

五分钟学会HTML5! (一)

HTML5 (/lib/tag/HTML5)

2012-01-18 17:21:29 发布

您的评价:

3.7

收藏


5收藏

摘要：本文涵盖新的语义标签、为绘图和动画而设计的画布（**canvas**）、音频和视频支持，以及在旧版浏览器上如何使用**HTML5**。看完这些内容所用的时间可能会比五分钟多一点点，但我保证会讲得快一些。请接着看下去，下面的内容肯定值得你这么做！

本文的原链接是<http://www.codeproject.com/KB/solution-center/HTML5-in-5.aspx> (<http://www.codeproject.com/KB/solution-center/HTML5-in-5.aspx>).

毫无疑问，对于开发人员而言，**HTML5**已是一个热点话题。如果你需要快速了解**HTML5**的功能的基本原理，阅读本文是你最好的选择。

本文来自The Code Project的付费搜索位置，由Solution Center提供。这里的文章致力于向大家提供我们认为对开发人员来说有用和有价值的相关产品和服务的信息。

由enter image description here和HTML5 / CSS3 Zone带来。

本文涵盖新的语义标签、为绘图和动画而设计的画布（**canvas**）、音频和视频支持，以及在旧版浏览器上如何使用**HTML5**。看完这些内容所用的时间可能会比五分钟多一点点，但我保证会讲得快一些。请接着看下去，下面的内容肯定值得你这么做！

语义标签和页面布局

我们首先讲一个关于一所大学的经典故事。这所大学在建设校园草地时，没有开辟任何步行小道，他们把整个空地种上草，然后等待。

一年以后，人们经常走过的地方，草都被踩光了。于是，这所大学就将这些地方铺设为真正的人行道。

棒极了！这些人行道是人们真正“走”出来的。

HTML5新语义元素的诞生正是基于同样的逻辑。（关于这一点，可以参看W3C设计指南中的“Pave the Cowpaths” (<http://www.w3.org/TR/html-design-principles/>)。）

语义元素清晰地浏览器和开发人员表明了它们的涵义和用途，要理解这一点，可以将它们与< **div**>标签进行对比。在**HTML**文件内，< **div**>标签能定义出一个分区或者一个小节，但是它无法告诉我们分区里的内容，不能传达任何清晰的涵义。

开发人员通常会将ID或者类名与< **div**>标签配合使用，这会向程序员传达更多的涵义。但遗憾的是，这并不能帮助浏览器领会那种标签的用途。< **div** id="header">

在**HTML5**中，有不少新的富含语义的元素，可以向浏览器和开发人员传达元素的用途。< **header**>

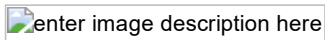
W3C挖掘了数亿个的Web页面，找出开发人员一直在使用的ID和类名。一旦开发人员抛出**div1**、**div2**这些无意义标签，他们就得到了一个描述非常细致的已经在使用的元素的列表，并让其成为W3C的标准设置。

下面是**HTML5**的一部分新的语义元素：

- **article**
- **aside**
- **figcaption**
- **figure**
- **footer**
- **header**
- **hgroup**
- **mark**
- **nav**
- **section**
- **time**

由于这些元素的语义很丰富，相信你可能会猜出其中大部分元素的作用。

为了说得更清楚，下面给出一张图示。



header和**footer**的作用不言自明，**nav**将创建一个导航条或者菜单条。此外，你可以用**section**和**article**将页面内容分为几个部分。最后，**aside**元素用来安置附带的内容，比如说，以边栏的形式放上一些相关链接。

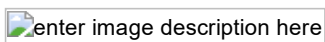
下面是一个简单的例子，其中的代码就用到了这些元素。

```
<!DOCTYPE html> <html> <head>      <meta charset="utf-8" />      <title>Title</title>      <link rel="stylesheet" href="css/main.css" />
</section>      </article>      <article>      <header>
</section>      </article>      </section>      <aside>      <section>
      src="http://www.windowsdevbootcamp.com/Images/JennMar.jpg"
      alt="Jennifer Marsman" />      <figcaption>Jennifer Marsman</figcaption>
```

当然，在这个例子里，我也引出了另外几个新元素。

不知道你是否注意到**hgroup**元素，它将**h1**和**h2**这两个**header**组合到一起了？

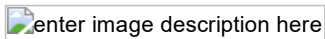
我们可以用**mark**元素将重要文本高亮显示或标记出来。如果要在内容中插入一张图（图像、图表、照片和代码片段等），可以使用**figure**元素。而**figcaption**元素能为图加上标题。当把以上代码和一些CSS代码组合后，得到的Web页面如下图所示。（注：这里的CSS部分来自我博学多才的同事Brandon Satrom所写的TechEd talk，而其他与页面美化无关的效果都是我做的。）



在一些擅长CSS的朋友（可惜我不擅长CSS）的帮助下，上面这个结果看起来靓爆了！实际上，有了HTML的描述性，完成这个页面非常容易。

还要说明一点，如果你是Visual Studio的拥趸，请确保已经安装了Visual Studio 2010 SP1。否则，你将发现Visual Studio并不理解HTML5元素，从而导致Web页面上到处是歪歪曲曲的线条。

然后，在Visual Studio菜单中依次选择“Tools”-“Options”，打开一个“Options”对话框。从左侧的导航面板，依次展开“Text Editor”和“HTML”，单击“Validation”。在“Target”下拉列表框中选择HTML5，这样就能得到对HTML5 IntelliSense的支持了。



为了深入理解语义元素，请访问以下链接：

How to Enable HTML5 Standards Support （如何启动HTML5标准支持）

(<http://msdn.microsoft.com/library/gg699338%28v=VS.85%29.aspx>)

Semantics section of the W3C HTML5 specification （W3C HTML5规范的语义部分）

(<http://dev.w3.org/html5/spec/Overview.html>)

Dive into HTML5 chapter on semantics entitled “What Does It All Mean?” （“What Does It All Mean?”关于语义的Dive into HTML5一章）(<http://diveintohtml5.org/semantics.html>)

使用Canvas元素在HTML5中进行绘画

HTML5中另外一个新元素是< canvas>标签。顾名思义，它就是一块用来绘画的空白平面。你需要使用JavaScript在这块画布上进行操作和绘画。

你可能需要为canvas元素赋予一个id属性，这样就能通过JavaScript代码以编程的方式访问它。如果你正在使用jQuery，并且在页面上只有一个canvas，你可以使用\$('canvas')来访问它，而不用为它命名。

你也可以为canvas指定height（高度）和width（宽度），但这不是必需的。在< canvas> 和< /canvas>标签之间，你还可以指定一些文本，显示在不支持canvas元素的浏览器中。

下面是使用canvas进行绘画的一个简单例子。（我希望画一张苏格兰国旗，如果画得不是非常精确，请见谅。）

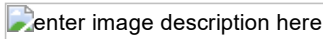
```
<!DOCTYPE HTML> <html> <body>      <canvas id="myCanvas">Your browser does not support the canvas element</canvas>
    var ctx = canvas.getContext('2d');

    // Draw blue rectangle      ctx.fillStyle = '#0065BD';
    ctx.fillRect(0, 0, 125, 75);

    // Draw white X      ctx.beginPath();
    ctx.lineWidth = "15";
    ctx.strokeStyle = "white";
    ctx.moveTo(0, 0);
    ctx.lineTo(125, 75);
    ctx.moveTo(125, 0);
    ctx.lineTo(0, 75);
    ctx.stroke();

</script> </body> </html>
```

上面的代码将生成以下结果。



现在我们从头到尾分析一下代码。

首先，我创建了一块画布（**canvas**），并为它赋予了一个ID叫“myCanvas”。如果运行这段代码的浏览器不支持**canvas**元素，它就会在旗帜原本要出现的位置显示“Your browser does not support the canvas tag”。

接下来，我编写了一段脚本。记住，**canvas**标签只是一个用来画图的容器，你必须用**JavaScript**来画图并将其呈现出来。我首先通过使用ID“myCanvas”获得了对**canvas**的一个引用，然后得到**canvas**的上下文。上下文所提供的方法和属性，都可以用来在**canvas**上操作图形和进行绘画。这里指定了参数“2d”，表示我将在2维的环境中进行绘画。

第三步，完成蓝色矩形的绘画。我用**fillStyle**方法指定了矩形的颜色为蓝色，再用**fillRect**方法画出了矩形，后者的参数指定了矩形的大小与位置。**fillRect(0, 0, 125, 75)**表示：从左上角的顶点（0,0）开始画一个矩形，宽为125像素，高为75像素。

最后，我在旗帜上画出了一个白色的“X”。我首先调用**beginPath**方法启动动画路径的进程。指定的**lineWidth**属性值（也就是路径的宽度）为15像素，这是通过不断猜想加尝试才找到的看起来最合适的值。另一个属性**strokeStyle**则被指定为“white”，以表示路径颜色为白色。接下来，依靠**moveTo**和**lineTo**两个方法描绘出了整个路径。这两个方法都会定位出一个用来绘图的光标，其区别在于：前者移动光标的时候不会画出一条线，而后者在移动的同时会画线。在画X的过程中，首先从（0,0）——左上角开始，然后画一条线到（125,75）——右下角。接着把光标移到（125,0）——右上角，一笔画到（0,75）——左下角。最后的**stroke**方法将真正地呈现这些笔画。

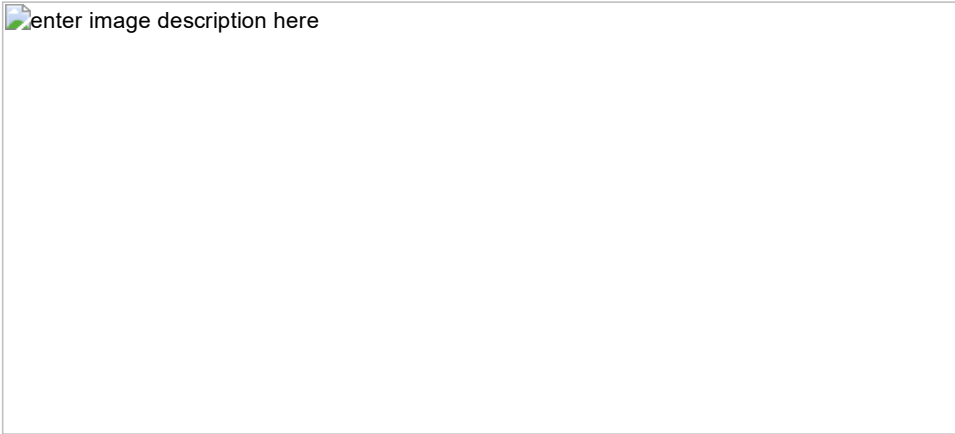
canvas与SVG的简单对比

可缩放矢量图形（**Scalable Vector Graphics**，简称**SVG**）是在浏览器窗口进行绘画的一个早期标准。随着**HTML5**的**canvas**的发布，许多人都想知道它们之间到底孰优孰劣。

在我看来，它们之间最大的区别就是：**canvas**使用立即呈现模式，而**SVG**使用保留呈现模式。这意味着，**canvas**可以让所画的图形立即呈现在显示器里。在上面给出的代码中，一旦旗帜画完，系统就会“忘掉”它，且不会有任何状态保留下来。如果对它进行改动，将会导致彻底的重画。与之相比，**SVG**保留了所呈现对象的一个完整模型。要对图做出改动时，你只需要指出改动的地方（例如矩形的新位置），浏览器来负责呈现新的图形。这节省了开发人员的工作，但也为维护模块付出了性能上的代价。

能够同时通过**CSS**和**JavaScript**来进行设计，是**SVG**另一个值得称道的优点。与之相比，**canvas**只能通过**JavaScript**来进行操作。

两者其他的区别如下表所示。



如果想了解更多细节的不同之处，推荐你去阅读下面一些文章（前面的表就是经过作者同意后从下面的文章里拿来的）。

- SVG和Canvas使用心得（Patrick Dengler 著）(<http://blogs.msdn.com/b/ie/archive/2011/04/22/thoughts-on-when-to-use-canvas-and-svg.aspx>)
- 深入HTML5 Canvas（Jatinder Mann 著）(<http://channel9.msdn.com/events/MIX/MIX11/HTM03>)
- HTML5 Canvas入门（John Bristowe 著）(<http://bristowe.com/blog/2011/7/9/an-introduction-to-html5-canvas-3.html>)

音频和视频支持

HTML5最大的新特色之一就是支持音频和视频。在HTML5之前，我们必须使用插件如Silverlight (<http://www.silverlight.net/downloads>) 或Flash (<http://get.adobe.com/cn/flashplayer/>)来实现这些功能。在HTML5中，你可以直接使用新标签< audio> 和 < video>将音频和视频嵌入到页面。

从编码的角度来看，audio和video元素使用起来很简单（下面我会深入地谈到它们的一些属性）。所有的主流浏览器都支持audio和video元素，包括最新版本的IE、Firefox、Chrome、Opera和Safari。虽然如此，有一点很关键：你需要编码解码器去播放音频和视频，而不同的浏览器支持的编码解码器是不同的（想要进一步了解视频容器和编码解码器，可以访问链接<http://diveintohtml5.org/video.html> (<http://diveintohtml5.org/video.html>)）。

幸运的是，这不会成为技术障碍，因为HTML5对音频和视频的支持方式非常灵活（各个浏览器支持的音频和视频格式一般有好几种，它会轮流使用这几种格式去播放音频和视频）。*当然，你最好提供多种格式的音频和视频源，以满足不同浏览器的需求。*此外，你还可以继续使用Silverlight或Flash插件。最后，在开始和结束标签（例如< audio> 和 < /audio>）之间的文本，会在浏览器不支持audio或video元素的时候显示到web页面上。

例如：

```
<audio controls="controls">      <source src="laughter.mp3" type="audio/mp3" />      <source s
```

执行这段代码时，浏览器将首先试图播放laughter.mp3文件。如果没有合适的编码解码器去播放，它会转向播放下一个文件laughter.ogg。如果浏览器根本不认识audio元素，它会在音频控件的位置显示文本“Your browser does not support the audio element”。

关于音频和视频，需要警告大家的是：HTML5并没有内置的数字版权管理（digital rights management，简称DRM）支持，作为开发人员，你必须自己实现它。关于这个话题，可以参看这里 (<http://www.w3.org/html/wiki/FAQs>)。（如果你需要DRM内容，也可以试试Silverlight DRM documentation，它或许是更便捷的解决方案。）

现在，让我们深入探讨这两个新元素。

音频

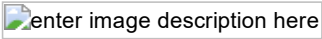
首先，让我们看看< audio>的更多细节。

```
<audio controls="controls">      <source src="laughter.mp3" type="audio/mp3" />      <source s
```

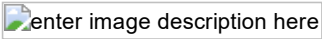
前面已经介绍过，浏览器会依次尝试播放每个音频源，直到找到一个能播放的源。

注意这里多了一个controls属性。使用这个属性后，浏览器将显示音频回放控件，包括播放/暂停按钮、时间显示控件、静音按钮和音量控件。在绝大多数情形下，把这些控件展示给用户是明智的。我非常讨厌打开一个网页时有声音响起，而且没法把它停止、静音或者调低，难道你不是这样？

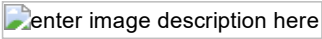
在IE浏览器中，各种音频控件如下图所示。



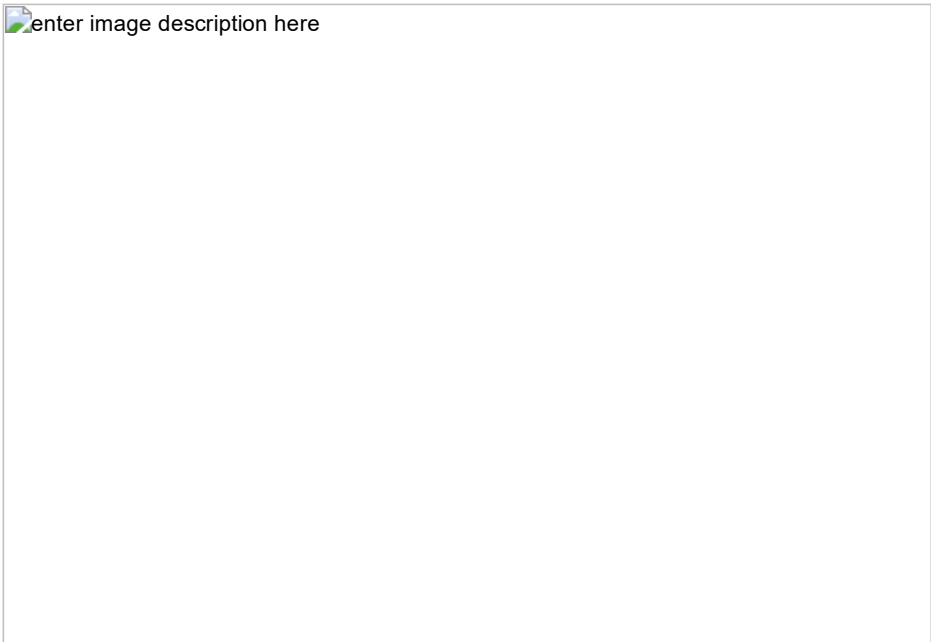
在不同的浏览器中，这些控件的外观并不一样。**Chrome**浏览器中的音频回放控件如下图所示（它正在播放一首歌）。当你的鼠标指针悬停在整个控件最右边的声音图标上时，将会弹出一个下拉式的音量控制条。



下图是**Firefox**中的控件样式，截图停在了一首歌暂停的时刻。和**Chrome**一样，它也有一个弹出式的音量控制条（本图未显示），可以通过将鼠标指针悬停在控件最右边的声音图标上弹出它。



audio元素的其他一些有趣的属性如下表所示。



下面这段示例代码，执行时将显示音频回放控件，并在音频文件加载完成后立刻自动播放，完成后再一遍又一遍地重复播放。

```
<audio controls="controls" autoplay="autoplay" loop="loop">      <source src="laughter.mp3" t
< /pre>
```

如果你想在自己的浏览器中好好研究< audio>元素，可以参考<http://w3schools.com/> (<http://w3schools.com/>)上的“Tryit Editor”。它可以用来编辑一些示例代码，然后看看会发生什么。此外，你也可以参阅下面这篇文章——[How to add an HTML5 audio player to your site](http://msdn.microsoft.com/library/gg589483%28v=VS.85%29.aspx)（如何向你的站点添加HTML5音频播放器）（<http://msdn.microsoft.com/library/gg589483%28v=VS.85%29.aspx>）。

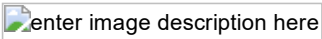
视频

现在，让我们试试< video>元素。

```
<video width="320" height="240" controls="controls">      <source src="movie.ogg" type="vide
< /pre>
```

前面讲过，video元素也支持多个源，它会按顺序依次尝试播放。

与audio元素相同，video元素也有一个controls属性。视频控件在IE浏览器中的截图如下所示。



video元素其他一些有趣的属性如下表所示。



阅读目录

- 语义标签和页面布
- 使用Canvas元素在
- 绘画
- canvas与SVG的简
- 音频和视频支持
- 音频
- 视频

如果你希望继续研究< video>元素，可以使用来自<http://w3schools.com/> (<http://w3schools.com/>)的“Tryit Editor”，用它来编辑一些示例代码，然后看看代码生成的效果。

如果想学习关于video和audio的更多内容，请访问以下几个链接。

- 使用< audio>和< video>必须知道的5件事情 (<http://channel9.msdn.com/events/mix/mix11/htm12>)
- 如何向你的站点添加HTML5音频文件 (<http://msdn.microsoft.com/library/gg589483%28v=VS.85%29.aspx>)
- W3C Schools HTML5 video (http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5_video_all)

同类热门经验

- HTML5资料大全 (</lib/view/open1353205667185.html>)
- 利用HTML5 Canvas实现一个时钟 (</lib/view/open1331473558109.html>)
- HTML5标签云 TagCanvas (</lib/view/open1323758129156.html>)
- 百行 HTML5 代码实现四种双人对弈游戏 (</lib/view/open1341242813621.html>)
- Anytodo - 基于HTML5的离线便签应用 (</lib/view/open1328749658405.html>)
- html5 audio音频播放全解析 (</lib/view/open1407401112973.html>)

相关文档 — 更多 (http://www.open-open.com/doc)	相关经验 — 更多 (http://www.open-open.com/lib)	相关讨论 — 更多 (http://www.open-open.com/solution)
• HTML5 设计原理.doc (http://www.open-open.com/doc/view/0c25518cafb4026ad99003f5973759d)	五分钟学会HTML5! (二) (/lib/view/open1326878383577.html)	• HTML5新手入门指南 (http://www.open-open.com/solution/view/1320764201952)
• HTML5 权威指南（英文版）.pdf (http://www.open-open.com/doc/view/9709f4cf0d34403db871a0879a380483)	HTML5 视频直播 (一) (/lib/view/open1430009218838.html)	• 当设计师遭遇 HTML5 (http://www.open-open.com/solution/view/1321250638312)
• HTML5讲义.pdf (http://www.open-open.com/doc/view/92e39d83d21641649b2fd9ba469d0424)	LibJS - 一个HTML5应用框架 (/lib/view/open1356935975479.html)	• HTML5将重塑Web世界? (http://www.open-open.com/solution/view/1320630981358)
• HTML5 权威指南.pdf (http://www.open-open.com/doc/view/320974f7b6cb4eee8a498831932f0ef3)	利用HTML5 Canvas实现一个时钟 (/lib/view/open1331473558109.html)	HTML5跨平台优势带动职位需求高涨 (http://www.open-open.com/solution/view/1417661811417)
• HTML5 移动开发即学即用.pdf (http://www.open-open.com/doc/view/f40ab4546c2e45188dcca5d694c565d5)	HTML5 Forms表单的一个子集模块 - HTML5Widget (/lib/view/open1337091202541.html)	HTML4 和 HTML5 的 10个关键区别 (http://www.open-open.com/solution/view/1318518374390)
• 《HTML5周刊》第005期.pdf (http://www.open-open.com/doc/view/c4faa0d926d74075b7668b192023e448)	一个简单的html5画板 (/lib/view/open1372381058010.html)	HTML5利用canvas构建 Web五子棋游 (http://www.open-open.com/solution/view/1318518374390)
• HTML5 精彩世界.pdf (http://www.open-open.com/doc/view/e805076400384578a7559b0ddf72d8c7)	HTML5 利用canvas构建 Web五子棋游 (http://www.open-open.com/solution/view/1318518374390)	摩托罗拉发布 RhoElements HTML5 框架
• HTML5权威指南（中文版）.pdf (http://www.open-open.com/doc/view/92e39d83d21641649b2fd9ba469d0424)		

- open.com/doc/view/4533e5ba94f24010bfd77f9aa6404a2f) 戏程序设计 (http://www.open-
- HTML5移动开发即学即用(双色).pdf (http://www.open-open.com/doc/view/eca930c1567045f8b4489e0fc504652c) (/lib/view/open1355141585664.html) open.com/solution/view/1318331643453)
 - HTML5 移动开发即学即用(双色).pdf (http://www.open-open.com/doc/view/793510d9be49473ea1aa296278b92ae0) (/lib/view/open1342353626838.html) HTML5技术介绍以及优缺点概要 (http://www.open-open.com/solution/view/1450427234657)
 - HTML5开发Android应用程序概述.pdf (http://www.open-open.com/doc/view/953b8ad4b32d46dcb5823fd1f478aab3) (/lib/view/open1402535430509.html) • 一个极快的HTML5游戏框架: Kiwi.js
 - Html5与css3.ppt (http://www.open-open.com/doc/view/6dc83ab36a254be6bd2b2e472f7762f3) (/lib/view/open1402535430509.html) • Envision.js – 一个基于HTML5技术的数
据可视化库
 - HTML5 教程指南.pdf (http://www.open-open.com/doc/view/06308476c3bf4c418b8441a7df7a614a) (/lib/view/open1333454741483.html) Html5+css3+angularjs+jquery+webAPI
 - HTML5 实战.pdf (http://www.open-open.com/doc/view/ae53883331fb412e805d440b8e20e6ca) (/lib/view/open1434258938396.html) 开发手机web (一)
 - HTML5 实战.pdf (http://www.open-open.com/doc/view/70f4155c454d4e0b9b7dd4ec9cbfa197) (/lib/view/open1424600649553.html) • 一个简单的HTML5媒体播放器: Plyr
 - HTML5教程.pdf (http://www.open-open.com/doc/view/d6aa705e993f42d198a9900840631735) (/lib/view/open1424600649553.html) • HTML5摇一摇(上)——如何判断设备
摇动
 - HTML5 for Web Designers.pdf (http://www.open-open.com/doc/view/ec98ca0edcf84c849b0d4402d9e43e96) (/lib/view/open1453099423792.html) Stitches - 一个 HTML5 Sprite Sheet 生
成器
 - HTML5 视频应用.pdf (http://www.open-open.com/doc/view/5f1de75030204ec899983a4b273dad7f) (/lib/view/open1347605416632.html)
 - HTML5 游戏开发.pdf (http://www.open-open.com/doc/view/66e2813e2f364887b038c02ee401e256)
 - Programming HTML5 Applications .pdf (http://www.open-open.com/doc/view/b39635dbf0ae419393ccf8333ca2fe37)

©2006-2016 深度开源



(http://www.open-open.com/)

浙ICP备09019653号-31

(http://www.miibeian.gov.cn/) 站长统计

(http://www.cnzz.com/stat/website.php?

web_id=1257892335)