**面试题-HTML5+CSS3(前端常见面试题带答案 )**

**HTML5+CSS3常见面试题目**

**1.**[**SGML（标准通用标记语言）和HTML**](http://blog.jobbole.com/78346/#q2)**（超文本标记语言），XML（可扩展标记语言）和HTML的之间有什么关系？**

SGML（标准通用标记语言）是一个标准，告诉我们怎么去指定文档标记。他是只描述文档标记应该是怎么样的元语言，HTML是被用SGML描述的标记语言。

因此利用SGML创建了HTML参照和必须共同遵守的DTD，你会经常在HTML页面的头部发现“DOCTYPE”属性，用来定义用于解析目标DTD

**<!DOCTYPE htmlPUBLIC "-//W3C//DTD HTML 4.01//EN""http://www.w3.org/TR/html4/strict.dtd">**

现在解析SGML是一件痛苦的事情，所以创建了XML使事情更好。XML使用了SGML，例如：在SGML中你必须使用起始和结束标签，但是在XML你可以有自动关闭的结束标签。

XHTML创建于XML，他被使用在HTML4.0中。你可以参考下面代码片段中展示的XML DTD

总之，SGML是所有类型的父类，较旧的HTML利用SGML，HTML4.0使用派生自XML的XHTML

**2.**[**什么是HTML5**](http://blog.jobbole.com/78346/#q3)**？**

HTML5是最新的HTML标准，他的主要目标是提供所有内容而不需要任何的像flash，silverlight等的额外插件，这些内容来自动画，视频，富GUI等

HTML5是万维网联盟（W3C）和网络超文本应用技术工作组（WHATWG）之间合作输出的

**3.**[**为什么HTML5**](http://blog.jobbole.com/78346/#q4)**里面我们不需要DTD（Document Type Definition文档类型定义）？**[**如果我不放入<! DOCTYPE html>标签，HTML5还会工作么？**](http://blog.jobbole.com/78346/#q5)

HTML5没有使用SGML或者XHTML，他是一个全新的东西，因此你不需要参考DTD，对于HTML5，你仅需放置下面的文档类型代码告诉浏览器识别这是HTML5文档

不会，浏览器将不能识别他是HTML文档，同时HTML5的标签将不能正常工作

**4.**[**哪些浏览器支持HTML5**](http://blog.jobbole.com/78346/#q6)**？**

几乎所有的浏览器Safari，Chrome，Firefox，Opera，IE都支持HTML5

**5.**[**HTML5的页面结构同HTML4**](http://blog.jobbole.com/78346/#q7)**或者更前的HTML有什么区别？**

一个典型的WEB页面包含头部，脚部，导航，中心区域，侧边栏。现在如果我们想在在HTML4的HTML区域中呈现这些内容，我们可能要使用DIV标签。

但是在HTML5中通过为这些区域创建元素名称使他们更加清晰，也使得你的HTML更加可读

以下是形成页面结构的HTML5元素的更多细节：

·        <header>：代表HTML的头部数据

·        <footer>：页面的脚部区域

·        <nav>：页面导航元素

·        <article>：自包含的内容

·        <section>：使用内部article去定义区域或者把分组内容放到区域里

·        <aside>：代表页面的侧边栏内容

**6.**[**HTML5中的datalist**](http://blog.jobbole.com/78346/#q8)**是什么？**

**7.**[**HTML5中哪些是不同的新的表单元素类型？**](http://blog.jobbole.com/78346/#q9)

**8.**[**HTML5中什么是输出元素？**](http://blog.jobbole.com/78346/#q10)

**9.**[**什么是SVG**](http://blog.jobbole.com/78346/#q11)**（Scalable Vector Graphics可缩放矢量图形）？**

SVG（Scalable Vector Graphics可缩放矢量图形）表示可缩放矢量图形。他是基于文本的图形语言，使用文本，线条，点等来进行图像绘制，这使得他轻便，显示更加迅速

**10.**[**我们能看到使用HTML5**](http://blog.jobbole.com/78346/#q12)**的SVG的简单例子么？**

**11.**[**HTML5中canvas**](http://blog.jobbole.com/78346/#q13)**是什么？**

Canvas是HTML中你可以绘制图形的区域

**12.**[**我们如何使用Canvas**](http://blog.jobbole.com/78346/#q14)**来画一条简单的线？**

**13.**[**Canvas和SVG**](http://blog.jobbole.com/78346/#q15)**图形之间的区别是什么？**

**14.**[**如何使用Canvas**](http://blog.jobbole.com/78346/#q16)**和HTML5中的SVG去画一个矩形？**

HTML5使用SVG绘制矩形的代码

**<svgxmlns="http://www.w3.org/2000/svg" version="1.1">**

**<rectstyle="fill: rgb(0, 0, 255); stroke-width: 1px; stroke: rgb(0, 0,0);" height="[object SVGAnimatedLength]" width="[objectSVGAnimatedLength]">**

**</rect>**

HTML5使用Canvas绘制矩形的代码

varc=document.getElementById("mycanvas");

varctx=c.getContext("2d");

ctx.rect(20,20,150,100);

ctx.stroke();

**15.**[**CSS（cascading style sheets**](http://blog.jobbole.com/78346/#q17)**级联样式表）中的选择器是什么？**

选择器在你想应用一个样式的时候，帮助你去选择元素。举例，下面是简单的被命名为”instro”的样式，他适用于HTML元素显示红色背景

<style>

.intro{

background-color:red;

}

</style>

应用上面的”intro”样式给div，我们可以使用”class”选择器，如下图所示

<divclass="intro">

<p>Myname is Shivprasad koirala.</p>

<p>Iwrite interview questions.</p>

</div>

**16.**[**如何使用ID**](http://blog.jobbole.com/78346/#q18)**值来应用一个CSS样式？**

假设，你有一个HTML段落标签，使用id是”mytext”，就和下面的片段中显示的那样

<p id="mytext">This is HTML interview questions.</p>

你可以使用”#”选择器和”id”的名字创建一种样式，并把CSS值应用到段落标签中，因此应用样式到”mytext”元素，我们可以使用”#mytext”，如下所示

<style>

#mytext

{

background-color:yellow;

}

</style>

迅速修订一些重要的选择器

设置所有段落标签背景色为黄色

divp

{

background-color:yellow;

}

设置所有div内部的段落标签为黄色背景

divp

{

background-color:yellow;

}

设置所有含有“target”属性的变为黄色背景

**a[target]**

**{**

**background-color:yellow;**

**}**

**<ahref="http://www.questpond.com">ASP.NET interviewquestions</a>**

**<ahref="http://www.questpond.com" target="\_blank">c#interview questions</a>**

**<ahref="http://www.questpond.org" target="\_top">.NETinterview questions with answers</a>**

当控制得到焦点的时候设置所有的元素为黄色背景

input:focus

{

background-color:yellow;

}

根据相关连接操作设置超链接样式

|  |  |
| --- | --- |
| 2  3  4 | a:link    {color:green;}  a:visited {color:green;}  a:hover   {color:red;}  a:active  {color:yellow;} |

**17.**[**CSS中使用列布局是什么？**](http://blog.jobbole.com/78346/#q19)

以下是完整代码

<style>

.magazine

{

-moz-column-count:3;/\* Firefox \*/

-webkit-column-count:3;/\* Safari and Chrome \*/

column-count:3;

-moz-column-gap:40px;/\* Firefox \*/

-webkit-column-gap:40px;/\* Safari and Chrome \*/

column-gap:20px;

-moz-column-rule:4pxoutset #ff00ff; /\* Firefox \*/

-webkit-column-rule:4pxoutset #ff00ff; /\* Safari and Chrome \*/

column-rule:6pxoutset #ff00ff;

}

</style>

你可以使用class属性来应用样式到文本

<divclass="magazine">

Your textgoes here which you want to divide in to 3 columns.

</div>

**18.**[**你能解释一下CSS**](http://blog.jobbole.com/78346/#q20)**的盒子模型么？**

CSS和模型是围绕在HTML元素周围的定义Border(边界)，padding(内边距)和margin(外边距)的矩形空间  
Border(边界)：定义了元素包含的最大区域，我们能够使边界可见，不可见，定义高度和宽度等；  
Padding(内边距)：定义了边界和内部元素的间距  
Margin：定义了边界和任何相邻元素的间距

例如以下是简单的CSS代码定义了盒子的边界，内边距和外边距值

.box {

    width: 200px;

    border: 10pxsolid #99c;

    padding: 20px;

    margin: 50px;

}

现在如果我们应用了以上的CSS到一个如下显示的DIV标签，你输出将会和下面图形中显示的那样。我已经创建两个测试“Some text”和“Some other text”，因此我们能看到多少margin(外边距)的属性功能

**19.**[**你能解释一些CSS3**](http://blog.jobbole.com/78346/#q21)**中的文本效果么？**

这里面试官期待你回答两个Css的文本效果，以下是两种需要注意的效果

**20.**[**什么是Web Workers**](http://blog.jobbole.com/78346/#q22)**？为什么我们需要他们？**

考虑以下会执行上百万次的繁重的循环代码

function  SomeHeavyFunction()

{

for (i = 0; i < 10000000000000; i++)

{

x = i + x;

}

}

比方说上面的循环代码在HTML按钮点击以后执行，现在这个方法执行是同步的，换句话说这个浏览器必须等到循环完成才能操作

<input type="button"onclick="SomeHeavyFunction();" />

**21.**[**Web Worker线程的限制是什么？**](http://blog.jobbole.com/78346/#q23)

Web worker线程不能修改HTML元素，全局变量和Window.Location一类的窗口属性。你可以自由使用Javascript数据类型，XMLHttpRequest调用等。

**22.**[**我们如何在JavaScript**](http://blog.jobbole.com/78346/#q24)**中创建一个worker线程？**

创建一个worker线程，我们需要通过Javascript文件名创建worker对象

varworker = newWorker("MyHeavyProcess.js");

我们需要使用“PostMessage”发送信息给worker对象，下面是相同的代码。

|  |  |
| --- | --- |
| 1 | worker.postMessage(); |

当worker线程发送数据的时候，我们在调用结束的时候，通过”onMessage”事件获取

**23.**[**如何中止Web Worker?**](http://blog.jobbole.com/78346/#q25)

w.terminate();

**24.**[**为什么我们需要HTML5**](http://blog.jobbole.com/78346/#q26)**的服务发送事件？**

网络世界的普遍需求是从服务器更新。以一个股票应用为例，浏览器必须定期从服务器更新最新的股票值。

现在实现这类需求开发者通常写一些PULL的代码，到服务器同时抓取某些区间数据。现在PULL的解决方案是很好的，但是这使得网络健谈有很多的调用，同时增加了服务器的负担。

因此相比于PULL，如果我们能采用某种PUSH的解决方案那会是很棒的。简而言之，当服务器更新的时候，将会发送更新到浏览器客户端，那可以被接受通过使用”SERVER SENT EVENT”

因此首要的是浏览器需要连接将会发送更新的服务器资源，比方说我们有一个”stock.aspx”页面会发送股票更新，因此连接该页面，我们需要使用附加时间来源对象，如下所示：

varsource=newEventSource("stock.aspx");

当我们将要接受服务器发送的更新信息时，我们需要附加功能。我们需要附加功能到”onmessage”事件就像以下显示的那样。

source.onmessage = function (event) {

  document.getElementById("result").innerHTML += event.data + "<br>";

};

现在来自服务端，我们需要去发送事件，下面是一些用命令需要从服务端发送的重要事件列表

Response.Write("retry: 10000");

如果你想附加事件，我们需要使用“addEventListener”事件，如下代码所示：

source.addEventListener('message',function(e) {

  console.log(e.data);

}, false);

来自服务器端的以下信息将会触发Javascript的”message”方法

event: message

data : hello

**25.**[**HTML5中的本地存储概念是什么？**](http://blog.jobbole.com/78346/#q27)

很多时候我们会存储用户本地信息到电脑上，例如：比方说用户有一个填充了一半的长表格，然后突然网络连接断开了，这样用户希望你能存储这些信息到本地，当网络恢复的时候，他想获取这些信息然后发送到服务器进行存储  
现代浏览器拥有的存储被叫做“Local Storage”,你可以存储这些信息。

**26.**[**我们如何从本地存储中添加和移除数据？**](http://blog.jobbole.com/78346/#q28)

数据添加到本地存储采用键值对，以下示例显示了城市数据”India”添加了键”Key001”

**27.**[**本地存储的生命周期是什么？**](http://blog.jobbole.com/78346/#q29)

本地存储没有生命周期，它将保留知道用户从浏览器清除或者使用Javascript代码移除。

**28.**[**本地存储和cookies**](http://blog.jobbole.com/78346/#q30)**（储存在用户本地终端上的数据）之间的区别是什么？**

**29.**[**什么是事务存储？我们如何创建一个事务存储？**](http://blog.jobbole.com/78346/#q31)

会话存储和本地存储类似，但是数据在会话中有效，简而言之数据在你关闭浏览器的时候就被删除了。  
为了创建一个会话存储你需要使用“sessionStorage.variablename.”在以下的代码我们创建了一个名为”clickcount”的变量；  
如果你刷新浏览器则数目增加，但是如果你关闭浏览器，“clickcount”变量又会从0开始。

if(sessionStorage.clickcount)

{

sessionStorage.clickcount=Number(sessionStorage.clickcount)+1;

}

else

{

sessionStorage.clickcount = 0;

}

**30.**[**本地存储和事务存储之间的区别是什么？**](http://blog.jobbole.com/78346/#q32)

本地存储数据持续永久，但是会话在浏览器打开时有效知道浏览器关闭时会话变量重置

**31.**[**什么是WebSQL**](http://blog.jobbole.com/78346/#q33)**？**

WebSQL是一个在浏览器客户端的结构关系数据库，这是浏览器内的本地RDBMS(关系型数据库系统)，你可以使用SQL查询

**32.**[**WebSQL 是HTML5**](http://blog.jobbole.com/78346/#q34)**的一个规范吗？**

不是，许多人把它标记为HTML5，但是他不是HTML5的规范的一部分，这个规范是基于SQLite的

**33.**[**我们如何使用WebSQL**](http://blog.jobbole.com/78346/#q35)**？**

第一步我们需要做的是使用如下所示的“OpenDatabase”方法打开数据库，第一个参数是数据库的名字，接下来是版本，然后是简单原文标题，最后是数据库大小；

vardb=openDatabase('dbCustomer','1.0','Customer app’, 2 \* 1024 \* 1024);

为了执行SQL，我们需要使用“transaction”方法，并调用”executeSql”方法来使用SQL

db.transaction(function(tx)

{

tx.executeSql('CREATETABLE IF NOT EXISTS tblCust(id unique, customername)');

tx.executeSql('INSERTINTO tblcust (id, customername) VALUES(1, "shiv")');

tx.executeSql('INSERTINTO tblcust (id, customername) VALUES (2, "raju")');

}

万一你要使用“select”查询你会得到数据”result”集合，我们可以通过循环展示到HTML的用户界面

db.transaction(function(tx)

{

  tx.executeSql('SELECT\* FROM tblcust', [], function (tx, results) {

  for (i = 0; i < len; i++)

{

    msg = "<p><b>" + results.rows.item(i).log +"</b></p>";

    document.querySelector('#customer).innerHTML+=  msg;

}

 }, null);

});

**34.**[**HTML5中的应用缓存是什么？**](http://blog.jobbole.com/78346/#q36)

一个最需要的事最终是用户的离线浏览，换句话说，如果网络连接不可用时，页面应该来自浏览器缓存，离线应用缓存可以帮助你达到这个目的  
应用缓存可以帮助你指定哪些文件需要缓存，哪些不需要。

**35.**[**HTML5中我们如何实现应用缓存？**](http://blog.jobbole.com/78346/#q37)

首先我们需要指定”manifest”文件，“manifest”文件帮助你定义你的缓存如何工作。以下是”mainfest”文件的结构

CACHEMANIFEST

# version1.0

CACHE :

Login.aspx

·        所有manifest文件都以“CACHE MANIFEST”语句开始.

·        #（散列标签）有助于提供缓存文件的版本.

·        CACHE 命令指出哪些文件需要被缓存.

·        Mainfest文件的内容类型应是“text/cache-manifest”.

以下是如何在ASP.NET C#使用manifest缓存

Response.ContentType= "text/cache-manifest";

Response.Write("CACHEMANIFEST \n");

Response.Write("#2012-02-21 v1.0.0 \n");

Response.Write("CACHE: \n");

Response.Write("Login.aspx\n");

Response.Flush();

Response.End();

创建一个缓存manifest文件以后，接下来的事情实在HTML页面中提供mainfest连接，如下所示：

<html manifest="cache.aspx">

当以上文件第一次运行，他会添加到浏览器应用缓存中，在服务器宕机时，页面从应用缓存中获取

**36.**[**我们如何刷新浏览器的应用缓存？**](http://blog.jobbole.com/78346/#q38)

应用缓存通过变更“#”标签后的版本版本号而被移除，如下所示：

|  |  |
| --- | --- |
| 2  3  4  5  6  7 | CACHEMANIFEST  # version2.0(new)  CACHE :  Login.aspx  Aboutus.aspx  NETWORK :  Pages.aspx |

**37.**[**应用缓存中的回退是什么？**](http://blog.jobbole.com/78346/#q39)

应用缓存中的回退帮助你指定在服务器不可访问的时候，将会显示某文件。例如在下面的manifest文件中，我们说如果谁敲击了”/home”同时服务器不可到达的时候，”homeoffline.html”文件应送达

FALLBACK:

/home/ /homeoffline.html

**38.**[**应用缓存中的网络是什么？**](http://blog.jobbole.com/78346/#q40)

网络命令描述不需要缓存的文件，例如以下代码中，我们说”home.aspx”永远都不应该被缓存或者离线访问。

|  |  |
| --- | --- |
| 1  2 | NETWORK:  home.aspx |

**39. 如何在 HTML5 页面中嵌入音频?**

HTML5 支持 MP3、Wav 和 Ogg 格式的音频，下面是在网页中嵌入音频的简单示例：

<audiocontrols>

    <sourcesrc=”jamshed.mp3″ type=”audio/mpeg”>

    Yourbrowser does’nt support audio embedding feature.

</audio>

**40. 响应式布局用什么实现？**

利用css3的media query媒体查询功能或bootsrap框架

**41. 如何在 HTML5 页面中嵌入视频？**

42. **<video** src="foo.mp4"  width="300" height="200" controls**>**

43.     Your browser does not support the **<video>** element.

44. **</video>**

**45. HTML5 有哪些不同类型的存储？**

**localStorage**适用于长期存储数据，浏览器关闭后数据不丢失

**sessionStorage**存储的数据在浏览器关闭后自动删除

**46. HTML5 引入什么新的表单属性？**

**47. HTML5 标准提供了哪些新的 API？**

多媒体：video、audio、

游戏：canvas、webgl、

存储：localstorage、sessonstorage、websql、indexedDB

网络：websocket

**48. HTML5的页面结构**

**<!DOCTYPEhtml>**

**<html>**

**<head>**

**<title></title>**

**</head>**

**<body></body>**

**</html>**

**49. web storage和cookie的区别**

"与Cookie相比，Web Storage存在不少的优势，概括为以下几点：

1. 存储空间更大：能提供5MB的存储空间（不同[浏览器](https://www.baidu.com/s?wd=%E6%B5%8F%E8%A7%88%E5%99%A8&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3nvDLPhn1uW6vuy79mvNB0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHRYrjc1nWcd)的提供的空间不同），Cookie仅4KB

2. 存储内容不会发送到服务器：当设置了Cookie后，Cookie的内容会随着请求一并发送的服务器，这对于本地存储的数据是一种带宽浪费。而Web Storage中的数据则仅仅是存在本地，不会与服务器发生任何交互。

3. 更多丰富易用的接口：Web Storage提供了一套更为丰富的接口，使得数据操作更为简便。

4. 独立的存储空间：每个域（包括子域）有独立的存储空间，各个存储空间是完全独立的，因此不会造成数据混乱。

**50. CSS中 link 和@import 的区别是？**

1. 老祖宗的差别。link属于XHTML标签，而@import完全是CSS提供的一种方式。 link标签除了可以加载CSS外，还可以做很多其它的事情，比如定义RSS，定义rel连接属性等，@import就只能加载CSS了。

2. 加载顺序的差别。当一个页面被加载的时候（就是被浏览者浏览的时候），link引用的CSS会同时被加载，而@import引用的CSS会等到页面全部被下载完再被加载。所以有时候浏览@import加载CSS的页面时开始会没有样式（就是闪烁），网速慢的时候还挺明显。

3. 兼容性的差别。由于@import是CSS2.1提出的所以老的浏览器不支持，@import只有在IE5以上的才能识别，而link标签无此问题。

4. 使用dom控制样式时的差别。当使用javascript控制dom去改变样式的时候，只能使用link标签，因为@import不是dom可以控制的。

**51. CSS3有哪些新特性？**

第 1 选择器  
第 2 RGBA和透明度  
第 3 多栏布局  
第 4 多背景图  
第 5 Word Wrap  
第 6 文字阴影  
第 7 @font-face属性  
第 8 圆角(边框半径)  
第 9 边框图片  
第 10 盒阴影  
第 11 盒子大小  
第 12 媒体查询  
第 13 语音

**52. HTML5中什么是输出元素？**

output：定义不同的输出类型，比如脚本