

2024前端高频—HTML篇

1.什么是 DOCTYPE，有何作用？

Doctype 是HTML5的文档声明，通过它可以告诉浏览器，使用哪一个HTML版本标准解析文档。在浏览器发展的过程中，HTML出现过很多版本，不同的版本之间格式书写上略有差异。如果没有事先告诉浏览器，那么浏览器就不知道文档解析标准是什么？此时，大部分浏览器将开启最大兼容模式来解析网页，我们一般称为 **怪异模式**，这不仅会降低解析效率，而且会在解析过程中产生一些难以预知的 **bug**，所以文档声明是必须的。

2.说说对 html 语义化的理解

HTML标签的语义化，简单来说，就是用正确的标签做正确的事情，给某块内容用上一个最恰当最合适的标签，使页面有良好的结构，页面元素有含义，无论是谁都能看懂这块内容是什么。

语义化的优点如下：

- 在没有CSS样式情况下也能够让页面呈现出清晰的结构
- 有利于SEO和搜索引擎建立良好的沟通，有助于爬虫抓取更多的有效信息，爬虫是依赖于标签来确定上下文和各个关键字的权重
- 方便团队开发和维护，语义化更具可读性，遵循W3C标准的团队都遵循这个标准，可以减少差异化

3.前端页面有哪三层构成，分别是什么？

构成：**结构层**、**表示层**、**行为层**

1. 结构层 (structural layer)

结构层类似于盖房子需要打地基以及房子的悬梁框架，它是由HTML超文本标记语言来创建的，也就是页面中的各种标签，在结构层中保存了用户可以看到的所有内容，比如说：一段文字、一张图片、一段视频等等

2. 表示层 (presentation layer)

表示层是由CSS负责创建，它的作用是如何显示有关内容，学名：**层叠样式表**，也就相当于装修房子，看你要什么风格的，田园的、中式的、地中海的，总之CSS都能办妥

3. 行为层 (behaviorlayer)

行为层表示网页内容跟用户之间产生交互性，简单来说就是用户操作了网页，网页给用户一个反馈，这是 `JavaScript` 和 `DOM` 主宰的领域

4. 什么是严格模式与混杂模式？

- 严格模式：是以浏览器支持的最高标准运行
- 混杂模式：页面以宽松向下兼容的方式显示，模拟老式浏览器的行为

5. 对于Web标准以及W3C的理解

`Web标准` 简单来说可以分为结构、表现、行为。其中结构是由HTML各种标签组成，简单来说就是body里面写入标签是为了页面的结构。表现指的是CSS层叠样式表，通过CSS可以让我们的页面结构标签更具美感。行为指的是页面和用户具有一定的交互，这部分主要由JS组成

`W3C`，全称：world wide web consortium是一个制定各种标准的非盈利性组织，也叫万维网联盟，标准包括HTML、CSS、ECMAScript等等，web标准的制定有很多好处，比如说：

- 可以统一开发流程，统一使用标准化开发工具（VSCode、WebStorm、Sublime），方便多人协作
- 学习成本降低，只需要学习标准就行，否则就要学习各个浏览器厂商标准
- 跨平台，方便迁移都不同设备
- 降低代码维护成本

6. 行级元素和块级元素分别有哪些及怎么转换？

常见的块级元素：`p`、`div`、`form`、`ul`、`li`、`ol`、`table`、`h1`、`h2`、`h3`、`h4`、`h5`、`h6`、`dl`、`dt`、`dd`

常见的行级元素：`span`、`a`、`img`、`button`、`input`、`select`

块级元素：

- 总是在新行上开始，就是每个块级元素独占一行，默认从上到下排列
- 宽度缺少时是它的容器的100%，除非设置一个宽度
- 高度、行高以及外边距和内边距都是可以设置的

- 块级元素可以容纳其它行级元素和块级元素

行内元素：

- 和其它元素都会在一行显示
- 高、行高以及外边距和内边距可以设置
- 宽度就是文字或者图片的宽度，不能改变
- 行级元素只能容纳文本或者其它行内元素

使用行内元素需要注意的是：

- 行内元素设置宽度 `width` 无效
- 行内元素设置 `height` 无效，但是可以通过 `line-height` 来设置
- 设置 `margin` 只有左右有效，上下无效
- 设置 `padding` 只有左右有效，上下无效

可以通过 `display` 属性对行内元素和块级元素进行切换(主要看第 `2、3、4` 三个值)：

7.H5有哪些新元素和新特性

HTML5主要是关于图像、位置、存储、多任务等功能的增加：

- 语义化标签，如：article、footer、header、nav等
- 视频video、音频audio
- 画布canvas
- 表单控件，calendar、date、time、email
- 地理
- 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
- 拖拽释放

8.如何使用HTML5中的Canvas元素绘制图形？

Canvas元素允许在网页上使用JavaScript绘制图形和动画。以下是一个简单的绘制矩形的示例：

```
<canvas id="myCanvas" width="200" height="200"></canvas>
<script>
  var canvas = document.getElementById("myCanvas")
  var ctx = canvas.getContext("2d")
  ctx.fillStyle = "red"
  ctx.fillRect(50, 50, 100, 100)
</script>
```

在这个示例中，使用 `document.getElementById()` 方法获取 `Canvas` 元素，并通过 `getContext("2d")` 获取2D绘图上下文。然后，使用 `fillStyle` 属性设置填充颜色，`fillRect()` 方法绘制一个矩形。

9.cookie、sessionStorage 和 localStorage 的区别

	cookie	localStorage	sessionStorage
由谁初始化	客户端或服务器，服务器可以使用 <code>Set-Cookie</code> 请求头。	客户端	客户端
过期时间	手动设置	永不过期	当前页面关闭时
在当前浏览器会话（browser sessions）中是否保持不变	取决于是否设置了过期时间	是	否
是否随着每个 HTTP 请求发送给服务器	是，Cookies 会通过 <code>Cookie</code> 请求头，自动发送给服务器	否	否
容量（每个域名）	4kb	5MB	5MB
访问权限	任意窗口	任意窗口	当前页面窗口

10.script、script async 和 script defer 的区别

- `<script>` \- HTML 解析中断，脚本被提取并立即执行。执行结束后，HTML 解析继续。

- `<script async>` \- 脚本的提取、执行的过程与 HTML 解析过程并行，脚本执行完毕可能在 HTML 解析完毕之前。当脚本与页面上其他脚本独立时，可以使用 `async`，比如用作页面统计分析。
- `<script defer>` \- 脚本仅提取过程与 HTML 解析过程并行，脚本的执行将在 HTML 解析完毕后进行。如果有多个含 `defer` 的脚本，脚本的执行顺序将按照在 document 中出现的位置，从上到下顺序执行。

注意：没有 `src` 属性的脚本，`async` 和 `defer` 属性会被忽略。

11.iframe的作用以及优缺点

优点

- 可以在页面上独立显示一个页面或者内容,不会与页面其他元素产生冲突。
- 可以在多个页面中重用同一个页面或者内容,可以减少代码的冗余。
- 加载是异步的,页面可以在不等待 iframe 加载完成的情况下进行展示。
- 方便地实现跨域访问

缺点

- 搜索引擎可能无法正确解析 iframe 中的内容
- 会阻塞主页面的 onload 事件
- 和主页面共享连接池,影响页面并行加载

12.Quirks（怪癖）模式是什么？它和Standards（标准）有什么区别？

页面如果写了DTD，就意味着这个页面采用对CSS支持更好的布局，而如果没有，则采用兼容之前的布局方式，这就是Quirks模式，有时候也叫怪癖模式、诡异模式、怪异模式。

区别：总体会有布局、样式解析、脚本执行三个方面区别，这里列举一些比较常见的区别：

- **盒模型**：在W3C标准中，如果设置一个元素的宽度和高度，指的是元素内容的宽度和高度，然而在Quirks模式下，IE的宽度和高度还包含了padding和border

- **设置行内元素的高宽**：在Standards模式下，给行内元素设置width和height都不会生效，而在Quirks模式下会生效
- **用margin: 0 auto设置水平居中**：在Standards模式下，设置margin: 0 auto；可以使元素水平居中，但是在Quirks模式下失效
- **设置百分比高度**：在Standards模式下，元素的高度是由包含的内容决定的，如果父元素没有设置百分比的高度，子元素设置百分比的高度是无效的

13.meta viewport 是做什么用的，怎么写？

Viewport，适配移动端，可以控制视口的大小和比例：

- `<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">`

其中，content 参数有以下几种：

- width viewport：宽度(数值/device-width)
- height viewport：高度(数值/device-height)
- initial-scale：初始缩放比例
- maximum-scale：最大缩放比例
- minimum-scale：最小缩放比例
- user-scalable：是否允许用户缩放(yes/no)