Web前端面试题

[Html+css部分 1](#_Toc481512465)

[JavaScript+jQuery部分 10](#_Toc481512466)

1. 如何去除li列表项样式

答： list-style：none

1. 网页默认字体大小

答：网页默认字体16px

1. 实现首行缩进两个字符

答：text-indent：2em

1. 常见的字体

答：

1. 中文字体：宋体（Sim-Sun）、黑体（SimHei）、微软雅黑（Microsoft YaHei）
2. 英文字体：Arial、sans-serif
3. 添加下划线

答：

text-decoration:underline

text-decoration: line-through;

text-decoration: overline;

1. 显示属性
2. 列举display常见的属性值及其作用

答：

1. display常见属性值：inline、block、inline-block、none
2. display不常见的属性值：table、list-item、table-cell、flex
3. display的各个显示类型中那些可以设置宽高

答：

1. display：block
2. display：inline-block
3. display：table
4. display：table-cell

行内元素也称为内联元素

1. display：none与visibility：hidden的区别

答：

1. display：none与visibility：hidden都可以实现隐藏一个块
2. 设置display：none的元素不占物理空间，设置visibility：hidden的元素只是视觉上不可视，但是依然保留物理空间
3. 标签属性
4. 常见的块级元素与行内元素\*

答：

1. 常见的块元素：div、p、h1-h6、dl、dt、dd、ul、li、form
2. 常见的行元素：span、a、img、strong、em、b、i、br
3. 块级元素和行内元素的区别

答：

1. 块元素独占一行，行元素并排显示
2. 块元素设置宽高有效，行元素宽高由内容撑开
3. 块元素margin和padding设置都有效，行元素只有横向有效
4. 通过display属性可以实现互转
5. 块元素可包含块元素和行元素，行元素不能包含块元素

**注意：p标签中不能包含div标签**

1. img标签的title和alt属性的区别

答：

1. 含义：alt表示图片不存在或加载失败的时候替代文本，title表示图片的说明信息（鼠标移动到图片上面显示的信息）
2. 浏览器：当img设置alt和title两种属性时，鼠标移动到图片会显示title，在IE6、IE7中鼠标移入图片时会显示alt的内容
3. SEO：alt有利于SEO，title通常放置对图片的进一步说明

SEO(搜索引擎优化、常见的搜索引擎 百度 搜狗 谷歌 搜搜 搜狐 有道)

1. 如何在新窗口中打开页面

答：target：\_blank

1. a标签的四个伪类选择器及书写顺序和原因

答：

1. a标签伪类有：link、：visited、：hover、：active
2. ：link a标签还未被访问的状态
3. ：visited a标签已被访问过的状态
4. ：hover 鼠标悬停在a标签上的状态
5. ：active 鼠标悬停在a标签，并按下时的状态

如果你做的网页兼容ie6、7、8

1. 什么是语义化标签 \*

答：

1. html语义化让页面的内容结构化，结构更清晰，便于对浏览器、搜索引擎解析;
2. 即使在没有样式CSS情况下也以一种文档格式显示，并且是容易阅读的;
3. 搜索引擎的爬虫也依赖于HTML标记来确定上下文和各个关键字的权重，利于SEO;
4. 使阅读源代码的人对网站更容易将网站分块，便于阅读维护理解。
5. 定位和浮动
6. 如何让元素固定在窗口底部

答：position：fixed；bottom：0

1. Position定位属性有哪些

答：position：static、relative、absolute、fixed、inherit（继承）

1. 绝对定位和相对定位的区别

答：

1. 相对定位不会脱离文档流，绝对定位会脱离文档流
2. 绝对定位的元素相互层叠，相对定位的元素不会
3. 绝对定位的元素相对于static以外的第一个父级进行定位，相对定位相对于其正常位置进行定位
4. 如何理解和应用z-index

答：

1. z-index设置为absolute元素的堆叠顺序，值可以为负
2. z-index值高的元素将处于较低元素的前面
3. **\*** z-index只针对同级的标签有效，即子标签的z-index值对父级标签是无效的，因为两者的级别不同z-index是无法进行比较的
4. 浮动对父级高度以及布局有哪些影响？

答：

1. 浮动能实现多个块级元素处在同一行
2. 子级浮动会使父级出现高度塌陷问题
3. 浮动元素会对后面的兄弟元素造成影响
4. 清除浮动的基本方法

答：

1. 使用overflow：hidden清除浮动
2. 使用空标签清除浮动
3. 使用br标签清除浮动
4. 利用伪类after清除浮动
5. 让父级发生浮动，实现清除浮动
6. 给父级设置高度
7. 如何让两个div并排显示

答：

1. 利用浮动布局，为两个div设置浮动
2. 采用定位布局，为两个div设置定位
3. 如何实现元素水平垂直居中

答：

1. 通过text-align与line-height来实现
2. 利用父级元素的内边距
3. 使用定位
4. 利用display：table与table-cell转换元素展示类型，并通过vertical-align：middle来实现
5. 盒模型
6. 盒模型的基本组成有哪些？

答：width、height、margin、padding、border

1. margin和padding的区别及布局时的选用

答：

1. margin :外边距，指元素与元素之间的距离
2. padding：内边距，指元素内容与边框之间的距离
3. 在布局的选用方法：父子级之间的空隙，优先选择使用父级的padding实现，兄弟级之间的空隙，优先使用margin实现
4. border：0和border：none的区别

答：

1. 性能差异：border：0；虽然在页面上无法看到边框，但浏览器依然渲染border-width、border-color属性。border ：0；浏览器解析none时将不会渲染，不会消耗内存
2. 兼容性差异：border：none对IE6/7无效，边框依然存在；border：0所有浏览器都一致把边框隐藏
3. 如何使一个块级元素在父级居中显示

答：margin：0px auto

1. px和em的区别是什么？

答：

1. px :像素，绝对度量单位
2. em ：字符，相对度量单位，相对于当前对象内文本的字体大小
3. 网页文档声明的作用

答：

1. 在定义HTML文档之后，需要进行基本类型的声明
2. 文档声明会使浏览器使用相应标准加载网页并显示；若不声明，浏览器将无法获知HTML文档类型，在IE6下会触发怪异解析
3. 浏览器
4. IE hack技术有哪些？\*

答：

1. \_width:200px; /\*IE6\*/
2. +width:200px; /\*IE6、IE7\*/
3. \*width:200px; /\*IE6、IE7\*/
4. width:200px\9; /\*IE6~IE10\*/
5. width:200px\0 /\*IE8~IE11\*/
6. 浏览器内核前缀

答：

1. IE浏览器 –Trident -ms
2. 谷歌、苹果浏览器 – webkit
3. 欧朋浏览器 -Presto -o
4. 火狐浏览器 Gecko -moz
5. 使用条件注释判断浏览器 \*

答：

<!--[if 属性 IE 版本号]>

具体HTML代码

<![endif]-->

属性值：gt 大于、lt 小于 、gte 大于等于、lte小于等于、！排除

1. 其他
2. css选择器的优先级以及计算方法

答：ID选择器 > class类选择器 > 标签选择器

ID：100

class:10

element:1

1. 如何使用选择器选择多个li的第一个元素

答：li:first-of-type或者li:nth-of-type(1)

1. link和@import的区别

答：

1. @import只能加载CSS，link除了加载CSS外，还定义RSS等其他业务
2. link 引用css时，在页面载入时同时加载，@import需要页面网页完全载入后加载
3. link 无兼容问题，而@import低版本的浏览器不支持
4. link 支持使用javascript控制DOM去改变样式，而@import不支持
5. css的引入方式

答：

1. 外部引入
2. 内部引入
3. 行内引入
4. html5和css3
5. css3新属性有哪些？\*

答：

1. 与响应式布局有关的属性：CSS3 Media Query（媒体查询）
2. 较为基础的CSS3功能和属性：圆角边框、盒阴影、文本阴影
3. 特效属性：变形、动画、过渡
4. 不常用的：倒影、蒙版、渐变
5. content-box与border-box的差别

答:content-box与border-box均为box-sizing的属性值，区别是两者的盒子宽度是否包含边框和内边距

1. 常见的HTML5新增标签以及应用方法\*

答：

1. article 标签定义外部内容
2. aside 定义页面内容之外的内容。aside内容与article内容相关
3. section 定义文档中的节（如章节、页眉、页脚）
4. header 定义头部
5. hgroup 定义一个标题集合
6. footer 定义底部
7. nav 定义页面导航链接
8. audio定义音频
9. video 定义视频
10. figure 定义一组媒介内容的分组以及他们的标题
11. API与框架
12. cookies、sessionStorage和localStorage的区别

答：

1. cookie 、sessionStorage、localStorage都是用于存储数据
2. cookie 是在浏览器和服务器间来回传递的，而sessionStorage与localStorage均属于本地存储
3. cookie的存储大小比较小，而其他两个存储要比cookie大的多
4. sessionStorage是临时性存储，localStorage是永久性存储，cookie可以设置存储的有效期
5. video和audio的区别

答：

1. 视频是由视频编码+音频编码共同组成，音频只存在音频编码
2. 对于视频和音频的API接口，两者基本相同，但是视频会多一个poster等一些关于封面图的接口
3. 移动端
4. 如何理解rem以及rem和em的区别

答：

1. rem是相对度量单位
2. rem相对于根元素的字体大小来计算
3. em是相对于父元素的字体大小的单位
4. 移动端开发中，如何选用rem、em、%

答：

1. 通常横向布局选用%，纵向布局选用rem
2. 横向布局当中，部分情况也可以使用rem或em
3. 基准字体大小设置

答：

1. 320像素移动端效果图当中，最小字体为12px
2. 720px的设计图字体大小为：12/320\*720=27~~28

JavaScript+jQuery部分

1. 事件
2. 简述事件委托的原理与使用

答：

1. 主要利用了事件冒泡，只指定一个事件处理程序，就可以管理一个类型的所有事件
2. 给元素的父级或者祖级，甚至页面绑定事件，然后利用事件冒泡的基本原理，通过事件目标对象进行检测，然后执行相关操作

优势

1. 大大减少了实践处理程序的数量，降低事件处理程序的时间
2. 整个页面占用的内存空间会更少，从而提高整体的性能
3. 只要元素在页面上呈现出来，他就立刻具备了相应的功能，无需等待
4. 事件冒泡机制

答：

1. 事件流：描述的是页面元素接收事件的顺序
2. 事件冒泡：事件开始的时候由最具体的元素接收，然后逐级向下传播到具体的节点
3. 事件捕获：事件开始的时候由最不具体的节点接收，然后逐级向下传播到具体的节点
4. 如何组织点击a链接时的跳转

答：阻止默认事件e.preventDefault()

1. IE与火狐的事件机制方面的兼容问题

答：

1. 事件绑定：addEventListener与attachEvent
2. 事件移除：removeEventListener与detachEvent
3. 事件对象：e与window.event
4. 获取事件目标对象：e.target与window.event.srcElement
5. 阻止冒泡：e.stopPropagation与window.event.cancelBubble
6. 阻止默认：e.preventDefault与window.event.returnValue

以上前者是火狐事件机制，后者是IE事件机制

1. 作用域
2. call()和apply()的区别

答：

1. 两者均已经用于改变函数的this指向
2. 两种方法的第一个参数均用于定义this的新指向，第二个参数开始往后的所有参数，均用于书写当前函数的参数
3. 在参数的书写方式方面，两者具有一定的区别。Call方法必须一个参数一个参数的进行书写，而apply方法可以使用数组的方式来罗列参数，更加方便
4. 函数
5. 获取函数的实参个数

答：在函数当中输出“函数名.arguments.length”即可

1. 如何实现点击当前元素输出其索引值

答：点击li输出当前index

1. eval的作用以及特点

答：eval()函数中的参数为字符串，该方法能够让字符串中的JavaScript代码运行

1. 内置对象和宿主对象的区别

答：

1. 内置对象：Global（全局）对象、Math对象

全局对象包括方法、属性、构造函数等

1. 方法：isNaN()、parseInt()、parseFloat()、eval()
2. 属性：undefined、NaN、infinity
3. 构造函数：Array、Object、Function、Boolean、String、Number、Date等
4. window对象：所有在全局作用域下声明的变量和函数，都是window对象（也是Global对象）的属性和方法

非本地对象都是宿主对象，即由ECMAScript实现宿主环境提供的对象

1. JS基础
2. JavaScript中运算符的优先级

答： 逻辑非>算数 > 关系 > 逻辑与、或 > 条件 > 赋值

注意：小括号可以提升优先级

乘除取余三种优先级要大于加减

1. 实现函数中两个实参值得交换

答：function fn(a,b){

//在这里写一段代码，让a和，b的值能够交换 不允许借助其他变量

a=a-b

}

1. ==和===的区别

答：

1. ==表示“等于”，===表示“全等”
2. 两者均属于比较运算符，“==”只是比较数值，“===”比较数据以及数据类型
3. JavaScript的三种弹窗

答：

1. alert() 警告弹窗
2. confirm() 确认弹窗
3. prompt()对话弹窗
4. JavaScript的基本数据类型

答：Number、String、Boolean、Object、Undefined、Null

1. JS数据中那些属于引用数据类型

答：数值、对象

1. 三元条件语句的使用方法

答：

1. 三元操作符为 ？：
2. 基本语法“判断表达式 ？ 表达式2 ：表达式3”
3. DOM操作
4. 节点增删改查的方法

答：

1. 查：querySelector、querySelectorAll、getElementById、getElementByTagName、parentNode、children、childNodes、firstChild、lastChild、nextSibling、preveiousSibling
2. 增：appendChild、insertBefore
3. 删：removeChild
4. 改：replaceChild
5. 克隆：cloneNode
6. 列举改标标签内容的JS方法

答：

1. innerHTML
2. outerHTML
3. innerText
4. outerText
5. 列举原生JS获取标签的方法

答：

1. getElementById()
2. getElementsByName()
3. getElementsByTagName()
4. getElementsByClassName()
5. querySelector()
6. querySelectorAll()
7. 如何隐藏一个元素不显示

答：

1. 将display属性设置为none
2. 将opacity属性设置成0
3. 将visibility属性设置成hidden
4. 原生JS创建标签的写法

答：

1. 通过createElement()方法来创建具体的标签
2. 通过createTextNode方法创建文本节点
3. 列举改变元素样式的方法

答：

1. 通过style属性进行定义
2. 通过className属性定义
3. AJAX
4. JSON对象与字符串的相互转换方法

答：

1. JSON.stringify() 用于将JSON对象转换为字符串
2. JSON.parse() 用于将字符串转JSON对象
3. JSON 与JSONP的区别

答：

1. JSON是一种数据格式，用于在前端与后台之间传递信息
2. JSONP是一种跨域技术
3. 网络中常见的状态码有哪些？

答:

1. 200：服务器已成功处理了请求并提供了请求的网页
2. 204：服务器成功处理了请求，但没有返回任何内容
3. 301：请求的网页已永久移动到新位置。当URL发生变化时，使用301代码。搜索引擎索引中保存新的URL
4. 400：由于包含语法错误，当前请求无法被服务器理解
5. 403：请求失败，请求所希望得到的资源未被在服务器上发现
6. 404：请求的内容无法得到响应
7. 500：服务器遇到了一个未曾预料的状况，导致他无法完成请求的处理
8. 503：由于临时的服务器维护或者过载，服务器当前无法处理请求
9. AJAX的含义及工作原理

答：

1. 创建请求：var xhr=new XMLHttpRequest()
2. 确认发送方式和地址： xhr.open(“get”,url,true)
3. 发送请求：xhr.send(null)
4. 确认后台加载：xhr.onload=function(){}
5. 请求返回的数据：xhr.responseText
6. 同步与异步的区别

答：

1. 同步加载，每次刷新的是整个页面
2. 异步加载，每次只刷新需要更换部分的内容

异步加载的优势

1. 浏览器可以从服务器同时请求多项内容
2. 浏览器请求返回的速度会快的多
3. 只有页面中真正改变的部分得到更新
4. 能够减少服务器数据流量
5. 用户可以在页面更新的同时继续工作
6. 有些改变无须服务器往返通信就可以处理
7. get与post有何区别

答：get方式

1. 更长用，更方便
2. 性能好
3. 明文发送数据，没有post安全
4. 传输数据大小有限制

post方式

1. 使用相对较少
2. 性能只有get的1/3
3. 比get稍微安全一点
4. 没有传输数据大小限制
5. jQuery
6. jQuery中bind()、live()、delegate()的区别

答：

1. bind最简单的绑定方法
2. live可以对未生成的元素进行绑定，运用DOM检索比对
3. delegate可以对未生成的元素进行绑定，运用缓存机制
4. live方法存在于JQ的那个版本中

答：live在JQ1.7中不在建议使用，在1.9中删除

1. 简述JQ中end()方法

答：end()方法结束当前链条中的最近筛选操作，并将匹配元素集还原为之前的状态

1. JQ获取元素src属性的方法有哪些

答：元素.attr(‘src’)

1. JQ书写AJAX实例

答：

$.ajax({

url:’ ‘,

data:{},

type:’get’,

dataType:’json’,

success:function(res){},

error:function(res){}

})

1. Window.onload与jQuery的ready()方法有何区别

答：

1. 加载机制不同
2. 函数 定义次数不同
3. $(document).ready()存在缩写。书写起来更方便$(function(){})
4. remove()何detach()有什么区别

答：

1. remove、detach方法均用于移除备选元素，包括所有文本和子节点
2. remove、detach方法都不会把匹配的元素从JQ对象中删除，因而可以在将来再使用这些匹配的元素
3. remove()方法不会保留元素的JQ数据，其他的比如绑定事件。附加的数据等都会被移除，而detach()方法会保留所有的绑定事件、附加数据
4. parent()和parents()有什么区别

答：

1. parent方法获取到的是当前元素的父元素
2. parents方法获取到的是当前元素的祖先元素
3. HTTP和HTTPS有什么区别

答：

1. 使用的端口不同
2. https比http更安全
3. 请简单阐述一个网站向服务器发请求到请求到数据的过程

答:

1. DNS解析
2. 建立连接
3. 服务器受理请求
4. 返回数据后断开连接
5. 优雅降级和渐进增强的含义与特点

答：

1. 渐进增强：针对低版本浏览器进行构建页面，保证最基本的功能，然后在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
2. 优雅降级：一开始就构建完整的功能，然后在针对低版本的浏览器进行兼容