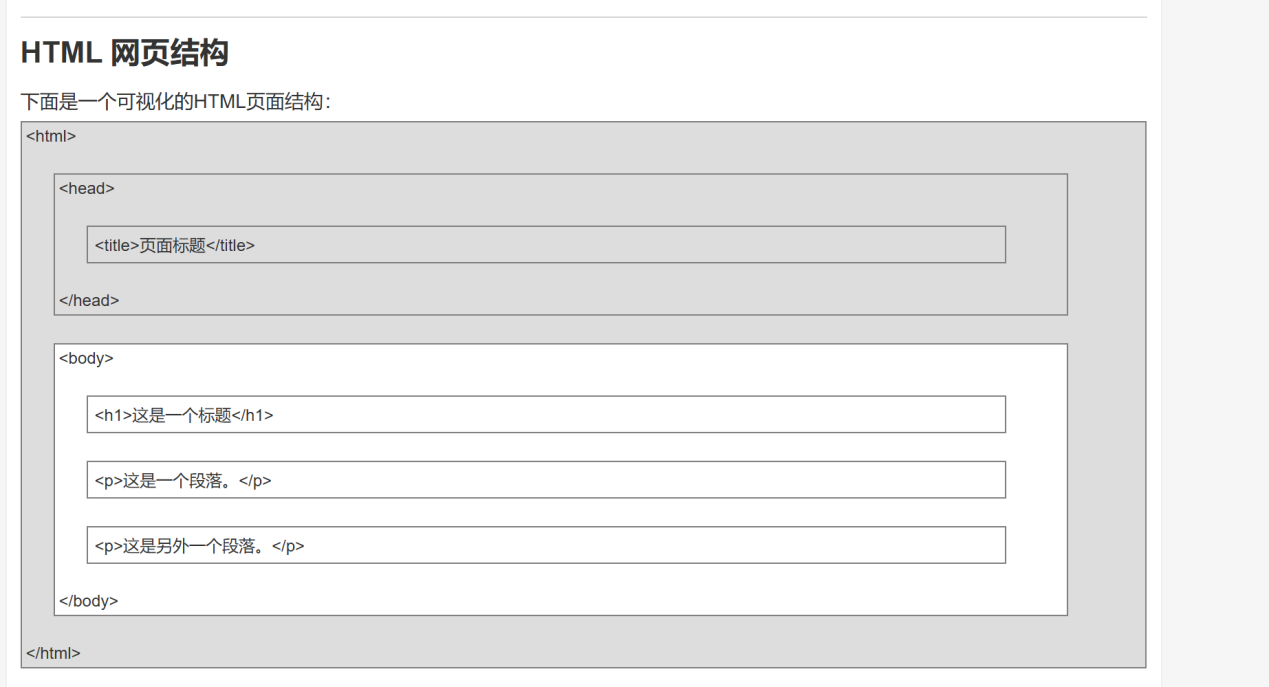
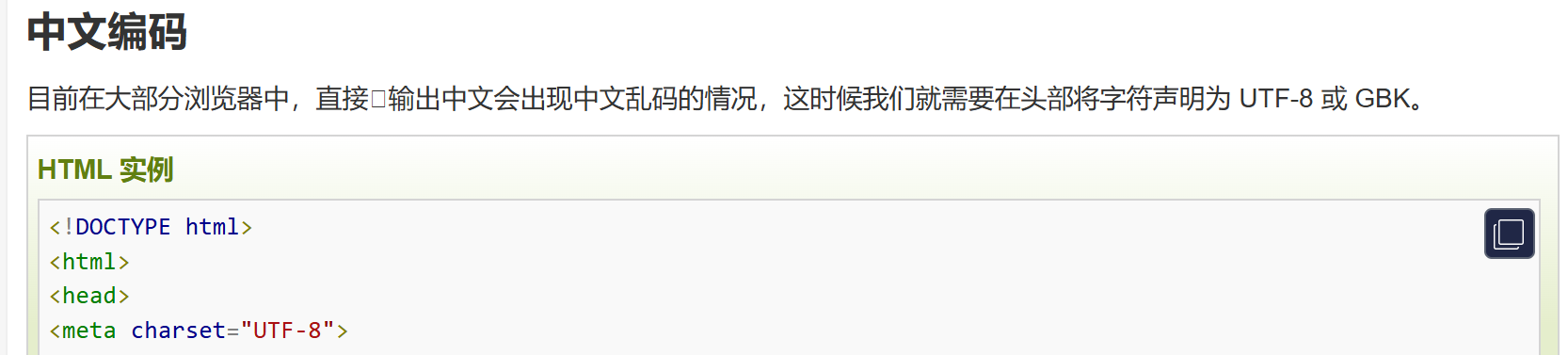
HTML简介(9.28)

* **<!DOCTYPE html>** 声明为 HTML5 文档
* **<html>** 元素是 HTML 页面的根元素
* **<head>** 元素包含了文档的元（meta）数据，如 **<meta charset="utf-8">** 定义网页编码格式为 **utf-8**。
* **<title>** 元素描述了文档的标题
* **<body>** 元素包含了可见的页面内容
* **<h1>** 元素定义一个大标题
* **<p>** 元素定义一个段落
* 
* <!DOCTYPE HTML>声明是不区分大小写的，
* 

1. 笔记

1. doctype 声明是不区分大小写的，用来告知 Web 浏览器页面使用了哪种 HTML 版本。

在HTML 4.01 中，<!DOCTYPE> 声明需引用 DTD （文档类型声明），因为 HTML 4.01 是基于 SGML（Standard Generalized Markup Language 标准通用标记语言）。

HTML 4.01 规定了三种不同的 <!DOCTYPE> 声明，分别是：Strict、Transitional 和 Frameset。

HTML5 不是基于 SGML，因此不要求引用 DTD。

2. 对于中文网页需要使用 <meta charset="utf-8"> 声明编码，否则会出现乱码。有些浏览器(如 360 浏览器)会设置 GBK 为默认编码，则你需要设置为 <meta charset="gbk">。

目前在大部分浏览器中，直接输出中文会出现中文乱码的情况，这时候需要在头部将字符声明为 UTF-8。

对于设置 **<meta charset="utf-8" />** 后出现网页乱码问题，其实归根到底就是：你**通过 meta 标签设置的编码**和**网页文件在保存时所使用的文档编码**不相同造成的！

至于有的人说什么 360 浏览器默认 **GBK** 会造成乱码，我只想说的是，

只要你在 html 文件里写了 **<!doctype hmtl>**和 **<meta charset="utf-8" />**，浏览器就绝对会按照 **utf-8** 编码解析网页，没有第二种可能！再次重点说明：**保存 html 文件时，文档编码和 meta 设置的编码，一定要相同，只要不相同，就一定会出现乱码！**

**之所以一定要写上 <!doctype html>，就是为了防止浏览器的怪异模式，强制浏览器按照标准模式渲染网页！**

**HTML编辑器(9.28)**

**·**



·编辑窗口右键***Show Preview，实时预览网页页面***

***HTML AI 编程助手（10.10）***

·按下 Ctrl+→ 键(mac系统为Command+→)接收单个词补全建议：

·按下 Tab 键接受所有补全建议：

·用户可通过点击左上角工具栏中的Fitten Code – 开始对话或者使用快捷键Ctrl+Alt+C(mac系统为Control+Option+C)打开对话窗口进行对话：

## 代码翻译

编辑代码功能可以实现不同语言之间的转换，如Python语法转换成C++语法等。选中需要进行编辑的代码段，右键选择 "Fitten Code – 编辑代码" 或点击左侧工具栏中的 "Fitten Code – 编辑代码" 或者使用快捷键 Ctrl+Alt+E (mac系统为Control+Option+E)，

## 6、生成注释

Fitten Code 能够根据您的代码自动生成相关注释，通过分析您的代码逻辑和结构，为您的代码提供清晰易懂的解释和文档，不仅提高代码的可读性，还方便其他开发人员理解和使用您的代码。先选中需要生成注释的代码段，然后右键选择 "Fitten Code – 生成注释"：

## 解释代码

Fitten Code 可以对一段代码进行解释，可以通过选中代码段然后右键选择 "Fitten Code – 解释代码" 进行解释，如下图所示：

## 生成测试

Fitten Code 拥有自动生成单元测试的功能，可以根据代码自动产生相应的测试用例，提高代码质量和可靠性。通过选中代码段后右键选择 "Fitten Code – 生成单元测试" 来实现，如下图所示

如果 VSCode 远程服务器 remote 无法连接外网时(设置)

HTML基础（10.11）

·HTML 链接是通过标签 <a> 来定义的

<a href="https://www.runoob.com">这是一个链接</a>

·HTML 图像是通过标签 <img> 来定义的.

<img src="/images/logo.png" width="258" height="39" />

# HTML 元素（10.11）

## 不要忘记结束标签

<html>  
<body>  
<p>这是第一个段落。</p>  
</body>  
</html>

·<br> 就是没有关闭标签的空元素（<br> 标签定义换行）

·即使 <br> 在所有浏览器中都是有效的，但使用 <br /> 其实是更长远的保障。

# HTML 属性（10.11）

·<a href="http://www.runoob.com">这是一个链接</a>

属性总是以名称/值对的形式出现，**比如：name="value"**

href为名称，“=”后为值对

属性值应该始终被包括在引号内



<hr> 标签在 HTML 页面中创建水平线。

hr 元素可用于分隔内容。

<p>这是一个段落。</p>

<hr>

<p>这是一个段落。</p>

## HTML 注释

开始括号之后（左边的括号）需要紧跟一个叹号 **!** (英文标点符号)，结束括号之前（右边的括号）不需要

<!-- 这是一个注释 -->

## 如何查看源代码

需要单击右键，然后选择"查看源文件"（IE）或"查看页面源代码"（Firefox）

希望在不产生一个新段落的情况下进行换行（新行），请使用 **<br>** 标签

<p>这个<br>段落<br>演示了分行的效果</p>

# HTML 文本格式化（10.11）

<b> 与<i> 定义粗体或斜体文本

**通常标签 <strong> 替换加粗标签 <b> 来使用, <em> 替换 <i>标签使用**

**<pre></pre>可以控制其中的文字的空格以及换行，其不受HTML的省略空格换行（唯一）控制**

**改变文字方向**

**<p><bdo dir="rtl">该段落文字从右到左显示。</bdo></p>**

显示为：该段落文字从右到左显示。

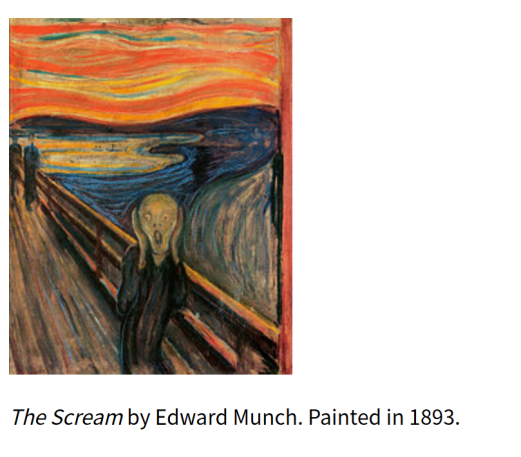
删除字效果和插入字效果：

**<p>My favorite color is <del>blue</del> <ins>red</ins>!</p>**

输出：

**使用 <cite> 标签来定义作品的标题：**

<p><cite>The Scream</cite> by Edward Munch. Painted in 1893.</p>

./.

**<address>定义地址**

**<address>**

**Written by <a href="mailto:webmaster@example.com">Jon Doe</a>.<br>**

**Visit us at:<br>**

**Example.com<br>**

**Box 564, Disneyland<br>**

**USA**

**</address>**

# <abbr> 标签定义缩写被标记的缩写词如下：

The<abbr title="World Health Organization">WHO</abbr> was founded in 1948.

<em>强调文本</em> **级别比<strong>低**

HTML链接

### **target：定义链接的打开方式。**

* \_blank: 在新窗口或新标签页中打开链接。
* \_self: 在当前窗口或标签页中打开链接（默认）。
* \_parent: 在父框架中打开链接。

\_top: 在整个窗口中打开链接，取消任何框架。

## 实例

<**a** href="https://www.example.com" target="\_blank" rel="noopener">新窗口打开 Example</**a**>

### **rel：定义链接与目标页面的关系。**

**nofollow**: 表示搜索引擎不应跟踪该链接，常用于外部链接。

**noopener** 和 **noreferrer**: 防止在新标签中打开链接时的安全问题，尤其是使用 **target="\_blank"** 时。

* noopener: 防止新的浏览上下文（页面）访问window.opener属性和open方法。
* noreferrer: 不发送referer header（即不告诉目标网站你从哪里来的）。
* noopener noreferrer: 同时使用noopener和noreferrer。
* 例子: <a href="https://www.example.com" rel="noopener noreferrer">安全链接</a>

### **download：提示浏览器下载链接目标而不是导航到该目标。**

例如：

<**a** href="file.pdf" download="example.pdf">下载文件</**a**>

### **title：定义链接的额外信息，当鼠标悬停在链接上时显示的工具提示。**

<**a** href="https://www.example.com" title="访问 Example 网站">访问 Example</**a**>

### **id：用于链接锚点，通常在同一页面中跳转到某个特定位置。 id使得页面中创建快捷方式到页面中的指定部分**

id 属性可用于创建一个 HTML 文档书签

*<!-- 链接到页面中的某个部分 -->*  
<**a** href="#section1">跳转到第1部分</**a**>  
<**div** id="section1">这是第1部分</**div**>

### **hreflang: 指定链接的目标URL的语言**

### URL是统一资源定位符，是互联网上标准资源的地址。基本URL包含模式（或称协议）、服务器名称（或IP地址）、路径和文件名

<**a** href="https://www.example.com/es" hreflang="es">访问西班牙语网站</**a**>

### **8、type: 指定链接资源的MIME类型**

<**a** href="style.css" type="text/css">样式表</**a**>

MIME类型由两部分组成：类型和子类型，两者通过一个斜线(/)分隔。例如，text/plain表示纯文本文件，image/jpeg表示JPEG图像文件

### **9、class: 用于指定元素的类名（CSS中定义）**

<**a** href="https://www.example.com" class="external-link">外部链接</**a**>

### **10、style: 直接在元素上定义CSS样式**

<**a** href="https://www.example.com" style="color: red;">红色链接</**a**>

CSS主要用于控制网页的外观和布局，可以定义颜色、字体、大小、位置等多种样式

图像链接

<a href="https://www.example.com">

<img src="example.jpg" alt="示例图片">

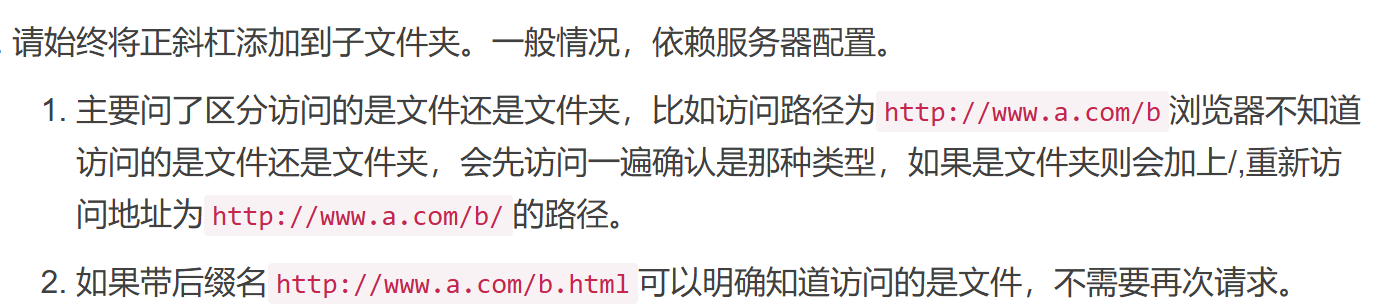
</a>

## 空链接

以下是常见的几种设置空链接的方法，以及它们之间的区别：

| **方法** | **作用** | **是否会跳转** | **场景适用性** |
| --- | --- | --- | --- |
| href="#" | 导航到页面顶部 | 是 | 占位符，捕获点击事件 |
| href="javascript:void(0)" | 阻止默认行为，不刷新页面 | 否 | 阻止跳转，配合 JS 使用 |
| href="" | 刷新当前页面 | 是 | 需要页面刷新时 |
| href="about:blank" | 打开空白页面 | 是 | 新窗口打开空白页面 |
| role="button" | 链接表现为按钮，无默认行为 | 否 | 配合 JS 实现按钮功能，无跳转 |

**正斜杠添加到子文件夹**



HTML邮件发送链接(10.13)

单词之间空格使用 %20 代替，以确保浏览器可以正常显示文本

<a href="mailto:someone@example.com?Subject=Hello%20again" target="\_top">

在进行邮件内容发送时，需要使用关键字：**mailto**

示例如下：

<a href="mailto:zhangrr601@163.com?subject=这是邮件的主题&body=这是邮件的内容" rel="nofollow">发送邮件</a>

|  |  |
| --- | --- |
| **参数** | **描述** |
| mailto:*name@email.com* | 邮件接收地址 |
| cc=*name@email.com* | 抄送地址 |
| bcc=*name@email.com* | 密件抄送地址 |
| subject=*subject text* | 邮件主题 |
| body=*body text* | 邮件内容 |
| ? | 第一个参数分隔符 |
| & | 其他参数分隔符 |

注：多个邮件地址用 **;** 隔开，空格用 **%20** 代替。

rel="noopener noreferrer"

意思是不会打开其他的网站，因为恶意病毒可能会修改你的浏览器空白页地址。

HTML头部(10.13)

**<base>定义所有链接的URL:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>菜鸟教程(runoob.com)</title>

<base href="https://www.runoob.com/images/" target="\_blank">

</head>



[<meta> - 提供了HTML文档的meta标记](https://www.runoob.com/try/try.php?filename=tryhtml_meta" \t "https://www.runoob.com/html/_blank)  
使用 <meta> 元素来描述HTML文档的描述，关键词，作者，字符集等



META 元素通常用于指定网页的描述，关键词，文件的最后修改时间，作者，和其他元数据。

元数据可以使用于浏览器（如何显示内容或重新加载页面），搜索引擎（关键词），或其他Web服务

## <meta> 标签- 使用实例

为搜索引擎定义关键词:

<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">

为网页定义描述内容:

<meta name="description" content="免费 Web & 编程 教程">

定义网页作者:

<meta name="author" content="Runoob">

每30秒钟刷新当前页面:

<meta http-equiv="refresh" content="30">

HTML CSS

内联样式

我们可以使用font-family（字体），color（颜色），和font-size（字体大小）属性来定义字体的样式:

## 实例

<h1 style="font-family:verdana;">一个标题</h1>

<p style="font-family:arial;color:red;font-size:20px;">一个段落。</p>

使用 text-align（文字对齐）属性指定文本的水平与垂直对齐方式：

## 实例

<h1 style="text-align:center;">居中对齐的标题</h1> <p>这是一个段落。</p>

内部样式

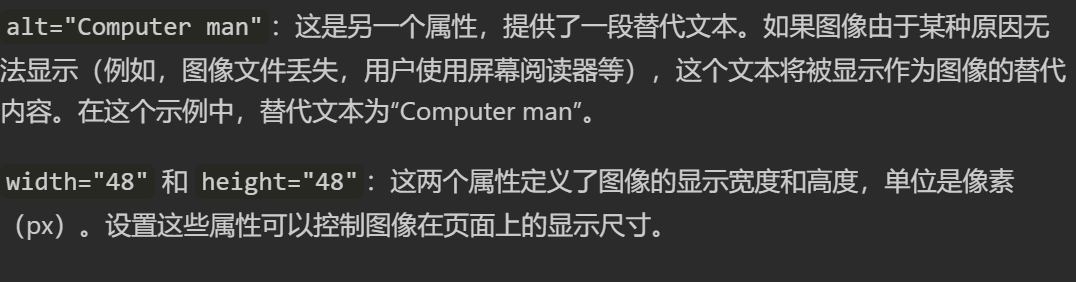
<head><style type="text/css">

body {background-color:yellow;}

p {color:blue;}</style></head>

HTML 图片

<img src="hackanm.gif" alt="Computer man" width="48" height="48">

  
<img> 是空标签，意思是说，它只包含属性，并且没有闭合标签。

要在页面上显示图像，你需要使用源属性（src）。src 指 "source"。源属性的值是图像的 URL 地址

这里我们使用了 CSS float 属性（图片浮于文字左方或右方）

<img src="smiley.gif" alt="Smiley face" style="float:right" width="32" height="32">

设置图像链接（点击图片跳转网页链接）

<p>创建图片链接:

<a href="http://www.runoob.com/html/html-tutorial.html">

<img border="10" src="smiley.gif" alt="HTML 教程" width="32" height="32"></a></p>

创建图像映射

<p>点击太阳或其他行星，注意变化：</p>

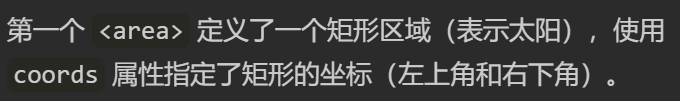
<img src="planets.gif" width="145" height="126" alt="Planets" usemap="#planetmap">

<map name="planetmap">

<area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">

</map>





~shape是点击区域的形状

获取图片坐标的方法：

把图像转换成图像映射，就能自动获取到图像中的点坐标了，可以利用以下代码实现。

<p>

<a href="http://www.baidu.com/" target="\_blank">

<img src="http://ww2.sinaimg.cn/bmiddle/69c7be75jw1ec4tmykr6nj20c807sjrl.jpg" ismap alt="柠檬下午茶">

</a>

</p>

之所以图片代码前引用一段url是为了鼠标停留在图片上时，可以在浏览器窗口的状态栏中看到返回值出现的点坐标，如图所示

矩形：(左上角顶点坐标为(x1,y1)，右下角顶点坐标为(x2,y2))

圆形：(圆心坐标为(X1,y1)，半径为r)

多边形：(各顶点坐标依次为(x1,y1)、(x2,y2)、(x3,y3) ......)

HTML表格

<table> 元素表示整个表格，它包含两个主要部分：<thead> 和 <tbody>

* **tr**：tr 是 table row 的缩写，表示表格的一行。
* **td**：td 是 table data 的缩写，表示表格的数据单元格。
* **th**：th 是 table header的缩写，表示表格的表头单元格。
* 显示边框<table border="1">
* 带有标题的表格

<table border="1">

<caption>标题</caption>

单元格合并两列：<th colspan="2">Telephone</th>

单元格合并两行 ：<th rowspan="2">Telephone:</th>

单元格边距：<table border="1" cellpadding="10">

标签常用属性

bgcolor="#fff" 表格整体的背景色

bordercolor="#fff" 表格边框的颜色

rowspan="3" 合并垂直水平方向的单元格

colspan="3" 合并水平方向单元格

valign="top" 单元格文字的垂直对齐方式

HTML列表（body部分）

有序列表<sart属性表示从50开始>&<type表示种类>：

<ol start="50"> &&&&&<ol type="A">

<li>Coffee</li>

<li>Tea</li>

<li>Milk</li>

</ol>

无序列表：

<ul>

<li>Coffee</li>

<li>Tea</li>

<li>Milk</li>

</ul>

自定义列表：

自定义列表以 <dl> 标签开始。每个自定义列表项以 <dt> 开始。每个自定义列表项的定义以 <dd> 开始

<dl>  
<dt>Coffee</dt>  
<dd>- black hot drink</dd>  
<dt>Milk</dt>  
<dd>- white cold drink</dd>  
</dl>

HTML块区

大多数 HTML 元素被定义为**块级元素**或**内联元素**

* <div> 是块级元素，它独占一行，可以设置宽度、高度以及边距等样式属性。它适合用于创建页面的大块结构，例如页面的主体区域、容器、布局等。
* <span> 是行内元素，它不会独占一行，宽度默认由其内容决定。它适合用于对文本或其他行内元素进行样式化、标记或包裹。

网页布局：

[使用 <div> 元素的网页布局](https://www.runoob.com/try/try.php?filename=tryhtml_layout_divs" \t "https://www.runoob.com/html/_blank)（多使用这种）

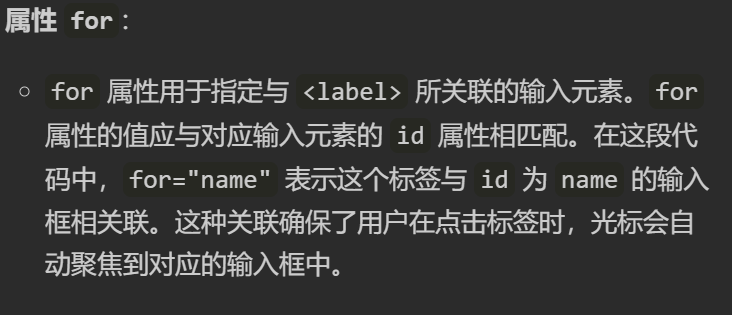
[使用 <table> 元素的网页布局](https://www.runoob.com/try/try.php?filename=tryhtml_layout_tables" \t "https://www.runoob.com/html/_blank)

<form action="/" method="post">

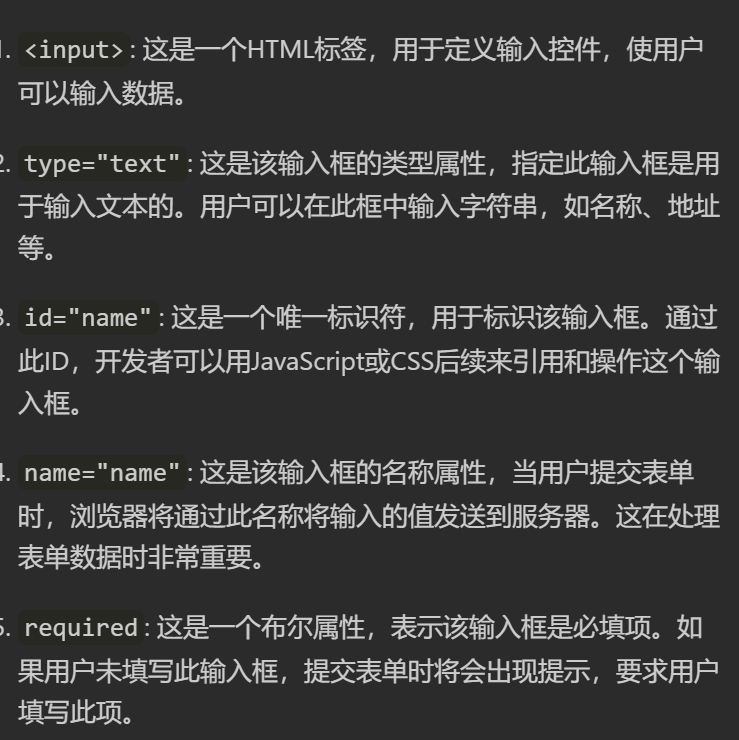
<!-- 文本输入框 -->

<label for="name">用户名:</label>

<input type="text" id="name" name="name" required>



<input type="password" id="password" name="password" required>



<!-- 单选按钮 -->

<label>性别:</label>

<input type="radio" id="male" name="gender" value="male" checked>

<label for="male">男</label>

<input type="radio" id="female" name="gender" value="female">

<label for="female">女</label>

<!-- 复选框 -->

    <input type="checkbox" id="subscribe" name="subscribe" checked>

    <label for="subscribe">订阅推送信息</label>

复选框：checkbox复选，独特id标识符便于css引用,name表示进入服务器数据类型，checked表示默选

<!-- 下拉列表 -->

    <label for="country">国家:</label>

    <select id="country" name="country">

        <option value="cn">CN</option>

        <option value="usa">USA</option>

        <option value="uk">UK</option>

    </select>

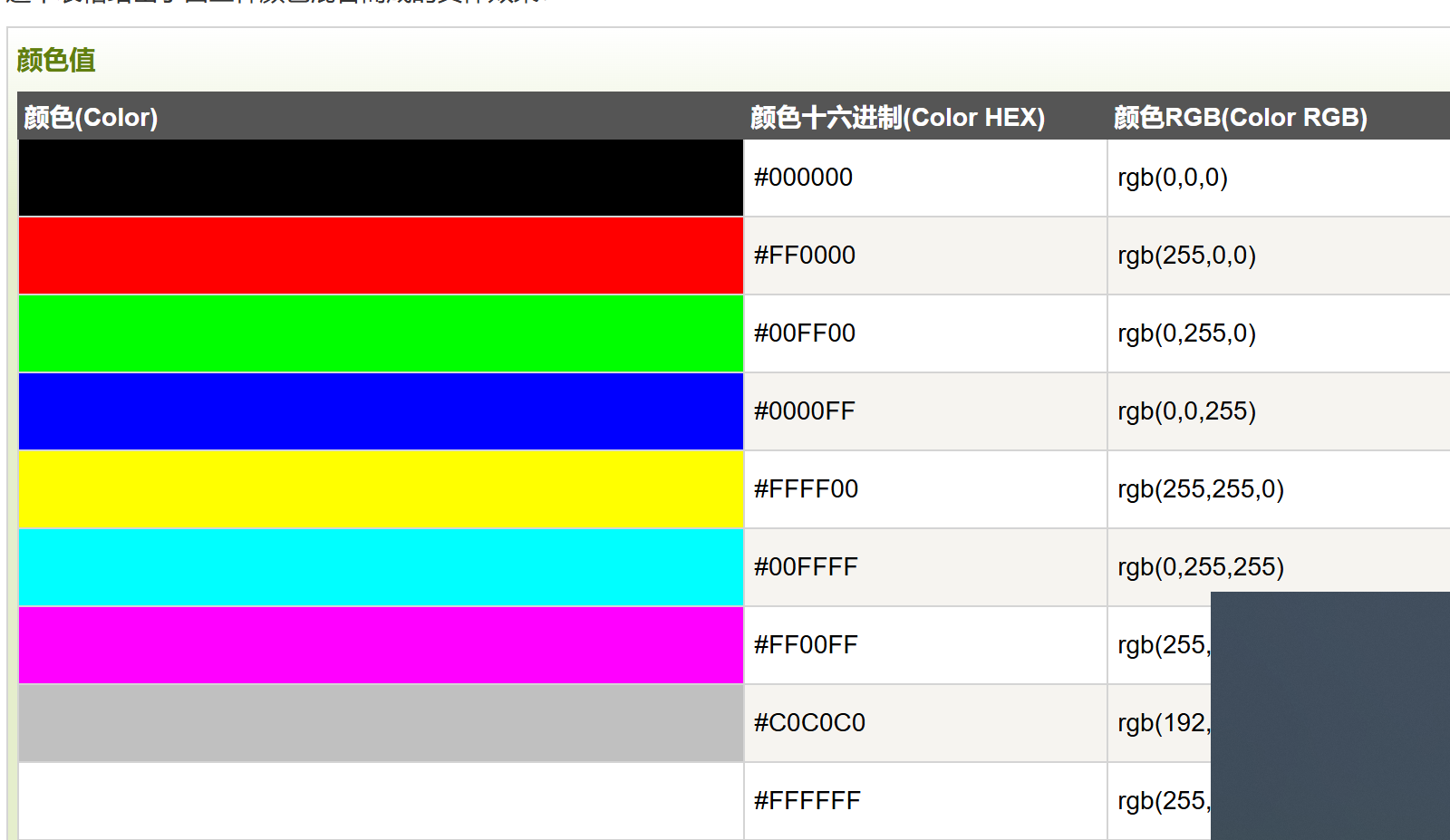
下拉表格：select元素表示收集一个国家数据，option表示下拉表格其中一个选项

 <!-- 提交按钮 -->

    <input type="submit" value="提交">

type="submit"表示输入元素类型为提交按钮，表单中所有数据会被提交到指定URL

HTML颜色



<p style="background-color:#FFF00">

通过十六进制设置背景颜色

</p>

<p style="background-color:rgb(250,250,250)">

通过 rgb 值设置背景颜色

</p>

<p style="background-color:yellow">

通过颜色名设置背景颜色

HMTL脚本

<script> 标签用于定义客户端脚本，比如 JavaScript

<script>

document.write("Hello World!")

</script>

<noscript>抱歉，你的浏览器不支持 JavaScript!</noscript>

<p>不支持 JavaScript 的浏览器会使用 &lt;noscript&gt; 元素中定义的内容（文本）来替代。</p>

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

HTML字符实体

小于号，我们必须这样写：**&lt;** 或**&#60;** 或 **&#060**

如果您在文本中写 10 个空格，在显示该页面之前，浏览器会删除它们中的 9 个。如需在页面中增加空格的数量，您需要使用 &nbsp

HTML URL

|  |  |  |
| --- | --- | --- |
| http | 超文本传输协议 | 以 http:// 开头的普通网页。不加密。 |
| https | 安全超文本传输协议 | 安全网页，加密所有信息交换。 |
| ftp | 文件传输协议 | 用于将文件下载或上传至网站。 |
| file |  | 您计算机上的文件。 |