# 前端面试知识点汇总

## HTML5

1. Doctype作用？严格模式与混杂模式如何区分？它们有何意义?

答：

（1）、<!DOCTYPE>声明位于HTML文档中的第一行，处于 <html> 标签之前。告知浏览器的解析器用什么文档标准解析这个文档。DOCTYPE不存在或格式不正确会导致文档以兼容模式呈现。

（2）、标准模式的排版 和JS运作模式都是以该浏览器支持的最高标准运行。在兼容模式中，页面以宽松的向后兼容的方式显示,模拟老式浏览器的行为以防止站点无法工作。

1. H5的新特性有哪些？移除了哪些？如何与 HTML 区分？

答：新增的主要是关于图像、位置、存储、多任务等功能。

Canvas、audio、video、localStorage、sessionStorage

语义化更好的标签：article/footer/header/nav/session

表单控件：calendar/date/time/email/url/search

新的技术：webworker/websocket/geolocation

移除的元素主要有：

纯表现的元素：basefont/big/center/font/s

对可用性产生负面影响的元素：frame/frameset/noframes

IE8/IE7/IE6支持通过document.createElement方法产生的标签，可以利用这一特性让这些浏览器支持HTML5新标签。

如何区分HTML5： DOCTYPE声明\新增的结构元素\功能元素。

1. 行内元素有哪些？块级元素有哪些？ 空(void)元素有那些？

答：CSS规范规定，每个元素都有display属性，且都有默认值。

（1）行内元素有：a label 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>

鲜为人知的是：

<area> <base> <col> <command> <embed> <keygen> <param> <source> <track> <wbr>

不同浏览器（版本）、HTML4（5）、CSS2等实际略有差异

1. iframe有那些缺点？

答：

阻塞主页面的onload事件；

搜索引擎的检索程序无法解读这种页面，不利于SEO；

和主页面共享连接池，而浏览器对相同域的连接有限制，所以会影响页面的并行加载；

如果需要使用iframe，最好是通过javascript动态给iframe添加src属性值，这样可以绕开以上两个问题。

1. 如何实现浏览器内多个标签页之间的通信? (阿里)

答：

WebSocket、ShardWorker

Localstorage、cookies等本地存储方式

Localstorage的原理: 它在另一个浏览上下文中被添加、修改或删除时，它都会触发一个事件。通过监听事件，控制它的值来进行页面通信。

## CSS3

1. 介绍一下标准的CSS的盒子模型？低版本IE的盒子模型有什么不同的？

答：IE 和 W3C 标准盒模型。主要有content/padding/margin/border

区别：IE的content部分把border和padding计算了进去。

1. Div 垂直居中的方法

答：绝对定位；绝对定位+transform；flex；

1. position的值relative和absolute定位原点是？

答：absolute：第一个不为static定位的父元素；relative：相对正常位置定位；fixed：相对于浏览器窗口定位；static：没有定位，正常文档流，会忽略 top/left/right/bottom/z-index 属性。

1. 经常遇到的浏览器的兼容性有哪些？原因，解决方法是什么，常用hack的技巧 ？

答：浏览器默认的margin和padding不同。解决方案是加一个全局的\*{margin:0;padding:0;}来统一。IE6双边距bug，使用“\9”、“+”、“\_”区分IE678浏览器；IE下获取获取自定义属性的办法，getAttribute；

事件：e = event || window.event;

坐标：IE下，event对象有x,y，而没有pageX、pageY，火狐下则相反，mX = event.x ? event.x : event.pageX;

触发事件的DOM：在IE中是window.event.srcElement，在Firefox中是event.target，而在Opera中则两者都支持。

IE和火狐下的兼容性：

使用obj.parentNode/document.formName.item(“itemname”)，使用[]获取集合类对象；

1. 对BFC规范(块级格式化上下文：block formatting context)的理解？

答：W3C CSS2.1 规范的一个概念，它是一个独立容器，决定了元素如何对其内容进行定位，以及与其他元素的关系和相互作用。

一个页面是由很多个 Box 组成的，元素的类型和 display 属性，决定了 Box的类型。

不同类型的 Box，会参与不同的 Formatting Context ，因此Box内的元素会以不同的方式渲染，也就是说 **BFC 内部的元素和外部的元素不会相互影响。**

BFC是block formatting context，也就是块级格式化上下文，是用于布局块级盒子的一块渲染区域

满足下列条件之一就可触发BFC

　　【1】根元素，即HTML元素

　　【2】float的值不为none

　　【3】overflow的值不为visible

　　【4】display的值为inline-block、table-cell、table-caption

　　【5】position的值为absolute或fixed

作用

【1】可以阻止元素被浮动元素覆盖

【2】可以包含浮动元素

【3】属于同一个BFC的两个相邻块级子元素的上下margin会发生重叠

1. 浏览器是怎样解析CSS选择器的？

答：样式系统从关键选择器开始匹配，然后左移查找规则选择器的祖先元素。

只要选择器的子树一直在工作，样式系统就会持续左移，直到和规则匹配，或者是因为不匹配而放弃该规则。

1. rem布局的优缺点
2. CSS3 动画——transform/transition/animate
3. Flex 布局，及适用场景

答：可以把列表放在同一个方向（从上到下排列，从左到右），并让列表能延伸到占用可用的空间。它的所有子元素自动成为容器成员，称为Flex项目（flex item），简称"项目"。常规布局是基于块和内联流方向，而Flex布局是基于flex-flow流可以很方便的用来做局中，能对不同屏幕大小自适应。在布局上有了比以前更加灵活的空间。

1. 为什么要清除浮动？清除方法有哪些？

答：浮动元素的父容器高度塌陷；

解决办法：

* 父元素定义高度；
* 父元素设置 overflow：hidden；
* 父元素浮动；
* .clearfix 类：

.clearfix::before,.clearfix:after{

Display:block;

Content: “.”;

Visibility:hidden;

Height:0;

Clear:both;

\*zoom:1; //触发 IE layout

}

## JavaScript

1. String 对象的 api
2. Array 对象的 api
3. Object 对象的 api
4. 介绍JavaScript的基本数据类型。

答：7种基本类型：Undefined、Null、String、Number、Boolean、Object、Symbol

1. JavaScript原型，原型链 ? 有什么特点？

答：每个构造函数都有一个原型对象，原型对象都包含一个指向构造函数的指针，而实例都包含一个指向原型对象的内部指针。当访问一个对象属性时，如果这个对象内部找不到，就会去prototype里找这个属性，而prototype里会有自己的prototype属性，于是一直找下去就是我们所说的原型链。

特点：JavaScript 对象是通过引用来传递的，我们创建的每个新对象实例中并没有一份属于自己的原型副本。当我们修改原型时，与之相关的对象实例也会继承这一改变。

1. JavaScript如何实现继承？如何使用构造函数实现？

答：使用原型链实现继承。

构造函数也可实现：在子类的构造函数中调用超类的构造函数。

构造函数继承的五种方法：

1. 构造函数

function super(){}

function sub(){

super.apply(this,arguments);

}

1. Prototype模式

Sub.prototype = new super();

Sub.prototype.constructor = sub;

1. 直接继承prototype

Sub.prototype = super.prototype;

Sub.prototype.constructor = sub;

1. 利用空对象作中介

function F(){}

F.prototype = Super.prototype;

Sub.prototype = new F();

Sub.prototype.constructor = sub;

空对象不占内存。

1. 拷贝继承

实现拷贝函数

非构造函数实现继承：

1）Object（）方法

function object(o){

function F(){}

F.prototype = o;

return new F();

}

2）浅拷贝

1. 什么是闭包（closure），为什么要用它？

答：闭包就是有权访问另一个函数作用域中的变量的函数。创建闭包最常用的方式就是在一个函数内创建另一个函数，通过另一个函数访问这个函数的局部变量，利用闭包可以突破作用域链，将函数内部的变量和方法传递到外部。

闭包的特性：

* 函数内再嵌套函数；
* 内部函数可以引用外部函数的变量和参数；
* 参数和变量不会被垃圾回收机制回收。

1. 写一个通用的事件侦听器函数(机试题)。

答：见 EventUtils.js

1. 关于事件，IE与火狐的事件机制有什么区别？ 如何阻止冒泡？

答：用户操作可以被 JavaScript 侦测到的行为。

事件处理机制：IE是事件冒泡；Firefox 同时支持捕获和冒泡。

阻止冒泡：e.stopPropagation() || e.cancelBubble = false;

1. Ajax 是什么? 如何创建一个Ajax？

答：见 Ajax.js

1. 请介绍一下JS之事件节流（throttle）？

答：函数节流：一个函数执行一次后，只有大于设定的执行周期后才会执行第二次。

函数节流的应用场景是需要间隔一定时间触发回调来控制函数调用频率，如下：

* DOM 元素的拖拽功能实现（mousemove）
* 搜索联想（keyup）
* 计算鼠标移动的距离（mousemove）
* Canvas 模拟画板功能（mousemove）
* 射击游戏的 mousedown/keydown 事件（单位时间只能发射一颗子弹）

例如：监听滚动事件判断是否到页面底部自动加载更多：给 scroll 加了 debounce 后，只有用户停止滚动后，才会判断是否到了页面底部；如果是 throttle 的话，只要页面滚动就会间隔一段时间判断一次。

Throttle 函数示例：见 ThrottleAndDebounce.js

1. 什么是JS的函数防抖（debounce）？

答：防抖函数：一个需要频繁触发的函数，在规定时间内，只让最后一次生效，前面的不生效。

函数防抖的应用场景为，对于连续的事件响应我们只需要执行一次回调，有以下场景：

* 每次 resize/scroll 触发统计事件
* 本输入的验证（连续输入文字后发送 AJAX 请求进行验证，验证一次就好）

## ES6

1. Promise 的实现原理
2. Class 和 Module
3. Symbol
4. Reflect

## jQuery

1. JQuery的源码看过吗？能不能简单概况一下它的实现原理？
2. jQuery 的属性拷贝(extend)的实现原理是什么，如何实现深拷贝？
3. 谈一下Jquery中的bind(),live(),delegate(),on()的区别？
4. Sizzle 引擎的实现原理？
5. 事件绑定和事件循环？

## 浏览器

1. 常见的浏览器内核有哪些？
2. 答：Trident内核：IE,MaxThon,TT,The World,360,搜狗浏览器等。[又称MSHTML]
   1. Gecko内核：Netscape6及以上版本，FF,MozillaSuite/SeaMonkey等
   2. Presto内核：Opera7及以上。 [Opera内核原为：Presto，现为：Blink;]
   3. Webkit内核：Safari,Chrome等。 [ Chrome的：Blink（WebKit的分支）]
3. 请简述浏览器加载URL的过程
4. 如何解决浏览器跨域问题？
5. 如何避免XSS和CSRF攻击？

## 前端框架

1. 请简述 Vue 生命周期
2. Vue 父子组件的通信方式有哪几种？
3. Vue-Router的工作原理
4. 如何使用Vuex进行跨组件状态管理
5. 服务端渲染（如Vue SSR），首屏直出当然是最理想的方案。

## 前端工程化，构建、打包

1. Webpack 常用的配置项
2. Webpack热更新实现原理?
3. css-loader 的原理

## 简历项目相关

1.实现左侧题型列表的点击、拖拽事件以及调整题目顺序的拖拽事件

mousemove/mouseup/mousedown

e = e || window.event;

x = e.clientX, y = e.clientY;

\_showMoveDom(e)

2.兼容 mui 关于 WebView 页面管理的部分接口，使业务模块开发人员可以不用关心接口实现，使用统一的接口完成 APP 和普通浏览器中的兼容；实现移动端引导页、节日欢迎页及首页的预加载和缓存机制；

3.本人还设计了第三方系统接入久其格格的单点登录系统和权限校验系统。

4.Webpack 常用的打包命令及常用配置项。

## 算法

## 计算机基础