

互联网应用开发技术

*Web Application Development*

---

## 第2课

# WEB前端—HTML和CSS

Episode Two  
**HTML & CSS**

陈昊鹏  
[chen-hp@sjtu.edu.cn](mailto:chen-hp@sjtu.edu.cn)



- HTML
  - HTML tags
  - HTML 5
- CSS
  - CSS syntax
  - CSS in HTML
  - 本节课代码样例来自于[www.w3school.com.cn](http://www.w3school.com.cn)网站

- Hypertext Markup Language (HTML)
  - is the Web's core language for creating documents and applications for everyone to use, anywhere.
  - in October 2014, the HTML Working Group published **HTML5** as W3C Recommendation.

```
<html>
<body>
  <h1>My First Heading</h1>
  <p>My first paragraph.</p>
</body>
</html>
```

- HTML Markup Tag, called HTML tag
  - Keyword enclosed by angle brackets , e.g. <html>
  - The tags appear in pairs, e.g. <b> and </b>
  - The first one is start tag while the second one is end tag

- HTML Document = Webpage
  - HTML documents describe webpages
  - An HTML document consists of HTML tags and text
  - HTML documents are called webpages
- Web Browser retrieves HTML docs, and presents them as webpages
  - Web Browser uses HTML tags to interpret the content of webpages rather than presents HTML tags

```
<html>
<body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
</body>
</html>
```

- HTML heading
  - <h1>, <h2>,...,<h6>

```
<html>  
  
<body>  
  
<h1>This is heading 1</h1>  
<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>  
<h4>This is heading 4</h4>  
<h5>This is heading 5</h5>  
<h6>This is heading 6</h6>  
  
</body>  
</html>
```

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

- HTML paragraph
  - <p>

```
<html>
<body>

<p>This is a paragraph.</p>
<p>This is a paragraph.</p>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a paragraph.

This is a paragraph.

This is a paragraph.

- HTML link

- <a>
- href attribute

```
<html>
<body>

<a
href="http://www.w3school.com.cn">
This is a link</a>

</body>
</html>
```

This is a link

- HTML image
  - <img>
  - src, width & height attributes

```
<html>
<body>



</body>
</html>
```



- HTML table

- <table>, <th>, <tr>, <td>.....

```
<html>  
  
  <body>  
  
    <table border="1">  
      <tr>  
        <th>Heading</th>  
        <th>Another Heading</th>  
      </tr>  
      <tr>  
        <td>row 1, cell 1</td>  
        <td>row 1, cell 2</td>  
      </tr>  
      <tr>  
        <td>row 2, cell 1</td>  
        <td>row 2, cell 2</td>  
      </tr>  
    </table>  
  
  </body>  
</html>
```

Heading	Another Heading
row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

- HTML list
  - <ul>, <ol>, <li>, .....

```
<html>  
  
<body>  
  
<ol>  
<li>Coffee</li>  
<li>Milk</li>  
</ol>  
  
</body>  
</html>
```

- 1. Coffee
- 2. Milk

```
<html>  
  
<body>  
  
<ul>  
<li>Coffee</li>  
<li>Milk</li>  
</ul>  
  
</body>  
</html>
```

- Coffee
- Milk

- block level element vs. inline element
- block level element
  - `<h1>`, `<p>`, `<ul>`, `<table>`, `<div>`, `<span>`
  - Browser present them in a new line
- inline element
  - `<b>`, `<td>`, `<a>`, `<img>`
  - Browser usually present them without a new line char.

- HTML layout
  - <div>, <table>

```
<!DOCTYPE html>
<html>
<body>

<div id="container">
  <div id="header">
    <h1>Main Title of Web Page</h1>
  </div>
  <div id="menu">
    <h2>Menu</h2>
    <ul>
      <li>HTML</li>
      <li>CSS</li>
      <li>JavaScript</li>
    </ul>
  </div>
  <div id="content">Content goes here</div>
  <div id="footer">Copyright W3School.com.cn</div>
</div>

</body>
</html>
```

## Main Title of Web Page

Content goes here

### Menu

- HTML
- CSS
- JavaScript

Copyright W3School.com.cn

- HTML form

- <form>, <input>
- type attribute

```
<html>
<body>
<form name="input" action="html_form_action.asp"
method="get">
    First name:
    <input type="text" name="firstname" />
    <br />
    Last name:
    <input type="text" name="lastname" />
    <br />
    <input type="radio" name="sex" value="male" /> Male
    <br />
    <input type="radio" name="sex" value="female" /> Female
    <br />
    <input type="checkbox" name="bike" />
    I have a bike
    <br />
    <input type="checkbox" name="car" />
    I have a car
    <br />
    <input type="submit" value="Submit" />
</form>
</body>
</html>
```

First name:

Last name:

Male

Female

I have a bike

I have a car

- HTML iframe
  - To present webpage in a webpage

```
<!DOCTYPE html>
<html>
<body>

<iframe src="/example/html/demo_iframe.html" width="200"
height="200"></iframe>

</body>
</html>
```

本页显示在  
内联框架  
中。

- **JavaScript**

```
<script type="text/javascript">  
    document.write("Hello World!")  
</script>
```

- **VBScript**

```
<script type="text/vbscript">  
    document.write("Hello World!")  
</script>
```

- <head>
  - Script, CSS, metadata
  - <title>, <base>, <link>, <meta>, <script>, <style>

标签	描述
<u>&lt;head&gt;</u>	定义关于文档的信息。
<u>&lt;title&gt;</u>	定义文档标题。
<u>&lt;base&gt;</u>	定义页面上所有链接的默认地址或默认目标。
<u>&lt;link&gt;</u>	定义文档与外部资源之间的关系。
<u>&lt;meta&gt;</u>	定义关于 HTML 文档的元数据。
<u>&lt;script&gt;</u>	定义客户端脚本。
<u>&lt;style&gt;</u>	定义文档的样式信息。

- HTML 5 features:

- canvas
- video and audio
- local storage
- content tags, e.g. article, footer, header, nav, section
- form tags, e.g. calendar, date, time, email, url, search

- 浏览器支持

- 最新版本的 Safari、Chrome、Firefox 以及 Opera
- Internet Explorer 9

# HTML 5 - video

```
<!DOCTYPE HTML>
<html>
<body>

<video width="320" height="240" controls="controls">
  <source src="/i/movie.ogg" type="video/ogg">
  <source src="/i/movie.mp4" type="video/mp4">
Your browser does not support the video tag.
</video>

</body>
</html>
```



属性	值	描述
<u>autoplay</u>	Boolean	autoplay 如果出现该属性，则视频在就绪后马上播放。
<u>controls</u>	Boolean	controls 如果出现该属性，则向用户显示控件，比如播放按钮。
<u>height</u>	pixels	height 设置视频播放器的高度。
<u>loop</u>	Boolean	loop 如果出现该属性，则当媒介文件完成播放后再次开始播放。
<u>preload</u>	Boolean	preload 如果出现该属性，则视频在页面加载时进行加载，并预备播放。  如果使用 "autoplay"，则忽略该属性。
<u>src</u>	url	src 要播放的视频的 URL。
<u>width</u>	pixels	width 设置视频播放器的宽度。

# HTML 5 – video 使用DOM控制

```
<!DOCTYPE html>
<html>
<body>

<div style="text-align:center;">
  <button onclick="playPause()">播放/暂停</button>
  <button onclick="makeBig()">大</button>
  <button onclick="makeNormal()">中</button>
  <button onclick="makeSmall()">小</button>
  <br />
  <video id="video01" width="420" style="margin-top:15px;">
    <source src="/example/html5/mov_bbb.mp4" type="video/mp4" />
    <source src="/example/html5/mov_bbb.ogv" type="video/ogg" />
    Your browser does not support HTML5 video.
  </video>
</div>

<script type="text/javascript">
var myVideo=document.getElementById("video01");

function playPause()
{
if (myVideo.paused)
  myVideo.play();
else
  myVideo.pause();
}

function makeBig()
{
myVideo.width=560;
}

function makeSmall()
{
myVideo.width=320;
}

function makeNormal()
{
myVideo.width=420;
}
</script>

</body>
</html>
```



```
<!DOCTYPE HTML>
<html>
<body>

<audio controls="controls">
  <source src="/i/song.ogg" type="audio/ogg">
  <source src="/i/song.mp3" type="audio/mpeg">
Your browser does not support the audio element.
</audio>

</body>
</html>
```



属性	值	描述
<u>autoplay</u>	⌚	autoplay 如果出现该属性，则音频在就绪后马上播放。
<u>controls</u>	⌚	controls 如果出现该属性，则向用户显示控件，比如播放按钮。
<u>loop</u>	⌚	loop 如果出现该属性，则每当音频结束时重新开始播放。
<u>preload</u>	⌚	preload 如果出现该属性，则音频在页面加载时进行加载，并预备播放。 如果使用 "autoplay"，则忽略该属性。
<u>src</u>	⌚	url 要播放的音频的 URL。

# HTML 5 - Drag & Drop

```
<!DOCTYPE HTML>
<html>
<head>
<style type="text/css">
#div1 {width:198px; height:66px;padding:10px; border:1px solid #aaaaaa;}
</style>
<script type="text/javascript">
function allowDrop(ev)
{
ev.preventDefault();
}

function drag(ev)
{
ev.dataTransfer.setData("Text",ev.target.id);
}

function drop(ev)
{
ev.preventDefault();
var data=ev.dataTransfer.getData("Text");
ev.target.appendChild(document.getElementById(data));
}
</script>
</head>
<body>

<p>请把 W3School 的图片拖放到矩形中: </p>

<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)">
</div>
<br />


</body>
</html>
```

请把 W3School 的图片拖放到矩形中:



**W3School**

请把 W3School 的图片拖放到矩形中:



**W3School**

- HTML5 的 canvas 元素使用 JavaScript 在网页上绘制图像

```
<!DOCTYPE HTML>
<html>
<body>

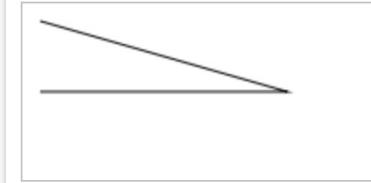
<canvas id="myCanvas" width="200" height="100" style="border:1px
solid #c3c3c3;">
Your browser does not support the canvas element.
</canvas>

<script type="text/javascript">

var c=document.getElementById("myCanvas");
var ctxt=c.getContext("2d");
ctxt.moveTo(10,10);
ctxt.lineTo(150,50);
ctxt.lineTo(10,50);
ctxt.stroke();

</script>

</body>
</html>
```



- HTML5 的 canvas 元素使用 JavaScript 在网页上绘制图像

```
<!DOCTYPE HTML>
<html>
<head>
<style type="text/css">
body
{
font-size:70%;
font-family:verdana,helvetica,arial,sans-serif;
}
</style>

<script type="text/javascript">
function cnvs_getCoordinates(e)
{
x=e.clientX;
y=e.clientY;
document.getElementById("xycoordinates").innerHTML="Coordinates: ("+
+ x + "," + y + ")";
}

function cnvs_clearCoordinates()
{
document.getElementById("xycoordinates").innerHTML="";
}
</script>
</head>

<body style="margin:0px;">

<p>把鼠标悬停在下面的矩形上可以看到坐标: </p>

<div id="coodiv"
style="float:left;width:199px;height:99px;border:1px solid #c3c3c3"
onmousemove="cnvs_getCoordinates(event)"
onmouseout="cnvs_clearCoordinates()"></div>
<br />
<br />
<br />
<div id="xycoordinates"></div>

</body>
</html>
```

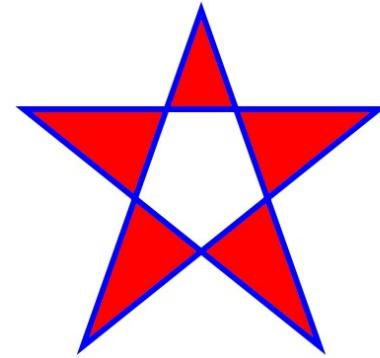


- Scalable Vector Graphics

```
<!DOCTYPE html>
<html>
<body>

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" height="190">
    <polygon points="100,10 40,180 190,60 10,60 160,180"
        style="fill:red;stroke:blue;stroke-width:3;fill-rule:evenodd;">
    />
</svg>

</body>
</html>
```



# HTML 5 – Geolocation API

```
<!DOCTYPE html>
<html>
<body>
<p id="demo">点击这个按钮，获得您的坐标: </p>
<button onclick="getLocation()">试一下</button>
<script>
var x=document.getElementById("demo");
function getLocation()
{
    if (navigator.geolocation)
    {

        navigator.geolocation.getCurrentPosition(showPosition,showError);
    }
    else{x.innerHTML="Geolocation is not supported by this
browser.";}
}
function showPosition(position)
{
    x.innerHTML="Latitude: " + position.coords.latitude +
    "<br />Longitude: " + position.coords.longitude;
}
function showError(error)
{
    switch(error.code)
    {
        case error.PERMISSION_DENIED:
            x.innerHTML="User denied the request for Geolocation."
            break;
        case error.POSITION_UNAVAILABLE:
            x.innerHTML="Location information is unavailable."
            break;
        case error.TIMEOUT:
            x.innerHTML="The request to get user location timed out."
            break;
        case error.UNKNOWN_ERROR:
            x.innerHTML="An unknown error occurred."
            break;
    }
}
</script>
</body>
</html>
```

Latitude: 31.021164888744444  
Longitude: 121.42027445601553

试一下

# HTML 5 – Web Storage

- HTML5 提供了两种在客户端存储数据的新方法：
  - localStorage - 没有时间限制的数据存储
  - sessionStorage - 针对一个 session 的数据存储

```
<!DOCTYPE HTML>
<html>
<body>

<script type="text/javascript">

if (localStorage.pagecount)
{
    localStorage.pagecount=Number(localStorage.pagecount) +1;
}
else
{
    localStorage.pagecount=1;
}
document.write("Visits: " + localStorage.pagecount + " time(s).");

</script>

<p>刷新页面会看到计数器在增长。</p>

<p>请关闭浏览器窗口，然后再试一次，计数器会继续计数。</p>

</body>
</html>
```

Visits: 8 time(s).

刷新页面会看到计数器在增长。

请关闭浏览器窗口，然后再试一次，计数器会继续计数。

- HTML5 提供了两种在客户端存储数据的新方法：
  - localStorage - 没有时间限制的数据存储
  - sessionStorage - 针对一个 session 的数据存储

```
<!DOCTYPE HTML>
<html>
<body>

<script type="text/javascript">

if (sessionStorage.pagecount)
{
    sessionStorage.pagecount=Number(sessionStorage.pagecount) +1;
}
else
{
    sessionStorage.pagecount=1;
}
document.write("Visits " + sessionStorage.pagecount + " time(s)
this session.");

</script>

<p>刷新页面会看到计数器在增长。</p>

<p>请关闭浏览器窗口，然后再试一次，计数器已经重置了。</p>

</body>
</html>
```

Visits 1 time(s) this session.

刷新页面会看到计数器在增长。

请关闭浏览器窗口，然后再试一次，计数器已经重置了。

- HTML5 引入了应用程序缓存，这意味着 web 应用可进行缓存，并可在没有因特网连接时进行访问。
- 每个指定了 manifest 的页面在用户对其访问时都会被缓存

```
<!DOCTYPE html>
<html manifest="/example/html5/demo_html.appcache">
<body>
<script type="text/javascript" src="/example/html5/demo_time.js">
</script>
<p id="timePara"><button onclick="getDateTime()">获得日期和事件
</button></p>
<p></p>
<p>请打开<a href="/example/html5/html5_html_manifest.html"
target="_blank">这个页面</a>, 然后脱机浏览, 重新加载页面。页面中的脚本和图像
依然可用。</p>
</body>
</html>
```

获得日期和事件

**W3School.com.cn**

领先的 Web 技术教程

请打开[这个页面](#), 然后脱机浏览, 重新加载页面。页面中的脚本和图像依然可用。

- manifest 文件是简单的文本文件，它告知浏览器被缓存的内容（以及不缓存的内容）。manifest 文件可分为三个部分：
  - **CACHE MANIFEST** - 在此标题下列出的文件将在首次下载后进行缓存
  - **NETWORK** - 在此标题下列出的文件需要与服务器的连接，且不会被缓存
  - **FALLBACK** - 在此标题下列出的文件规定当页面无法访问时的回退页面（比如 404 页面）

CACHE MANIFEST

# 2012-02-21 v1.0.0

/theme.css

/logo.gif

/main.js

NETWORK:

login.asp

FALLBACK:

/html5/ /404.html

- web worker 是运行在后台的 JavaScript， 不会影响页面的性能

```
<!DOCTYPE html>
<html>
<body>

<p>计数: <output id="result"></output></p>
<button onclick="startWorker()">开始 Worker</button>
<button onclick="stopWorker()">停止 Worker</button>
<br /><br />

<script>
var w;

function startWorker()
{
if(typeof(Worker)!=="undefined")
{
if(typeof(w)=="undefined")
{
w=new Worker("/example/html5/demo_workers.js");
}
w.onmessage = function (event) {
document.getElementById("result").innerHTML=event.data;
};
}
else
{
document.getElementById("result").innerHTML="Sorry, your browser
does not support Web Workers...";
}
}

function stopWorker()
{
w.terminate();
}
</script>

</body>
</html>
```

计数: 12

[开始 Worker](#) | [停止 Worker](#)

- HTML5 服务器发送事件 (server-sent event) 允许网页获得来自服务器的更新

```
<!DOCTYPE html>
<html>
<body>
<h1>获得服务器更新</h1>
<div id="result"></div>

<script>
if(typeof(EventSource)!=="undefined")
{
  var source=new EventSource("/example/html5/demo_sse.php");
  source.onmessage=function(event)
  {
    document.getElementById("result").innerHTML+=event.data + "<br />";
  };
}
else
{
  document.getElementById("result").innerHTML="抱歉, 您的浏览器不支持 server-sent 事件 ...";
}
</script>

</body>
</html>
```

## PHP 代码 (demo\_sse.php):

```
<?php
header('Content-Type: text/event-stream');
header('Cache-Control: no-cache');

$time = date('r');
echo "data: The server time is: {$time}\n\n";
flush();
?>
```

# HTML 5 – Form: Input

- HTML5 拥有多个新的表单输入类型
- Email

The screenshot shows a browser's developer tools interface with a code editor and a preview pane. The code editor contains the following HTML:

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
E-mail: <input type="email" name="user_email" /><br />
<input type="submit" />
</form>

</body>
</html>
```

The preview pane shows a simple form with a label "E-mail:" followed by an input field containing "sjtu". A red speech bubble above the input field says "输入电子邮件地址".

- URL

The screenshot shows a browser's developer tools interface with a code editor and a preview pane. The code editor contains the following HTML:

```
<!DOCTYPE HTML>
<html>
<body>

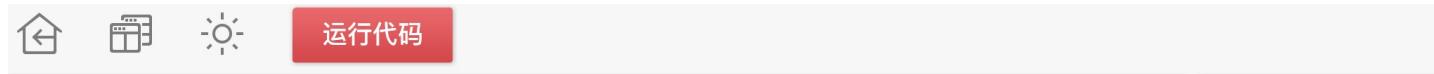
<form action="/example/html5/demo_form.asp" method="get">
Homepage: <input type="url" name="user_url" /><br />
<input type="submit" />
</form>

</body>
</html>
```

The preview pane shows a simple form with a label "Homepage:" followed by an input field containing "a". A red speech bubble above the input field says "输入 URL".

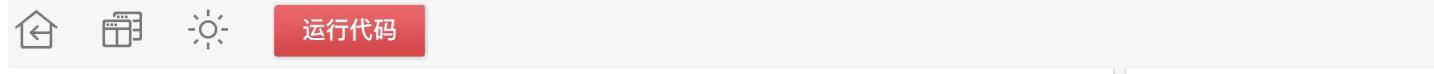
# HTML 5 – Form: Input

- HTML5 拥有多个新的表单输入类型
- Number



Points:  提交

- Range



Points:  提交

# HTML 5 – Form: Input

- Date Pickers

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Date: <input type="date" name="user_date" />
<input type="submit" />
</form>

</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Month: <input type="month" name="user_date" />
<input type="submit" />
</form>

</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Week: <input type="week" name="user_date" />
<input type="submit" />
</form>

</body>
</html>
```

Date: 2020/02/17

2020年02月

周日	周一	周二	周三	周四	周五	周六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

Month: ----年--月

2020年02月

周日	周一	周二	周三	周四	周五	周六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

Week: ----年第--周

2020年02月

周	周日	周一	周二	周三	周四	周五	周六
5	26	27	28	29	30	31	1
6	2	3	4	5	6	7	8
7	9	10	11	12	13	14	15
8	16	17	18	19	20	21	22
9	23	24	25	26	27	28	29

# HTML 5 – Form: Input

- Date Pickers

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Time: <input type="time" name="user_date" />
<input type="submit" />
</form>

</body>
</html>
```

Time: 上午 12:23 提交

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Date and time: <input type="datetime-local" name="user_date" />
<input type="submit" />
</form>

</body>
</html>
```

Date and time: 年 / 月 / 日 ----:-- 提交

2020年02月

周日	周一	周二	周三	周四	周五	周六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

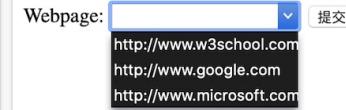
# HTML 5 – 表单元素

- **datalist** 元素规定输入域的选项列表

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Webpage: <input type="url" list="url_list" name="link" />
<datalist id="url_list">
    <option label="W3School" value="http://www.w3school.com.cn" />
    <option label="Google" value="http://www.google.com" />
    <option label="Microsoft" value="http://www.microsoft.com" />
</datalist>
<input type="submit" />
</form>

</body>
</html>
```



- **keygen** 元素的作用是提供一种验证用户的可靠方法

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Username: <input type="text" name="usr_name" />
Encryption: <keygen name="security" />
<input type="submit" />
</form>

</body>
</html>
```

Username:  Encryption:  2048 (高级)

# HTML 5 – 表单属性

- **autocomplete** 属性规定 form 或 input 域应该拥有自动完成功能

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get" autocomplete="on">
First name: <input type="text" name="fname" /><br />
Last name: <input type="text" name="lname" /><br />
E-mail: <input type="email" name="email" autocomplete="off" /><br />
<input type="submit" />
</form>
```

<p>请填写并提交此表单，然后重载页面，来查看自动完成功能是如何工作的。</p>  
<p>请注意，表单的自动完成功能是打开的，而 e-mail 域是关闭的。</p>

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



请填写并提交此表单，然后重载页面，来查看自动完成功能是如何工作的。  
请注意，表单的自动完成功能是打开的，而 e-mail 域是关闭的。

- **autofocus** 属性规定在页面加载时，域自动地获得焦点

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
User name: <input type="text" name="user_name" autofocus="autofocus" />
<input type="submit" />
</form>

</body>
</html>
```

User name:  提交

# HTML 5 – 表单属性

- form 属性规定输入域所属的一个或多个表单

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get" id="user_form">
First name:<input type="text" name="fname" />
<input type="submit" />
</form>
```

<p>下面的输入域在 form 元素之外，但仍然是表单的一部分。</p>

```
Last name: <input type="text" name="lname" form="user_form" />

</body>
</html>
```

- 表单重写属性（form override attributes）允许重写 form 元素的某些属性设定

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get" id="user_form">
E-mail: <input type="email" name="userid" /><br />
<input type="submit" value="Submit" /><br />
<input type="submit" formaction="/example/html5/demo_admin.asp" value="Submit as
admin" /><br />
<input type="submit" formnovalidate="true" value="Submit without validation" /><br />
</form>

</body>
</html>
```

First name:  提交

下面的输入域在 form 元素之外，但仍然是表单的一部分。

Last name:

E-mail:

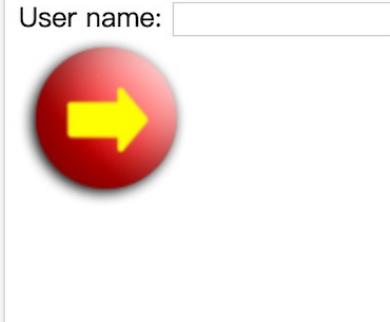
# HTML 5 – 表单属性

- height 和 width 属性规定用于 image 类型的 input 标签的图像高度和宽度

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
User name: <input type="text" name="user_name" /><br />
<input type="image" src="/i/eg_submit.jpg" width="99" height="99" />
</form>

</body>
</html>
```

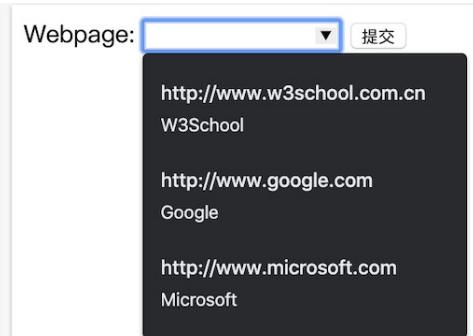


- list 属性规定输入域的 datalist

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Webpage: <input type="url" list="url_list" name="link" />
<datalist id="url_list">
    <option label="W3School" value="http://www.w3school.com.cn" />
    <option label="Google" value="http://www.google.com" />
    <option label="Microsoft" value="http://www.microsoft.com" />
</datalist>
<input type="submit" />
</form>

</body>
</html>
```



- min、max 和 step 属性用于为包含数字或日期的 input 类型规定限定（约束）

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Points: <input type="number" name="points" min="0" max="10" step="3"/>
<input type="submit" />
</form>

</body>
</html>
```

Points:  提交

- multiple 属性规定输入域中可选择多个值

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Select images: <input type="file" name="img" multiple="multiple" />
<input type="submit" />
</form>

<p>当您浏览文件时，请试着选择多个文件。</p>

</body>
</html>
```

Select images:  未选择任何文件

当您浏览文件时，请试着选择多个文件。



# HTML 5 – 表单属性

- `novalidate` 属性规定在提交表单时不应该验证 form 或 input 域

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get" novalidate="novalidate">
E-mail: <input type="email" name="user_email" />
<input type="submit" />
</form>

</body>
</html>
```

E-mail: abf 提交

服务器接收到的用户输入为：

**user\_email=abf**

- `pattern` 属性规定用于验证 input 域的模式(pattern)

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Country code: <input type="text" name="country_code" pattern="[A-z]{3}" title="Three letter country code" />
<input type="submit" />
</form>

</body>
</html>
```

Country code: A\_3 提交

! 请与所请求的格式保持一致。  
Three letter country code

# HTML 5 – 表单属性

- placeholder 属性提供一种提示(hint)，描述输入域所期待的值

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
<input type="search" name="user_search" placeholder="Search W3School" />
<input type="submit" />
</form>

</body>
</html>
```

Search W3School 提交

- required 属性规定必须在提交之前填写输入域(不能为空)

```
<!DOCTYPE HTML>
<html>
<body>

<form action="/example/html5/demo_form.asp" method="get">
Name: <input type="text" name="usr_name" required="required" />
<input type="submit" />
</form>

</body>
</html>
```

Name:  提交  
! 请填写此字段。

- CSS (Cascading Style Sheet)

- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS3 is the latest CSS standard

```
<!DOCTYPE html>
<html>
<head>
<style>
div{
  width:100px;
  height:75px;
  background-color:yellow;
  border:1px solid black;
}
div#div2{
  transform:rotate(30deg);
  -ms-transform:rotate(30deg); /* IE 9 */
  -moz-transform:rotate(30deg); /* Firefox */
  -webkit-transform:rotate(30deg); /* Safari and Chrome */
  -o-transform:rotate(30deg); /* Opera */
}
</style>
</head>
```



```
<body>
<div>你好。这是一个 div 元素。</div>
<div id="div2">你好。这是一个 div 元素。
</div>
</body>
</html>
```

## Welcome to My Homepage

Use the menu to select different Stylesheets

- Stylesheet 1
- Stylesheet 2
- Stylesheet 3
- Stylesheet 4
- No Stylesheet

### Side-Bar

Loreum ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

## Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links:

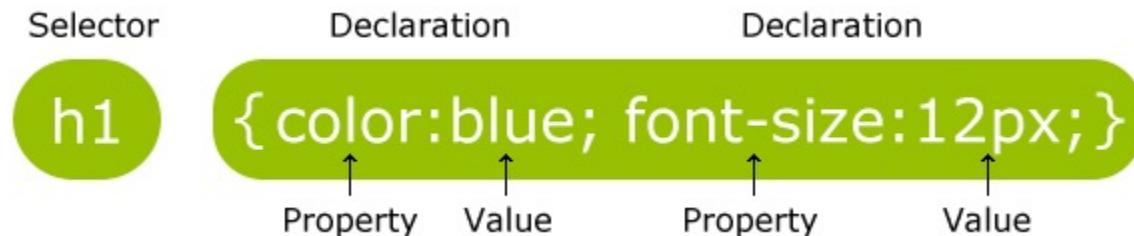
[Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

## No Styles

This page uses DIV elements to group different sections of the HTML page.

Click here to see how the page looks like with no stylesheet:

- A CSS rule-set consists of a selector and a declaration block:
  - The selector points to the HTML element you want to style.
  - The declaration block contains one or more declarations separated by semicolons.
  - Each declaration includes a CSS property name and a value, separated by a colon.
  - A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.



- The element Selector

- The element selector selects elements based on the element name.

```
p {  
    text-align: center;  
    color: red;  
}
```

- The id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.
  - The id of an element should be unique within a page, so the id selector is used to select one unique element!

```
#para1 {  
    text-align: center;  
    color: red;  
}
```

- The class Selector

- The class selector selects elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the name of the class.

```
.center {  
    text-align: center;  
    color: red;  
}
```

```
<html>  
<head>  
    <style type="text/css">  
        .center {  
            text-align:center;  
            color:red;}  
    </style>  
</head>  
<body>  
    <h1 class="center">Header 1</h1>  
    <p>A paragraph.</p>  
    <p class="center">Note that this is an important  
    paragraph.</p>  
</body>  
</html>
```

## Header 1

A paragraph.

Note that this is an important paragraph.

- The class Selector

- You can also specify that only specific HTML elements should be affected by a class.

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>

</body>
</html>
```

This heading will not be affected

This paragraph will be red and center-aligned.

# Grouping Selector

```
h1 {  
    text-align: center;  
    color: red;  
}
```

```
h2 {  
    text-align: center;  
    color: red;  
}
```

```
p {  
    text-align: center;  
    color: red;  
}
```



```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

- Comments

- are used to explain the code, and may help when you edit the source code at a later date.
- Comments are ignored by browsers.
- A CSS comment starts with /\* and ends with \*/. Comments can also span multiple lines:

```
p {  
    color: red;  
    /* This is a single-line comment */  
    text-align: center;  
}
```

```
/* This is  
a multi-line  
comment */
```

# Three Ways to Insert CSS

- There are three ways of inserting a style sheet:
  - External style sheet
  - Internal style sheet
  - Inline style

- With an external style sheet, you can change the look of an entire website by changing just one file!
  - Each page must include a reference to the external style sheet file inside the `<link>` element. The `<link>` element goes inside the `<head>` section:

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

# External Style Sheet

- An external style sheet can be written in any text editor.
  - The file should **not** contain any html tags.
  - The style sheet file must be saved with a **.css** extension.
  - Here is how the "myStyle.css" looks:

```
body {  
    background-color: lightblue;  
}
```

```
h1 {  
    color: navy;  
    margin-left: 20px;  
}
```

# External Style Sheet

**搜 狐**  
**SOHU.com**

搜狗 输入法 浏览器 地图 邮件 博客 社区 我说两句 微门户 大视野 视频 游戏 新天龙 星际战甲 手机搜狐 手机游戏 e购房  
新闻 军事 文化 历史 读书 评论 体育 NBA CBA 中超 高尔夫 财经 理财 股票 基金 科技 数码 手机 创客 汽车 二手车  
时尚 健康 旅游 教育 出国 高考 母婴 美食 星座 公益 播客 娱乐 美剧 音乐 应用 房产 二手房 家居 彩票 城市

金吉列留学免费评估  
搜狐媒体平台入驻直达  
搜狐媒体平台官方账号  
苏宁1212天下惠！  
尖货抢先看早买更便宜  
欢迎入驻搜狐媒体平台  
教育 澳大利亚留学月  
雾霾催生新移民潮  
出国 拼爹不如出国？  
低龄出国留学指南

院线大片任性看！ 熊出没之熊大快跑 植物大战僵尸王！ 泡泡龙（星球版）  
雷电战机之星战版 滑雪大冒险西游版 地下城勇士神枪手 搜狐手游权威媒体

总价200万内精装2-3房  
浦东张江新房120万起  
嘉定37平LOFT单价9988元  
11号线小户型9988元起  
宝山85平3房12月开盘  
闵行12.19推90平洋房  
嘉定97平三房203万起  
首付36万起入住精装房  
12月楼市233盘享优惠  
东郊华宅最高优惠20万



百度翻译 Elsevier Editorial SystemTM Elsevier Editorial SystemTM 百度地图

我的 搜狐 用户名:邮箱/手机号 登录 邮件 相册 说两句 手机看新闻

- 习近平访南非 签419亿大单力挺当地经济
- 习近平和彭丽媛出席中非峰会欢迎宴会 | 漫评
- 李克强谈“国人双11海淘”：我们供给没跟上 | 漫画
- 张德江出席铁路奠基仪式 | 全国政协双周座谈会
- 搜狐视频 | 正在直播《高品格单恋》见面会
- 香港国际创客节开幕 助力香港和内地合作发展
- 教育部:2016年高考26个省份将用全国卷(名单)
- 首个落马巡视组长涉案2亿 家中现金发霉
- 上海厅官6年前享副部级待遇 现位10年14变
- 港校有人抨击简体字 内地生晒书法回应(图)
- 日本将出5亿助华绿化 日媒:减少“越境污染”
- 高铁换大米！中泰两国连签两项重磅协议
- “油城”克拉玛依有多富?私家车太多需增警力
- 打普京脸？伊拉克库尔德人：油罐车我们的
- 土耳其总理态度生变罕见对俄示好 划两条底线
- 大批轰6K穿越美日封锁线 美媒：没法打航母
- 国产准航母大喜讯：“心脏”堪比西北风舰(图)
- 北京空军机场现大批歼-7 排长列望不到头(图)
- 藏传佛教高僧看“白马纯林供品”被翻拍混乐

# Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.

- Internal styles are defined within the `<style>` element, inside the `<head>` section of an HTML page:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-color: linen;}
      h1 {
        color: maroon;
        margin-left: 40px;}
    </style>
  </head>
<body>
  <h1>This is a heading</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

This is a heading

This is a paragraph.

- An inline style may be used to apply a unique style for a single element.
  - To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

```
<!DOCTYPE html>  
<html>  
<body>  
  <h1 style="color:blue;margin-left:30px;">  
    This is a heading.  
  </h1>  
  <p>This is a paragraph.</p>  
</body>  
</html>
```

This is a heading.

This is a paragraph.

- If some properties have been defined for the same selector (element) in different style sheets
  - the value from the last read style sheet will be used.
- Assume that an **external** style sheet has the following style for the `<h1>` element:

```
h1 {  
    color: navy;  
}
```

- then, assume that an **internal** style sheet also has the following style for the `<h1>` element:

```
h1 {  
    color: orange;  
}
```

# Multiple Style Sheet

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style>
h1 {
  color: orange;
}
</style>
</head>
```

---

**This is a heading**

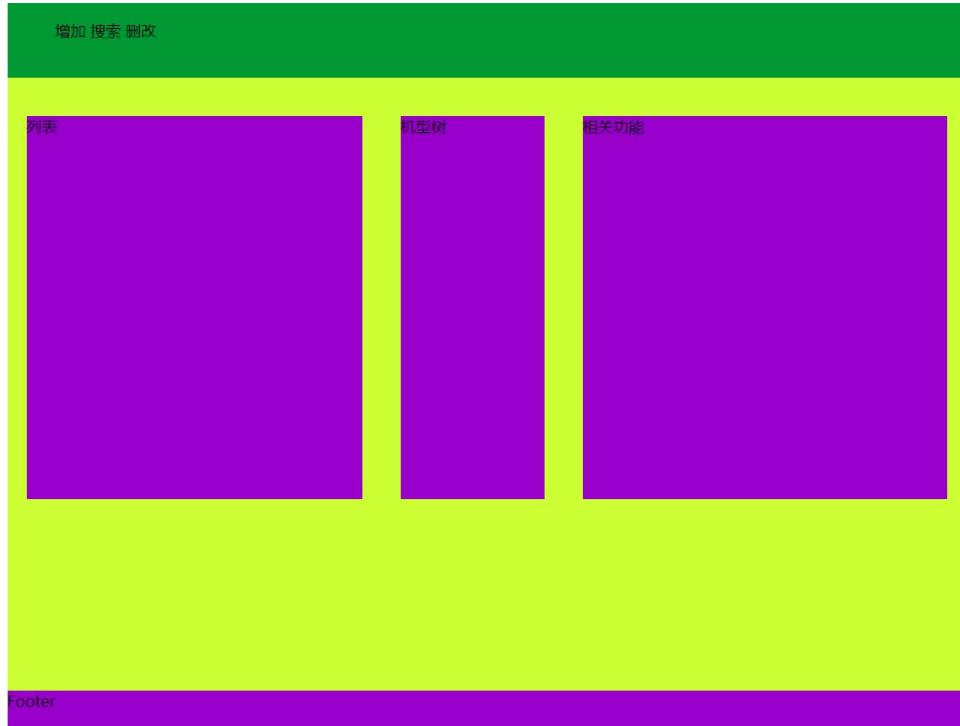
```
<head>
<style>
h1 {
  color: orange;
}
</style>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

**This is a heading**

- Generally speaking we can say that all the styles will "**cascade**" into a new "virtual" style sheet by the following rules, where number one has the highest priority:
  - Inline style (inside an HTML element)
  - External and internal style sheets (in the head section)
  - Browser default
- So,
  - an inline style (inside a specific HTML element) has the highest priority,
  - which means that it will override a style defined inside the <head> tag, or in an external style sheet, or a browser default value.

- 一小时搞定DIV+CSS布局-固定页面开度布局 - Slevens - 博客园
  - <https://www.cnblogs.com/slevens/p/6829118.html>
- DIV+CSS布局中主要CSS属性介绍:
  - Float:
    - <div>标签默认一行只能显示一个，而使用Float属性可以实现一行显示多个div的功能。
  - Margin:
    - Margin属性用于设置两个元素之间的距离。
  - Padding:
    - Padding属性用于设置一个元素的边框与其内容的距离。
  - Clear:
    - 使用Float属性设置一行有多个DIV后（多列），在下一行开始之前使用Clear属性清除浮动，否则上面的布局会影响到下面。

```
<!DOCTYPE html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
<html xmlns="http://www.w3.org/1999/xhtml">
<body>
  <div id="Container">
    <div id="Header">
      <div id="logo">@*增加 搜索 删改</div>
    </div>
    <div id="Content">
      <div id="Content-Left">列表</div>
      <div id="Content-Main">树</div>
      <div id="Content-Right">相关功能</div>
    </div>
    <div class="Clear"></div>
    <div id="Footer">Footer</div>
  </div>
</body>
</html>
```



```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>DIV+CSS布局教程</title>
<style type="text/css">
#Container {
    width: 1000px;
    margin: 0 auto; /*设置整个容器在浏览器中水平居中*/
    background: #CF3;
}
#Header {
    height: 80px;
    background: #093;
}
#logo {
    padding-left: 50px;
    padding-top: 20px;
    padding-bottom: 50px;
}
```

```
#Content {  
    height: 600px; /*一般使用overflow:auto; */  
    margin-top: 20px; /*设置content与上面header元素之间的距离*/  
    background: #OFF;  
}  
#Content-Left {  
    height: 400px;  
    width: 350px;  
    margin: 20px;  
    float: left; /*设置浮动，实现多列效果，div+Css布局中很重要的*/  
    background: #90C;  
}  
#Content-Main {  
    height: 400px;  
    width: 150px;  
    margin: 20px;  
    float: left  
    background: #90C;  
}
```

```
#Content-Right {  
    height: 400px;  
    width: 380px;  
    margin: 20px;  
    float: right  
    background: #90C;  
}  
  
#Footer {  
    height: 40px;  
    background: #90C;  
    margin-top: 20px;  
}  
  
.Clear {  
    clear: both;  
}  
  
</style>  
</head>
```

- 迭代1总目标：运用HTML、CSS和React/Vue开发前端静态页面的网站，第5周验收
- 作业1需完成：
  - 根据《互联网应用开发技术》课程大作业“e-Book系统”要求，用HTML 5 和 CSS 3设计用户界面，应该包括登录页面、网站首页、书籍浏览页、书籍详情页、下订单页等，本次作业并非定稿版，以后可以持续优化。
  - 在工程中往往使用到多个CSS，在本次作业中，如果你在你的页面上使用多个CSS样式表，那么你应该能够正确地显示你想要的样式效果，能够解决在不同的CSS中样式定义重复的问题；
  - 具体的样式表可以是你自己编写的，也可以是你引用框架的CSS，但是引用框架CSS时，应该把样式表下载到本地，从本地文件系统中引用，这样你的页面不用每次在加载时都必须到网上去加载样式表。
- \* 请将你的HTML、CSS和图片文件等**压缩后上传**

- HTML5
  - <https://www.w3.org/TR/html/>
- HTML5 教程
  - <https://www.w3school.com.cn/html5/index.asp>
- CSS3教程
  - <https://www.w3school.com.cn/css3/index.asp>



- *Web*开发技术
- *Web Application Development*

# Thank You!