

# HTML，主讲：汤小洋

---

## 一、HTML简介

### 1. HTML是什么

HTML: Hyper Text Markup Language 超文本标记（标签）语言

由各种标签组成，用来制作网页，告诉浏览器该如何显示页面

### 2. 作用

- 制作网页，控制网页和内容的显示
- 插入图片、音乐、视频、动画等多媒体
- 通过链接检索信息
- 使用表单获取用户信息，实现交互

### 3. 版本

W3C: World Wide Web Consortium 万维网联盟，制定Web技术相关标准和规范的组织，HTML就是W3C制定的标准

两个版本：HTML 4.01、HTML5

官网：<https://www.w3cschool.cn/>

### 4. 后缀名

HTML文件是以.html或.htm结尾

## 二、HTML文档结构

### 1. 基本结构

#### 1.1 简介

- HTML标签是由尖括号包围的关键词，如 `<html>`，通过都是成对出现的，如 `<html> </html>`
- 以 `<html>` 为根标签，包含：`<head>` 头部和 `<body>` 主体部分
- 头部提供关于网页的相关信息，如标题、文档类型、字符编码、关键字等摘要信息

- 主体部分提供网页的具体内容，真正显示在页面中
- 合理的进行缩进
- 标签名不区分大小写，但一般都习惯全小写

## 1.2 开发工具

记事本notepad、Sublime、Notepad++、Dreamweaver、VSCode、WebStorm等

使用步骤：

1. 新建文件（Ctrl+N），然后保存（Ctrl+S），指定后缀名为.html
2. 编写HTML代码
3. 在浏览器中打开文件

使用技巧：

- 先保存再写代码，否则没有颜色提示
- 创建一个文件夹，用于保存所有的网页内容，将文件夹拖放到Sublime中，便于管理
- 显示/隐藏侧边栏

方式1：查看——>侧边栏——>显示/隐藏侧边栏

方式2：先按Ctrl+K，紧接着按B

- 显示多栏

方式1：查看——>布局——>列数：2列

方式2：按Alt+Shift+2

## 1.3 浏览器

常见浏览器：IE微软、Chrome谷歌、Firefox火狐、Safari苹果

浏览器的作用是读取HTML文件，并以网页的形式显示

浏览器不会直接显示HTML标签，而是使用标签来解释网页的内容

## 2. 标签

### 2.1 标签组成

一个完整的HTML标签的组成：

```
<标签名 属性名="属性值">内容</标签名>
```

属性值要用引号引起来，一般使用双引号

## 2.2 标签分类

根据标签是否关闭，分为：关闭型、非关闭型

- 关闭型：有结束标签，即成对出现

```
<title></title>
<body></body>
<h1></h1>
```

- 非关闭型：没有结束标签

```
<meta>
<br>
<hr>
```

根据标签是否独占一行，分为：块级标签、行级标签

- 块级标签：显示为块状，独自占一行

```
<h1></h1>
<hr>
```

- 行级标签：在行内显示，可以与其他内容在同一行显示

```
<span></span>
```

## 3. 注释

注释在浏览器中是不会显示，是用来标注解释，但通过查看源代码是可以看到

```
<!-- 注释内容 -->
```

## 4. 实体字符

也称为特殊字符，用于显示一些特殊符号，如：< > & 空格 等

语法：

```
&实体字符名称;
```

常用实体字符：

|  |   |      |              |
|--|---|------|--------------|
| &lt;                                     | < | 小于号  | less than    |
| &gt;                                     | > | 大于号  | greater than |
| &nbsp;                                   |   | 空格   | space        |
| 对于连续的空白字符（包含空格、缩进、换行等），在浏览器中显示时只会显示为一个空格 |   |      |              |
| &amp;                                    | & | 与    |              |
| &quot;                                   | " | 双引号  |              |
| &copy;                                   | © | 版权符号 | copyright    |
| &reg;                                    | ® | 注册符号 | register     |
| &times;                                  | × | 关闭符号 |              |

注：实体字符名称是区分大小写，即大小写敏感

## 5. 文档类型

在HTML文档的第一行，使用<!DOCTYPE>声明HTML文档的类型

用来告诉浏览器页面的文档类型，简单来说，用来指定使用的HTML版本规范

目前基本上都使用HTML5: <!DOCTYPE html>

## 三、常用标签

### 1. 基本标签

| 标签         | 含义       | 说明                     |
|------------|----------|------------------------|
| br         | 换行标签     | 非关闭型                   |
| p          | 段落标签     | 关闭型，块级标签，前后有明显的间隔      |
| h1、h2...h6 | 标题标签     | 按照h1到h6逐渐变小，块级标签，加粗显示  |
| pre        | 预格式化文本标签 | 保留编码时的格式               |
| div        | 分区标签     | 常作用容器来使用，一般用来页面布局，块级标签 |
| span       | 范围标签     | 默认没有任何效果，一般用来设置行内的特殊样式 |
| ol、li      | 有序列表     | 有顺序的项目列表               |
| ul、li      | 无序列表     | 无顺序的项目列表               |
| dl、dt、dd   | 定义列表     | 对术语、图片等进行描述定义的列表       |
| hr         | 水平线标签    | 非关闭型、块级标签              |
| img        | 图像标签     | 非关闭型、行级标签              |

#### 1.1 有序列表

ol: ordered list

li: list item

默认使用阿拉伯数字，从1开始标记，可以通过属性进行修改

- **type**属性：设置列表前的符号标记，取值：数字1（默认）、字母a或A、罗马数字i或I
- **start**属性：设置起始值，值必须是数字

## 1.2 无序列表

ul: unordered list

li: list item

默认使用实心圆作为符号标记，可以通过属性进行修改

- **type**属性：设置列表前的符号标记，取值：**disc**实心圆（默认）、**circle**空心圆、**square**正文形、**none**不显示符号

## 1.3 定义列表

dl: definition list

dt: definition title

dd: definition description

## 1.4 水平线标签

hr: horizontal

常用属性：

- **color**: 颜色

两种写法：

颜色名称，如red、green、blue、white、black、pink、orange等

16进制的RGB: Red、Green、Blue，用法：#RRGGBB，每个颜色的范围是0-255，转换为16进制为00-FF

如#FF0000、#00FF00、#0000FF、#FFFFFF、#000000、#CCCCCC、#FF7300

- **size** 粗细，数值
- **width** 宽度

两种写法：

像素，绝对值（固定值）

百分比，相对值，相对于父容器的宽度的百分比

- **align** 对齐

取值：center（默认）、left、right

## 1.5 图像标签

img: image

常见图片格式: .jpg、.png、.gif、.bmp

常用属性:

- src: source 指定图片的路径（来源），必须有  
如果图片与html页面在同一个文件夹中，可以直接写图片名称  
习惯上，会将多个图片放到统一的文件夹中，如images，此时需要在图片名称的前面添加文件夹的名称  
images/
- alt: 当图片无法显示时的提示信息
- title: 当鼠标停留在图片上时显示的提示信息
- width/height: 设置图片的宽和高  
默认图片原尺寸显示  
如果只设置其中的一个，则另一个按比例缩放  
如果同时设置宽和高，可能会导致图片变形  
两种写法:  
像素，绝对值（固定值）  
百分比，相对值，相对于父容器的尺寸的百分比

## 2. 其他标签

| 标签      | 含义            | 说明                                    |
|---------|---------------|---------------------------------------|
| i       | <u>斜体</u> 标签  | italic                                |
| em      | <u>强调</u> 的内容 | 在浏览器中显示时一般为斜体                         |
| address | <u>地址</u>     | 在浏览器中显示时一般为斜体，块级标签                    |
| b       | <u>加粗</u> 显示  | bold                                  |
| strong  | 强调的内容         | 在浏览器中显示时一般为 <u>加粗</u>                 |
| del     | <u>删除线</u>    | delete                                |
| ins     | <u>下划线</u>    |                                       |
| sub     | 下标            | H <sub>2</sub> O                      |
| sup     | 上标            | 2 <sup>3</sup> =8                     |
| bdo     | 设置文本方向        | 通过属性dir="rtl"设置文本从右到左显示 right to left |
| abbr    | 设置文字缩写        | 通过title属性设置当鼠标停留在文字上时显示的提示信息          |
| small   | 相当于当前字号减小一号   |                                       |
| big     | 相当于当前字号增加一号   |                                       |

为了更好的语义化

### 3. 头部标签

- **meta** 定义网页的摘要信息，如字符编码、关键字、描述、作者等 还可以实现刷新跳转
  - **title** 定义网页的标题
  - **style** 定义内部CSS样式
  - **link** 引用外部CSS样式
  - **script** 定义或引用脚本
  - **base** 定义基础路径
- 默认以当前页面文件所在位置为相对路径的参照

```
<meta charset="UTF-8">
```

```
<meta name="author" content="汤小洋">
```

```
<meta http-equiv="refresh"
```

```
content="2;url=http://www.baidu.com">
```

**<noscript>** 标签提供无法使用脚本时的替代内容，比方在浏览器禁用脚本时，或浏览器不支持客户端脚本时。

<noscript> 元素可包含普通 HTML 页面的 body 元素中能够找到的所有元素。

只有在浏览器不支持脚本或者禁用脚本时，才会显示 <noscript> 元素中的内容。

实例：

```
<script>
document.write("Hello W3Cschool!")
</script>
<noscript>抱歉，您的浏览器不支持 JavaScript!</noscript>
```

### 4. 标签嵌套

一个标签中嵌套着另一个标签

标签不能瞎嵌套，如以下嵌套是错误的：

```
<p style="width: 300px;height: 300px;background: green;">
  <div style="width: 200px;height: 200px; background: blue;">world</div>
</p>
```

浏览器渲染后显示的页面代码与编码时有所不同

浏览器提供的开发工具：用来帮助开发人员查看和调试页面

如何打开：

- 在页面中右击——>检查/审查元素/查看元素
- 按F12

常用工具：

- **Elements**：从浏览器的角度查看页面，浏览器渲染页面时的结构内容
- **Console**：控制台，显示各种警告和错误信息
- **Network**：查看网络请求信息，浏览器向服务器请求了哪些资源、资源大小、加载资源花费的时间

## 四、超链接

### 1. 简介

使用超链接可以从一个页面跳转到另一个页面，实现页面间的导航

当鼠标移动到超链接文本上时，鼠标箭头会变成一只小手

超链接的有三种类型：

- 普通链接/页面间链接，跳转到另一个页面
- 锚链接，跳转到锚点

- 功能性链接，实现特殊功能

## 2. 基本用法

使用 `<a>` 标签创建超链接

语法：

```
<a href="链接地址" target="链接打开位置">链接文本或图像</a>
```

常用属性：

- href 链接地址/路径，链接地址
- target 链接打开位置，取值：

\_self（自身、当前，默认值）、\_blank（空白、新的）、\_parent（父层框架）、\_top（顶层框架）、自定义

路径分类：

- 相对路径

相对于当前文件的所在路径

不是以根开始的路径都叫相对路径

. 表示当前路径

.. 表示上一级路径

```
<a href="../../../d.html">d</a>
```

- 绝对路径

以根开始的路径

<http://www.baidu.com>

D:/Software/b.html

## 3. 锚链接

### 3.1. 简介

点击链接后跳转指定的位置（锚点 anchor）

锚链接的分类：

- 页面内的锚链接
- 页面间的锚链接

### 3.2 页面内的锚链接

步骤：

1. 定义锚点（标记）



```
<a name="锚点名称">目标位置</a>
```

## 2. 链接锚点

```
<a href="#锚点名称">链接文本</a>
```

## 3.3 页面间的锚链接

```
<a href="目标页面#锚点名称">链接文本</a>
```

## 4. 功能性链接

```
<body>
<a href="images/heihei.gif">点击此处下载图片</a>
<br>
<a href="mailto:tangxiaoyang@qq.com">联系我们</a>
<br>
<a href="javascript:alert('哈哈')">点我试试</a>
</body>
```

## 5.URL

### 5.1 简介

URL: Uniform Resource Locator 统一资源定位符，用来定位资源所在位置，最常见的就是网址

```
https://t1.huanqiu.cn/4ac5853976fa0f2a7fbdac13593ddb32.jpg
https://www.baidu.com/img/bd_logo1.png
https://www.w3cschool.cn/
```

### 5.2 组成

```
1 https:// 2 www.baidu.com:80 3 /web/html/images/logo.png 4 ?name=tom&age=21&sex=male 5 #first 6
https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=1&rsv_idx=1&rsv_sug1=10&rsv_sug7=100&bs=ssl
file://d:\Users\User\Desktop\code\day02
ftp://ftp.itany.com/note/HTML.md
https://www.w3cschool.cn/
http://xiazai.zol.com.cn/btzhongzi_soft_index/btzhongzi_page_1.html
```

一个完整的URL由8个部分组成:

1 协议 protocol, 如:

http 超文本传输协议，用来访问Web网站 HyperText Transfer Protocol

https 更加安全的协议

ftp 文件传输协议，用来访问服务器上的文件，实现文件的上传和下载 File Transfer Protocol

file 文件协议，用来访问本地文件

2 主机名 hostname 服务器地址，如www.baidu.com、ftp.itany.com

3 端口 port 位于主机名的后面，使用冒号隔开

不同的协议使用不同的端口号，如：http使用80，https使用443，ftp使用21

如果使用的是默认端口，则端口可以省略

如果使用的不是默认端口，则必须指定端口

4 路径 path 目标文件所在的路径结构，如：/web/html/images/

5 资源 resource 要访问的目标文件，如：logo.png

6 查询字符串 query string，也称为参数

在资源后面使用?开头的一组名称/值

名称和值之间以=隔开，多个之间以&隔开，如：?name=tom&age=21&sex=male

7 锚点 anchor，在资源后面使用#开头的文本，如：#first

8 身份认证 authentication，指定身份信息，如：ftp://账户:密码@ftp.itany.com/note/HTML.md

## 五、表格

### 1. 简介

表格是一个规则的行列结构，每个表格由若干行组成，每行由若干单元格组成

下面是 HTML 表格标签描述列表：

table、row、column

### 2. 基本结构

#### 2.1 table 标签

用来定义表格

常用属性：

- border 边框，默认为0
- width/height 宽度/高度
- align 水平对齐，取值：left、center、right
- bordercolor 边框颜色
- bgcolor 背景颜色
- background 背景图片
- cellspacing 间距，单元格与单元格之间的距离 default: 1
- cellpadding 边距，单元格内容与边界之间的距离

| 标签                      | 描述             |
|-------------------------|----------------|
| <u>&lt;table&gt;</u>    | 定义表格           |
| <u>&lt;th&gt;</u>       | 定义表格的表头（表头单元格） |
| <u>&lt;tr&gt;</u>       | 定义表格的行         |
| <u>&lt;td&gt;</u>       | 定义表格单元（标准单元格）  |
| <u>&lt;caption&gt;</u>  | 定义表格标题         |
| <u>&lt;colgroup&gt;</u> | 定义表格列的组        |
| <u>&lt;col&gt;</u>      | 定义用于表格列的属性     |
| <u>&lt;thead&gt;</u>    | 定义表格的页眉        |
| <u>&lt;tbody&gt;</u>    | 定义表格的主体        |
| <u>&lt;tfoot&gt;</u>    | 定义表格的页脚        |

#### 2.2 tr 标签

用来定义行，table row

常用属性：

- align 水平对齐，取值：left、center、right
- valign 垂直对齐，取值：top、middle、bottom
- bgcolor 背景颜色

- background 背景图片

## 2.3 td标签

用户定义单元格, table data

常用属性: align、valign、bgcolor、background

注: 表格必须由行组成, 行必须由单元格组成, 数据必须放到单元格中

## 3. 合并单元格      自动生成表格: `table>(tr>td{hello$}*4)*4`

也称为表格的跨行跨列

两个属性:

- rowspan  
设置单元格所跨的行数, 如rowspan=2, 表示跨越2行
- colspan  
设置单元格所跨的列数, 如colspan=4, 表示跨越4列

步骤:

1. 在跨越的单元格中设置rowspan/colspan属性
2. 将被跨越的单元格删除

必须要保证每行的实际列数是相同的, 否则表格可能会出现错乱

## 4. 高级标签

### 4.1 caption标签

表格的标签

### 4.2 thead标签

表格的头部 table head

### 4.3 th标签

表格的头部标题 table head title

一般用在thead中, 设置头部的标题, 替代td标签, 与td的区别: 加粗和居中对齐

## 4.4 tbody 标签

表格的主体 table body

## 4.5 tfoot 标签

表格的底部 table foot

# 六、表单

## 1. 简介

表单是一个包含若干表单元素的区域，用于获取不同类型的用户信息

表单元素是允许用户在表单中输入信息的元素，如：文本框、密码框、单选按钮、复选框、下拉列表、按钮等

## 2. 表单结构

### 2.1 表单语法

```
<form action="表单提交地址" method="提交方式">
  多个表单元素
</form>
```

### 2.2 form 标签

用来定义表单，可以包含多个表单元素

常用属性：

- action 提交数据给谁处理，即处理数据的程序，默认为当前页面
- method 提交数据的方式，取值：get（默认）、post

get和post的区别：

- get：以查询字符串形式提交，在地址栏中能看到，长度有限制，不安全
- post：以表单数据组形式提交，在地址栏中看不到，长度无限制，安全 有文件数据时要改
- enctype 提交数据的编码，取值：application/x-www-form-urlencoded（默认）、multipart/form-data（文件上传）

## 3. 表单元素

大多数表单元素都是使用 <input> 标签来定义的，通过设置属性type来定义不同的表单元素

```
<input type="表单元素类型" name="名称" value="初始值" size="宽度">
```

| 表单元素类型          | 含义    | 说明                         |
|-----------------|-------|----------------------------|
| <u>text</u>     | 单行文本框 | 省略时默认就是text                |
| <u>password</u> | 密码框   | 输入时以点号显示，安全                |
| radio           | 单选按钮  | 只能选择其中一个                   |
| checkbox        | 复选框   | 可以同时选择多个                   |
| submit          | 提交按钮  | 提交表单数据 由<form action="">指定 |
| reset           | 重置按钮  | 重置表单元素的初始值                 |
| image           | 图像按钮  | 可以使用图片作为按钮，也具有提交的功能        |
| button          | 普通按钮  | 默认无功能                      |
| file            | 文件选择器 | 选择要上传的文件                   |
| hidden          | 隐藏域   | 在页面上不显示，但会提交，可以用来存储数据      |

### 3.1 单行文本框 <input type="text">

常用属性:

- name 名称，很重要，如果没有指定name，则该表单元素的数据是无法提交的
- value 初始值
- size 宽度
- maxlength 最大字符数，默认是没有限制
- readonly 只读，readonly="readonly"，可简写readonly，即只写属性名
- disabled 禁用，disabled="disabled"，可简写disabled，完全被禁用

readonly和disabled区别: readonly的数据会提交，而disabled的数据不会提交

表单元素被提交的两个条件: 1.有name属性 2.非disabled

### 3.2 单选按钮

常用属性:

```
性别: <input type="radio" name="sex" value="male" checked>  
<input type="radio" name="sex" value="female">
```

- name 名称，多个radio的name属性值必须相同，才能实现互斥 (单选)
- value 值
- checked 是否选中，两种状态: 选中、未选中 checked="checked"，简写 checked

### 3.3 复选框

常用属性与radio类似

### 3.4 文件选择器

常用属性：

- **name** 名称
- **accept** 设置可选的文件类型，用来限制上传的文件类型

使用MIME格式字符串对资源类型进行限制

常见的MIME类型：

- 纯文本：text/plain text/html text/xml
- 图像：image/jpeg image/png image/gif

#### 4. 特殊表单元素 不通过input标签实现

| 表单元素            | 含义               | 说明                     |
|-----------------|------------------|------------------------|
| <u>select</u>   | <u>下拉列表</u>      |                        |
| <u>option</u>   | <u>列表选项</u>      |                        |
| <u>optgroup</u> | <u>选项组</u>       | 用来对option进行分组          |
| <u>textarea</u> | <u>文本域/多行文本框</u> | 用来创建多行文本框 <u>会保留格式</u> |

##### 4.1 下拉列表

select常用属性：

- **name** 名称
- **size** 行数，同时显示多个选项
- **multiple** 允许同时选择多个
- **disabled** 禁用

option常用属性：

- **value** 选项值 必须要
- **selected** 设置默认选中项

optgroup常用属性：

- **label** 分组的标题

##### 4.2 文本域

常用属性：

- **name** 名称
- **rows** 行数
- **cols** 宽度

#### 5. 其他表单元素

## 5.1 label标签

为表单元素提供标签，当选中标签中的文本内容时会自动将焦点切换到与之关系的表单元素

常用属性：

- **for** 必须将该属性值设置为与之关联的表单元素的id属性值相同

注：几乎所有HTML标签都具有id属性，且id值必须唯一

## 5.2 button标签

也表示按钮，与input按钮类似

语法：

```
<input type="submit" value="提交按钮">
<input type="reset" value="重置按钮">
<input type="image" src="../day02/images/submit.gif">
<input type="button" value="普通按钮">
<br>
<button type="submit">提交按钮</button>
<button type="reset">重置按钮</button>
<button type="button">普通按钮</button>
<button></button>
<br>
```

```
<button type="按钮类型">按钮文本或图像</button>
```

常用属性：

- type 按钮的类型，取值：submit（默认）、reset、button

## 5.3 fieldset和legend标签

fieldset标签，对表单元素进行分组

legend标签，对分组添加标题

## 七、内嵌框架

### 1. 简介

使用iframe可以在一个页面中引用另一个页面，实现复用，灵活

### 2. 基本用法

语法：

```
<iframe src=""></iframe>
```

常用属性：

- src 引用的页面
- width/height 宽度/高度，像素或百分比
- frameborder 是否显示边框，取值：1(yes)、0(no)

- scrolling 是否显示滚动条，取值：yes、no、auto 内容太多放不下时显示
- name 为框架定义名称

### 3. 在框架中打开链接

```
<iframe name="hello">  
您的浏览器不支持iframe  
</iframe>
```

```
<a href="01.特殊表单元素.html" target="hello">01.html  
</a>  
<a href="02.其他表单元素.html" target="hello">02.html  
</a>  
<br>  
<iframe src="foot.html" width="600px" height="500px"  
" style="background: #ccc" name="hello"></iframe>
```

```
<a href="链接地址" target="hello">链接文本或图像</a>
```

表示在指定名称的框架中打开

## 八、HTML5简介

### 1. 发展

W3C于1992.12发布HTML4.0.1标准

W3C于2014.10发布HTML5标准

### 2. 特点

- 取消了过时的标签，如font、center等，它们仅用于展示外观
- 增加了一些更具有语义化的标签，如header、footer、aside等
- 增加了一些新功能标签，如audio、video、canvas等 在页面上绘图
- 增加了一些表单控件，如email、date、time、url、search等
- 可以直接在浏览中绘画（canvas），无需flash
- 增加了本地存储的支持

### 3. 兼容性

<http://caniuse.com>

提供了各浏览器版本对HTML5和CSS3规范的支持度

## 九、HTML5新增内容

### 1. 结构相关标签

用来进行页面结构布局，本身无任何特殊样式，需要使用CSS进行样式设置

- article 定义一个独立的内容，完整的文章
- section 定义文档的章节、段落
- header 文章的头部、页眉、标题
- footer 文章的底部、页脚、标注
- aside 定义侧边栏
- figure 图片区域



- `figcaption` 为图片区域定义标题
- `nav` 定义导航菜单

结构标签只是表明各部分的角色，并无实际外观样式，与普通的

相同

## 2. 语义相关标签

### 2.1 mark标签

标注，用来突出显示文本，默认添加黄色背景

### 2.2 time标签

定义日期和时间，便于搜索引擎智能查找

### 2.3 details和summary标签

默认显示summary中的内容，点击后显示details中的内容

注：并不是所有浏览器都兼容，Chrome、Opera支持，Firefox不支持

### 2.4 meter标签

计数仪，表示度量

常用属性：

- `max` 定义最大值，默认为1
- `min` 定义最小值，默认为0
- `value` 定义当前值
- `high` 定义限定为高的值
- `low` 定义限定为低的值
- `optimum` 定义最佳值

规则：

```
<meter max="100" min="1" value="70" high="60" low="20" optimum="50"></meter>
```

1. 如果`optimum`大于`high`，则表示值越大越好  
当`value`大于`high`时为绿色  
当`value`在`low`和`high`之间时为黄色  
当`value`小于`low`时为红色
2. 如果`optimum`小于`low`，则表示值越小越好  
当`value`小于`low`时为绿色  
当`value`在`low`和`high`之间时为黄色  
当`value`大于`high`时为红色

3. 当optimum介于low和high之间，则表示值在low和high之间最好  
当value在low和high之间时为绿色，否则为黄色

## 2.5 progress标签

进度条，表示运行中的进度

常用属性：

- value 定义当前值
- max 定义完成的值

## 3. 表单相关

### 3.1 新增表单元素

新增以下type类型：

- email 接收邮箱
- url 接收URL
- tel 接收电话号码，目前仅在移动设备有效
- search 搜索文本框
- number/range 接收数字/数字滑块，包含min、max、step属性
- date/month/week/time/datetime 日期时间选择器，兼容性不好
- color 颜色拾取

作用：

- 具有格式校验的功能
- 可以与移动设备的键盘相关联

### 3.2 新增表单属性

form标签的属性：

之前输入过

- autocomplete 是否启用表单的自动完成功能，取值：on（默认）、off
- novalidate 提交表单时不进行校验，默认会进行表单校验

### 3.3 新增表单元素属性

新增表单元素属性，input/select/textarea等

- placeholder 提示文字
- required 是否必填
- autocomplete 是否启用该表单元素的自动完成功能
- autofocus 设置初始焦点元素
- pattern 使用正则表达式（RegExp后面会讲解），进行数据校验
- list 使文本元素具有下拉列表的功能，需要配置datalist和option标签一起使用

```
<label for="age">年龄: </label>
<input type="text" name="age" id="age" pattern="\d{1,2}">
<br>
```

```
<input list="browsers">
<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

- form 可以将表单元素写在form标签外面，然后通过该属性来关联指定的表单

```
</form>

<input type="submit" value="外部的提交按钮" form="
myForm">
```

## 4. 多媒体标签

### 4.1 audio 标签

在页面中插入音频，不同的浏览器对音频格式的支持不一样

audio常用属性：

- src 音频文件的来源
- controls 是否显示控制面板，默认不显示
- autoplay 是否自动播放，默认不自动播放
- loop 是否循环播放
- muted 是否静音
- preload 是否预加载，取值：none不预加载、auto 预加载（默认）、metadata只预加载元数据

如果设置了autoplay属性，则该属性无意义

可以结合source标签使用，指定多个音频文件，浏览器会检测并使用第一个可用的音频文件

### 4.2 video 标签

在页面中插入视频，不同的浏览器对视频格式的支持不一样

用法和audio标签基本相同，增加属性：

- width/height 视频播放器的尺寸
- poster 在视频加载前显示的图片