**为什么HTML5里面我们不需要DTD（Document Type Definition文档类型定义）？**

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，你仅需要通过DOCTYPE告诉浏览器识别这是HTML5文档即可.

**如果我不放入<! DOCTYPE html> ，HTML5还会工作么？**

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

**哪些浏览器支持HTML5？**

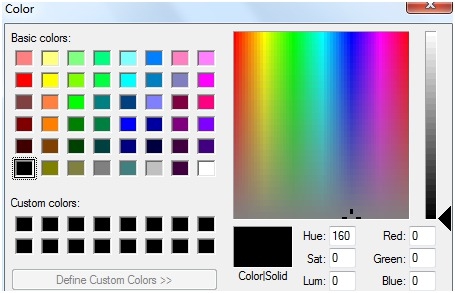
几乎所有的浏览器Safari，Chrome，Firefox，Opera，IE都支持HTML5

**HTML5中有哪些新的表单元素类型？**

这里有10个重要的新的表单元素在HTML5介绍

如果你想显示颜色选择对话框

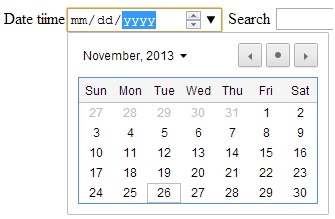
1  <input type="color" name="favcolor">



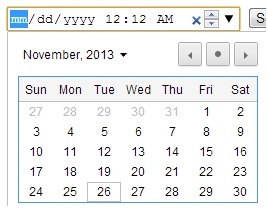
如果你想显示日历对话框

1  <input type="date" name="bday">

如果你想显示含有本地时间的日历



1  <input type="datetime‐local" name="bdaytime">



如果你想创建一个含有email校验的HTML文本框，我们可以设置类型为“email”

1  <input type="email" name="email">



对于URL验证设置类型为”url”，如下图显示的HTML代码

1  <input type="url" name="sitename">



如果你想用文本展示数字范围，你可以设置类型为“number”

1  <input type="number" name="quantity" min="1" max="5">



如果你想显示范围控制，你可以使用类型”range”

1  <input type="range" min="0" max="10" step="2" value="6">



想让文本框作为搜索引擎

1  <input type="search" name="googleengine">

想只能输入时间

1  <input type="time" name="usr\_time">

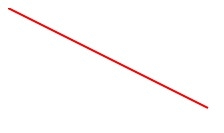
如果你想使用文本框接受电话号码

1  <input type="tel" name="mytel">

**什么是SVG（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.我们能看到使用HTML5的SVG的简单例子么？

比方说，我们希望使用HTML5 SVG去显示以下简单的线条



下面是HTML5代码

1 <svg id="svgelem" height="[object SVGAnimatedLength]" xmlns="http://w

2 <line style="stroke: rgb(255, 0, 0); stroke‐width: 2px;" y2="[object

* </line>

**HTML5中canvas是什么?**

Canvas是HTML中你可以绘制图形的区域

**我们如何使用Canvas来画一条简单的线？**

(1)定义Canvas区域 (2)获取访问canvas上下文区域 (3)绘制图形

定义Canvas区域

定义Canvas区域你需要使用下面的HTML代码，这定义了你能进行绘图的区域

1  <canvas id="mycanvas" width="600" height="500" style="border:1px soli



获取画布区域的访问

在画布上进行绘图我们首先需要获取上下文区域的关联，下面是获取画布区域的代码。

1 var c=document.getElementById("mycanvas");



var ctx=c.getContext("2d");

绘制图形

现在一旦你获取了访问上下文，我们就可以开始在上下文中绘制了。首先调用“move”方法并从一个点开始， 使用线条方法绘制线条然后使用stroke方法结束。

ctx.moveTo(10,10);

* ctx.lineTo(200,100);
* ctx.stroke();

以下是完整的代码

1 <body onload="DrawMe();">

* <canvas id="mycanvas" width="600" height="500" style="border:1px sol
* </body>
* <script>
* function DrawMe()
* {

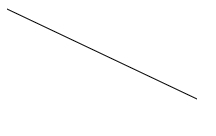
7 var c=document.getElementById("mycanvas");

* var ctx=c.getContext("2d");
* ctx.moveTo(10,10);

1. ctx.lineTo(200,100);
2. ctx.stroke();
3. }



你可以得到以下输出



**Canvas和SVG图形的区别是什么？**

|  |  |
| --- | --- |
| SVG | Canvas |
|  |  |
| 这个就好像绘制和记忆，换句话说任 | Canvas就像绘制和忘记，一旦绘制完 |
| 何使用SVG绘制的形状都能被记忆和 | 成你不能访问像素和操作它 |
| 操作，浏览器可以再次显示 |
|  |
|  |  |
| SVG对于创建图形例如CAD软件是良 | Canvas在绘制和忘却的场景例如动画 |
| 好的，一旦东西绘制，用户就想去操 | 和游戏是良好的 |
| 作它 |
|  |
|  |  |
| 因为为了之后的操作，需要记录坐 | 因为没有记住以后事情的意向，所以 |
| 标，所以比较缓慢 | 更快 |
|  |  |
| 我们可以用绘制对象的相关事件处理 | 我们不能使用绘制对象的相关事件处 |
| 理，因为我们没有他们的参考 |
|  |
|  |  |
| 分辨率无关 | 分辨率相关 |
|  |  |

**如何使用Canvas和HTML5中的SVG画一个矩形**

HTML5使用SVG绘制矩形的代码

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

2 <rect style="fill: rgb(0, 0, 255); stroke‐width: 1px; stroke: rgb(0,

* </rect>



HTML5使用Canvas绘制矩形的代码

1 var c=document.getElementById("mycanvas");

* var ctx=c.getContext("2d");

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

* ctx.stroke();

**Web Worker线程的限制是什么？**

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。

**本地存储和cookies（储存在用户本地终端上的数据）之间的区别是什么？**

|  |  |  |
| --- | --- | --- |
|  | Cookies | Local storage |
|  |  |  |
| 客户 | 客户端和服务端都能访 | 只有本地浏览器端可访问数据，服务器不 |
| 问数据。Cookie的数据 |
| 端/服 | 能访问本地存储直到故意通过POST或 |
| 通过每一个请求发送到 |
| 务端 | 者GET的通道发送到服务器 |
| 服务端 |
|  |  |
|  |  |  |
| 大小 | 每个cookie有4095byte | 每个域5MB |
|  |  |  |
|  | Cookies有有效期，所 | 没有过期数据，无论最后用户从浏览器删 |
| 过期 | 以在过期之后cookie和 | 除或者使用Javascript程序删除，我们都 |
|  | cookie数据会被删除 | 需要删除 |
|  |  |  |