是在文档布局中仍保留原来的空间。

### 18. CSS中 link 和@import 的区别是？

(1) link属于HTML标签，而@import是CSS提供的;

(2) 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再加载;

(3) import只在IE5以上才能识别，而link是HTML标签，无兼容问题; (4) link方式的样式的权重 高于@import的权重.

### 19. 为什么要初始化CSS样式。

因为浏览器的兼容问题，不同浏览器对有些标签的默认值是不同的，如果没对CSS初始化往往会出现浏览器之间的页面显示差异。当然，初始化样式会对SEO有一定的影响，但鱼和熊掌不可兼得，但力求影响最小的情况下初始化。

20.

二、JavaScript:

1.举例说明javascript中函数eval的作用.

它的功能是把对应的字符串解析成JS代码并运行；应该避免使用eval，不安全，非常耗性能（2次，一次解析成js语句，一次执行）。由JSON字符串转换为JSON对象的时候可以用eval，var obj =eval('('+ str +')');

2.什么是单例模式？写出其实现.

单例模式的含义是: 保证一个类只有一个实例,并提供一个访问它的全局访问点。实现的方法是:使用一个变量来标志当前是否已经为某个类创建过对象,如果创建了,则在下一次获取该类的实例时,直接返回之前创建的对象,否则就创建一个对象。这就确保了一个类只有一个实例对象。

var Singleton = function(name){

this.name = name;

// 使用instance 该标志来判断是否创建了一个实例

this.instance = null;

};

Singleton.prototype.getName = function(){

console.log(this.name);

};

Singleton.getInstance = function(name) {

if(!this.instance) {

this.instance = new Singleton(name);

}

return this.instance;

}

3.window.onload和jquery的ready有什么区别？

1.执行时间   
        window.onload必须等到页面内包括图片的所有元素加载完毕后才能执行。   
        $(document).ready()是DOM结构绘制完毕后就执行，不必等到加载完毕。   
2.编写个数不同  
         window.onload不能同时编写多个，如果有多个window.onload方法，只会执行一个   
         $(document).ready()可以同时编写多个，并且都可以得到执行  
3.简化写法   
         window.onload没有简化写法   
         $(document).ready(function(){})可以简写成$(function(){});

4.如何让规避javascript团队开发过程中出现的命名冲突的问题？

命名时遵循BEM命名规范，BEM的意思就是块（block）、元素（element）、修饰符（modifier）,是由Yandex团队提出的一种前端命名方法论。这种命名方法可以使项目的开发更加透明而且更有意义，使项目的开发更加严禁规范。

5.你了解哪些javascript库？

1.[jQuery](http://lib.csdn.net/base/jquery" \t "_blank" \o "jQuery知识库)库

2、[AngularJS](http://lib.csdn.net/base/angularjs" \o "AngularJS知识库" \t "_blank)企业级框架

3、[React](http://lib.csdn.net/base/react" \o "React知识库" \t "_blank)框架

4、vue未来的主流

6.你是如何理解闭包的？

我认为闭包就是能够读取其他函数内部变量的函数。由于在Javascript语言中，只有函数内部的子函数才能读取局部变量，因此可以把闭包简单理解成"定义在一个函数内部的函数"。所以，在本质上，闭包就是将函数内部和函数外部连接起来的一座桥梁。

7.请尽可能详细解释下Ajax的工作原理？

Ajax是一种支持异步请求的技术，XMLHttpRequest是ajax的核心机制，Ajax的原理简单来说通过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。

8.数组去重.

以下是数组去重的三种方法：

Array.prototype.unique1 = function () {

var n = []; //一个新的临时数组

for (var i = 0; i < this.length; i++) {//遍历当前数组

//如果当前数组的第i已经保存进了临时数组，那么跳过，

//否则把当前项push到临时数组里面

if (n.indexOf(this[i]) == -1) n.push(this[i]);

}

return n;

}

Array.prototype.unique2 = function(){

var n = {},r=[]; //n为hash表，r为临时数组

for(var i = 0; i < this.length; i++) { //遍历当前数组

if (!n[this[i]])｛ //如果hash表中没有当前项

n[this[i]] = true; //存入hash表

r.push(this[i]); //把当前数组的当前项push到临时数组里面

}

}

return r;

}

Array.prototype.unique3 = function(){

var n = [this[0]]; //结果数组

for(var i = 1; i < this.length; i++){ //从第二项开始遍历

//如果当前数组的第i项在当前数组中第一次出现的位置不是i，

//那么表示第i项是重复的，忽略掉。否则存入结果数组

if (this.indexOf(this[i]) == i) n.push(this[i]);

}

return n;

}

9.如何发送ajax请求后台数据.

10. DOM操作——怎样添加、移除、移动、复制、创建和查找节点。

（1）创建新节点

createDocumentFragment() //创建一个DOM片段

createElement() //创建一个具体的元素

createTextNode() //创建一个文本节点

（2）添加、移除、替换、插入

appendChild()

removeChild()

replaceChild()

insertBefore() //在已有的子节点前插入一个新的子节点

（3）查找

getElementsByTagName() //通过标签名称

getElementsByName() //通过元素的Name属性的值(IE容错能力较强，会得到一个数组，其中包括id等于name值的)

getElementById() //通过元素Id，唯一性

11. null和undefined的区别？

null是一个表示"无"的对象，转为数值时为0；undefined是一个表示"无"的原始值，转为数值时为NaN。

当声明的变量还未被初始化时，变量的默认值为undefined。  
null用来表示尚未存在的对象，常用来表示函数企图返回一个不存在的对象。

undefined表示"缺少值"，就是此处应该有一个值，但是还没有定义。典型用法是：

（1）变量被声明了，但没有赋值时，就等于undefined。

（2) 调用函数时，应该提供的参数没有提供，该参数等于undefined。

（3）对象没有赋值的属性，该属性的值为undefined。  
（4）函数没有返回值时，默认返回undefined。

null表示"没有对象"，即该处不应该有值。典型用法是：

（1） 作为函数的参数，表示该函数的参数不是对象。

（2） 作为对象原型链的终点。

三、jQuery:

1.jQuery选择器有哪些？

四、AngularJs:

五、Ajax:

1.提交表单请求时，post和get有哪些区别？

get方式：

1.数据是通过URL发送，显示在URL栏中

2.传递数据大小有限

3.字符集会被转化为iso-8859-1，一定要在服务端进行解码。

4.解决乱码方式：

①服务器解码

②使用过滤器

③修改服务器配置，tomcat URIEnconding=“utf8”（推荐）

post方式：

1.数据是通过请求头传输，不显示在地址栏中，相对而言比较安全

2.理论上说是无限

3.字符集可在请求头设置，服务器端自动识别

4.解决乱码方式：

①.服务器解码

②使用过滤器（推荐）

③客户端设置请求头

2. ajax过程

(1)创建XMLHttpRequest对象,也就是创建一个异步调用对象.

(2)创建一个新的HTTP请求,并指定该HTTP请求的方法、URL及验证信息.

(3)设置响应HTTP请求状态变化的函数.

(4)发送HTTP请求.

(5)获取异步调用返回的数据.

(6)使用JavaScript和DOM实现局部刷新.

六、其他：

1.浏览器兼容问题.

## 2. 你如何对网站的文件和资源进行优化？

文件合并

优化图片

文件最小化/文件压缩（css.js的压缩）

使用 CDN 托管

缓存的使用（多个域名来提供缓存）

3.跨域

参考：<https://segmentfault.com/a/1190000000718840>

4. 说说最近最流行的一些东西吧？常去哪些网站？

Node.js、VUE、npm、MVVM、angular、React 。

网站：w3cfuns,sf,hacknews,CSDN,慕课，博客园，InfoQ,w3cplus等

## 5. 请解释一下 JavaScript 的同源策略。

概念:同源策略是客户端脚本（尤其是Javascript）的重要的安全度量标准。它最早出自Netscape Navigator2.0，其目的是防止某个文档或脚本从多个不同源装载。

这里的同源策略指的是：协议，域名，端口相同，同源策略是一种安全协议。指一段脚本只能读取来自同一来源的窗口和文档的属性。

## 6. 什么是 "use strict"; ? 使用它的好处和坏处分别是什么？

ECMAscript 5添加了第二种运行模式："严格模式"（strict mode）。顾名思义，这种模式使得Javascript在更严格的条件下运行。

设立"严格模式"的目的，主要有以下几个：

- 消除Javascript语法的一些不合理、不严谨之处，减少一些怪异行为;

- 消除代码运行的一些不安全之处，保证代码运行的安全；

- 提高编译器效率，增加运行速度；

- 为未来新版本的Javascript做好铺垫。

注：经过测试IE6,7,8,9均不支持严格模式。

缺点：  
现在网站的JS 都会进行压缩，一些文件用了严格模式，而另一些没有。这时这些本来是严格模式的文件，被 merge 后，这个串就到了文件的中间，不仅没有指示严格模式，反而在压缩后浪费了字节。

## AMD和CMD 规范的区别？

1. JS原型链

参考：https://segmentfault.com/a/1190000000733959