

零基础HTML入门教程

乘风者系列图书

CHENWENYANG
陈文阳

乘风者计划专家博主
高校软件技术专业教师



HTML零基础入门级教学

理论与实践相结合

HTML常用标签汇总与详解

从理论到实践，教你从零开始入门HTML

阿里云开发者社区
ALIBABA CLOUD DEVELOPER COMMUNITY

乘风者计划
THE WIND RIDERS PROGRAM



关注【程序员大阳】
学习更多零基础教程



阿里云开发者“藏经阁”
海量电子手册免费下载

目录

零基础 HTML 入门教程(1)--教程说明	5
零基础 HTML 入门教程(2)--了解编程语言	8
零基础 HTML 入门教程(3)--网站与网页	11
零基础 HTML 入门教程(4)--开发第一个网页	12
零基础 HTML 入门教程(5)--第一段代码	16
零基础 HTML 入门教程(6)--学习 body 标签	20
零基础 HTML 入门教程(7)--支持中文的技巧	23
零基础 HTML 入门教程(8)--设置网页标题	26
零基础 HTML 入门教程(9)--添加网页段落	29
零基础 HTML 入门教程(10)--块级元素与行级元素	31
零基础 HTML 入门教程(11)--初识 VSCode	34
零基础 HTML 入门教程(12)--VSCode 基础用法	38
零基础 HTML 入门教程(13)--网页类型声明	46
零基础 HTML 入门教程(14)--水平线	49
零基础 HTML 入门教程(15)--注释	51
零基础 HTML 入门教程(16)--理解标签的属性	54
零基础 HTML 入门教程(17)--图像	56

零基础 HTML 入门教程(18)--超级链接.....	59
零基础 HTML 入门教程(19)--列表.....	61
零基础 HTML 入门教程(20)--换行.....	63
零基础 HTML 入门教程(21)--表格.....	66
零基础 HTML 入门教程(22)--表单.....	70
零基础 HTML 入门教程(23)--综合实例.....	76

零基础 HTML 入门教程(1)--教程说明

1. 教程背景

我写过很多入门教程，有一些我自认为还不错，也获得了一些网友的好评。但是没有写过零基础的，从入门到深入的系列教程。

我一直想写，但是一直感觉没有这个必要，因为市面上的教学教程太多了，我没有必要再给初学者增加选择的烦恼了。

直到今年，我成为一名高等职业院校软件技术专业教师，在教学实践的过程中，我逐渐对软件开发教程有了更加真实的体会。我感觉不论是我的学生，还是社会上希望学习软件开发技术的新手，都需要这样的一种教程：

- 零基础入门，只有零基础就能上手学习，才能面向各种层次的学生和人群，这是作为一门教程的基本修养吧。
- 通俗易懂，这个很重要，如果写的不够通俗，则容易导致新手在学习过程中经历太多挫折感，这是我所不希望的，我是希望能够帮助到新手。
- 具备一定的趣味性，这一点也很重要，如果是无趣的教程，则容易让人感到乏味，不易学习到心里去。当然，让教程变得有趣，是很困难的，我目前有一定的心得，但是我也不知道是否能够做到，且行且观察吧。
- 短教程，这个是参考短视频而来。现在的学生，都是容易疲惫的，他们喜欢在紧凑的节奏中获得满足。看短视频是如此，做教程也要做的短而有内容，让读者能在极快的时间内体验获得知识的快感。这点，可以说是我写这门教程的特色吧。

当然，促使我写这门教程还有一个直接因素，我在学校带了一些对软件开发感兴趣的学生，却发现他们的空闲时间都是分散开的。这就意味着如果我要亲自教他们，需要无数次的重复。

所以我索性将我的教学理念、教学思路、知识点融为一套教程，让教程代我去教我的学生。

2. 教程形式

首先是要有文字教程的，我喜欢通过文字整理思路，并表达我的思想。

当然文字还是不够直观，所以也在考虑后续推出系列视频教程，视频教程的话录制比较耗费心思。

目前正在持续录制视频教程了，当然源代码也会如数奉上。

3. 面向人群

本教程虽然是为高等职业院校学生编写的，但是其实可以面向具备一定数学和英语基础的、爱好计算机编程的所有人群。

为何要有一定数学基础，因为编程很多时候是需要数学运算的，但是这个数学基础不用太深，初中数学足矣。我个人认为只要能理解函数的概念，就具备了成为程序员的数学基础。如果函数都不懂，我还是建议先去学习下数学。

为何要有一定的英语基础，按照国际惯例我们写代码时，很多东西是用英文命名的。所以具备一定的英语基础，会更加容易理解代码。据我个人的认知，初中英语水平也是完全足够的。此处也需要注意一点，如果完全不懂英语，也是可以学习本教程的，只是要更加困难，付出更多努力而已。

最后我希望强调一点，就是爱好计算机编程。虽然我致力于编写有一定趣味性的教程，但是实质上编程是一门技能，是一种知识体系。在学习的过程中难免有困难的地方，而且困难的地方还不少，这就需要爱好来支撑大家渡过困难的日子，直到迎来阳光灿烂。

4. 知识体系

本教程是零基础系列教程的一部分，全套教程具体学习内容还包含前端的 HTML、CSS、JavaScript、

jQuery、Bootstrap、Vue，以及后端的 Java、Servlet、JDBC、MySQL 数据库、SSM、Spring Boot、Spring Cloud，本教程讲解其中的 HTML 技术。

由于是入门级教程，所以重在引导学会，而不是全面展开。

不管学习何种后端语言，前端这套技术基本是确定的，所以前端这部分不管是什么路线，都可以进行学习。

5. 小结

小结下，这门教程要简单一些，每章的内容少一些，有趣一些。

零基础 HTML 入门教程(2)--了解编程语言

1. 什么是语言

在世界上生存，总是要沟通的。语言就是用来在不同个体之间沟通的交际工具。

人与人之间沟通使用的语言，一般称为自然语言，例如汉语、英语、法语、日语、德语。

程序员要想与计算机进行沟通，让计算机执行自己想要的任务，就得有一种语言跟计算机进行沟通，这就是编程语言了。

2. 自然语言的特点

自然语言是人类自然社会发展过程中逐渐形成的，所以它受历史影响、受地域影响，它是非常复杂的。

一般来说，学习一门自然语言，例如我们中国人普遍学习的英语，需要一个很长的过程。

但是自然语言是有规则的，英语有很多语法，汉语也有。

既然语言是用来在不同个体之间沟通的，所以大家都得遵从语言的规则，才能互相理解。不然你说你的，我说我的，不按规矩来，就没法互相沟通。

自然语言，庞大而复杂。

3. 编程语言的特点

编程语言是程序员和计算机之间沟通的语言。

计算机可不像人类那么聪明，计算机本质上是一个不知疲倦的机器，它只能按照程序员的指令完成既定任务。

这就决定了，编程语言要比较简洁、比较明确，毕竟计算机是无法理解人类语言中的语气、腔调等内在含义的。

编程语言是规则更加明确的语言，它有着非常固化的模式，一般而言学习一门编程语言，只需要几个月的时间，就能初步达成与计算机沟通的目的。

而且不同的编程语言，具备高度的相似性，掌握一门语言后，再学习其他编程语言是一件挺容易的事情。

4. 编程语言的江湖

大家想必听说过 C 语言的名头，我们是用的 Windows 操作系统就是它编写的。

也想必听说过 Java 语言的名头，国内知名电商阿里巴巴的主力语言就是 Java。

另外可能 Python 也比较出名了，甚至诞生了“人生苦短、我用 Python”的名言。

其实各种语言并没有很明显的优劣之分，龙生九子各有所长。每种语言都有特长，都有特别适合它发挥的领域。

作为初学者来说，我是特别推荐学习 Java 语言的，第一是因为它简单易懂；第二我认为它是招聘岗位最多的语言，换言之学习这门语言最容易找工作；第三，这门语言博大精深，可以为学习者提供一个从入门到高手的足够长远的职业生涯发展路线。

5. 规则不可避免

不管是自然语言，还是编程语言，都需要遵循规则。

按照既定的规则，则双方都能互相理解，程序员和计算机才能达成共识。

学习语言，就是理解规则，利用规则的过程。

人在社会中存在，又何尝不是如此。

6. 小结

纸上得来终觉浅，绝知此事要躬行。

编程语言的规则不算难，但是要理解掌握，还是需要在计算机上不断练习。

读书百遍其义自见，这句话应用与编程学习领域实在是再合适不过了，很多技术可能一时半会理解不了，但是写得多了，潜移默化就会用了。都已经会用了，那就是已经理解了。

加油吧。

零基础 HTML 入门教程(3)--网站与网页

1. 什么是网站

本门课程是要带大家学习使用 Java 开发 Web 应用，所谓的 Web 应用，通俗的讲就是网站。

什么是网站呢，百度是一个网站、淘宝是一个网站、网易是一个网站、京东也是一个网站。

一个网站就是互联网上的一个站点，大家可以通过浏览器访问它们，获取一些特定的信息。

2. 什么是网页

大多数网站都是有很多网页组成的，例如淘宝有很多商品，点击每个商品都会打开一个新的网页。

例如网易新闻，点击一条新闻就打开了展示该新闻信息的具体网页。

网页就是网站的一个页面，一般来说网站都是由许多网页构成的。

3. 网页是怎么开发的

其实网页也是一个文件，你还记得 word 文件么，我们可以使用 Word 编写一个 word 文字文档。同样我们可以用 Excel 软件编写一个 Excel 表格文档。再同样我们可以用 PowerPoint 编写一个 PPT 演示文稿。

网页其实就是用网页开发工具，开发一个网页文档，当我们打开一个浏览器的网页时，实际上也是打开了一个对应的网页文档。从本质上讲，都是用一个专门的软件打开一个对应的文档。

那么网页是用什么软件开发的呢，其实只要是文本编辑器都能开发网页，例如大家最熟悉的记事本。

4. 小结

网页就是一种特定的文档，可以用文本编辑器去编写。

零基础 HTML 入门教程(4)--开发第一个网页

1. 文件类型

在学习编写网页之前，大家一定要掌握一个非常基础且不可或缺的概念，就是文件类型。

我们知道文件分为 txt 文本文件、word 文件、excel 表格文件、ppt 演示文档、mp3 音乐文件等等不同的类型。

那么电脑是怎么区分这些类型的呢，电脑只有区分开文件类型，才能使用正确的程序打开这些文件。

实际上，我们的电脑是通过文件的拓展名，来区分文件的，也是通过文件的拓展名，来决定打开文件时使用什么程序。

例如：记事本编写的文件，拓展名是.txt；word 编写的文档，拓展名是.doc 或者.docx，doc 是早期使用的拓展名，后来又推出了 docx；excel 编写的表格闻到那个，拓展名是.xls 或.xlsx；mp3 音乐文件的拓展名，那就是.mp3 了。

2. 隐藏拓展名

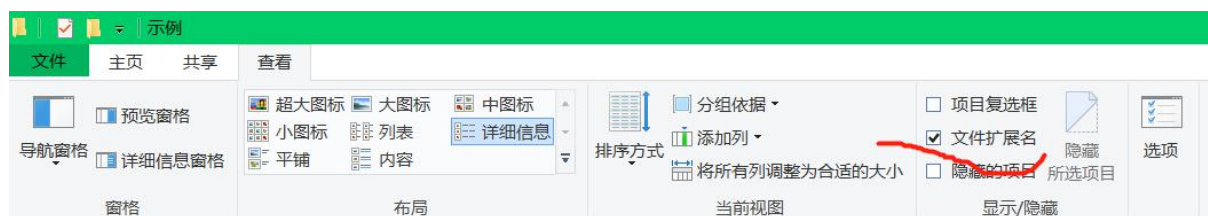
电脑上的文件，基本都是有拓展名的，如下图，我们可以通过拓展名轻易的区分开这是一个什么文件。

名称	类型
 新建 Microsoft Excel 工作表.xlsx	Microsoft Excel 工作表
 新建 Microsoft Word 文档.docx	Microsoft Word 文档
 新建文本文档.txt	文本文档

但是有些同学的电脑上，是不显示拓展名的，如下图：



一定要注意的，不显示文件的拓展名并不代表文件没有拓展名，只是隐藏了而已，我们可通过勾选文件拓展名复选框来显示拓展名。



作为一名程序员来说，我们应该始终展示文件的拓展名，便于我们随时分析文件类型。

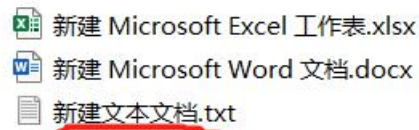
3. 网页的拓展名

OK，分析到这里，我们只需要知道网页的拓展名，就可以创建一个网页文件了。

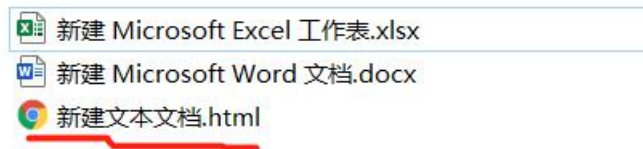
网页的拓展名一般是.html，所以我们可以创建拓展名为.html的文件，因为该文件的拓展名是.html，所以电脑就会将它作为网页文件来处理。

一般来说，我们使用右键新建文件时，并没有选项让我们直接创建一个网页文件，我们可以选择创建记事本文件，然后将其拓展名.txt 改为.html。

新建文本文件，如下图：



修改其拓展名，如下图：

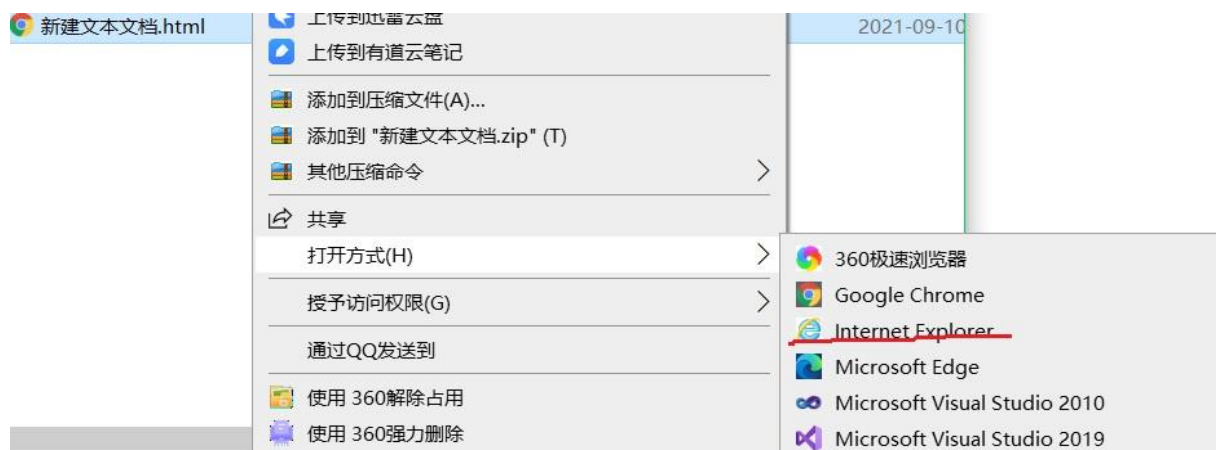


这样我们就创建了一个网页了。

4. 网页的打开方式

一般来说，我们会使用浏览器打开网页，但是如果我们想编辑网页的内容，我们可以通过记事本或其他编辑工具打开网页。

可以右键点击网页文件，选择具体打开方式，如下图，我们可以选择 IE 浏览器打开这个网页，来查看它的效果。



5. 小结

我们本章开发了一个空白网页，使用浏览器打开后是空白的，但它已经是一个网页文件了。

接下来，我们就学习如何编写网页文件的内容。

零基础 HTML 入门教程(5)--第一段代码

1. 什么是 HTML

在上一章的学习中，我们已经开发了一个空白的网页。但是空白的网页啥作用呢？

所以我们需要设计一个好看的，有内容的网页，而 HTML 语言，就是用来编写网页内容的语言。

我们使用 HTML 语言规范编写的网页，浏览器就能按照相应的规则展示我们设计的内容。

2. 为啥叫 HTML

HTML，英文全称 HyperText Markup Language，翻译为中文：超文本标记语言。

OK，语言很好理解，编程语言就是程序员和计算机之间的沟通工具，HTML 是程序员设计网页内容的语言。

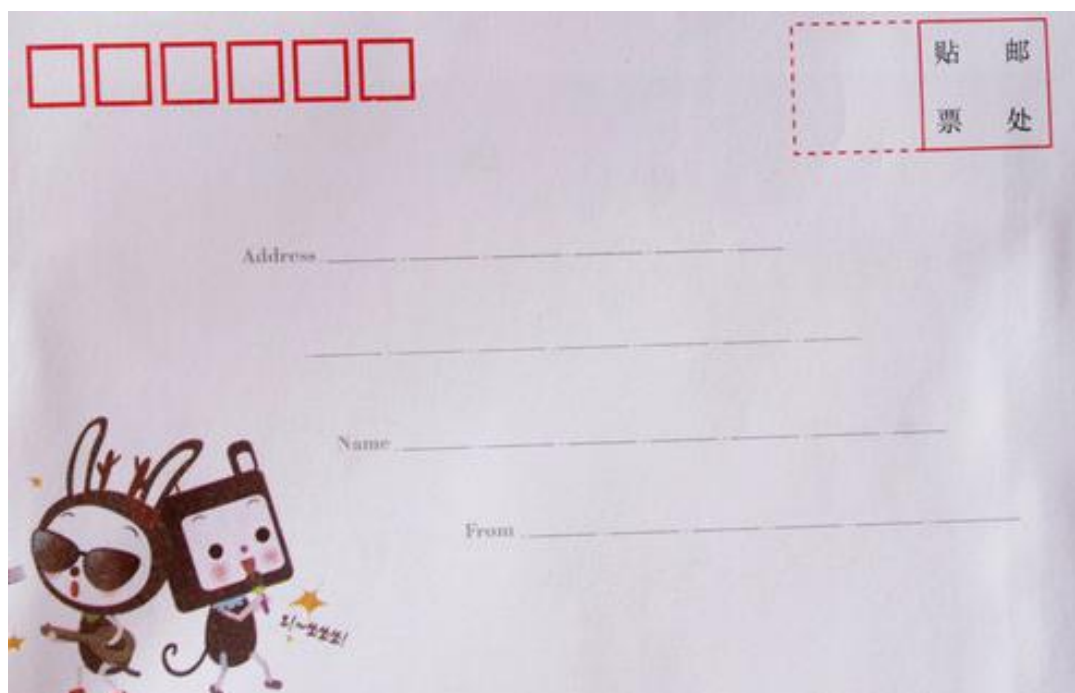
文本也很好理解，就是文字的意思。

那么超文本是啥意思呢？这里的超其实是超越的意思，也就是说这门语言超越了文本。因为 HTML 网页不光能展示文本，还能展示图片、音乐、视频的内容，所以是超越了文本的。

最后是重点：标记，这体现了这门语言的特点。HTML 是通过标记来描述网页的各个部分的。

3. 什么是标记

举个例子先，我们看一下信封的格式。



我们分析下，左上角是邮编，右上角是邮票，中间依次是地址、姓名、寄信人。

这其实就是一种标记，通过不同的位置标记出来这个区域代表的内容。

所以寄信人发出邮件后，邮政公司就能根据不同位置标记的内容，获得对应的邮编、地址等信息。

4. HTML 的标记

HTML 的标记不是通过位置来实现的，它是通过标签。

例如 `<title>淘宝网</title>` 就是一个网页标题的标签，都是以 `<xxx>` 开头，且以 `</xxx>` 结束，这就是标记，标记名称 `xxx` 代表标记的用途，`<xxx>` 和 `</xxx>` 中间的部分表示标记的内容。

所以对于 `<title>淘宝网</title>` 来说，这个标签的用途就是 `title`，也就是设置网页的标题；这个标签的内容是 `淘宝网`，也就是网页的标题设置为淘宝网。

5. 编写第一段 HTML 网页代码

我们学习最简单的 3 个标签：`<html>`、`<head>`、`<title>`，第一个标签表示 HTML 网页，第一个表示网页的头部，第三个表示网页的标题。

打开记事本，新建一个 `hello.html`，然后内容如下。这样的话，标签中间的部分就是网页的内容了。

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

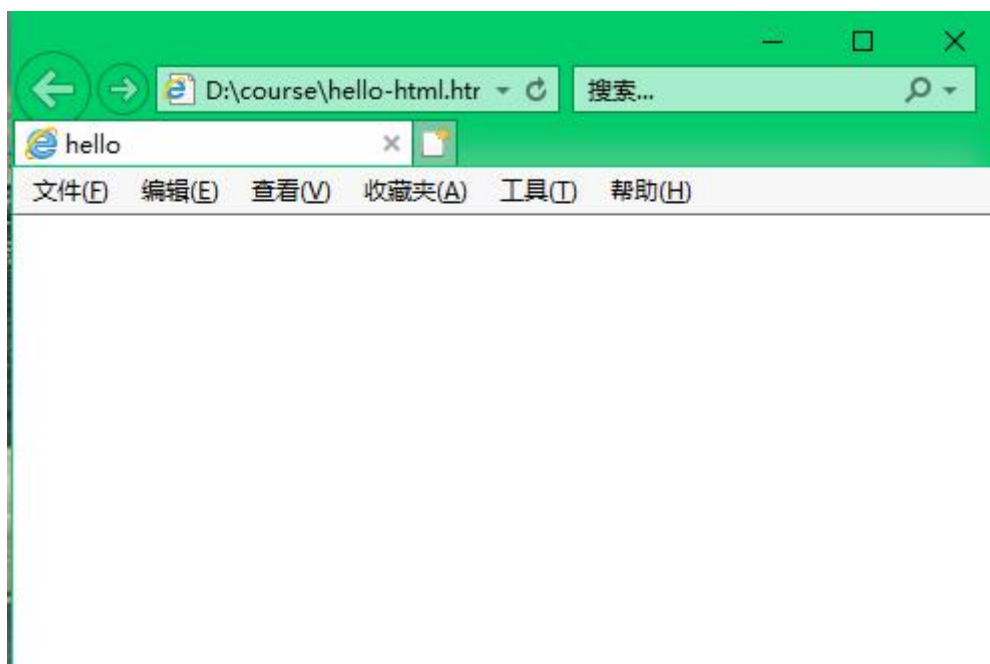
然后在中间添加 `<head>` 标签，因为网页的头部属于网页，所以放到 `<html>` 标签的中间。

```
<html>  
<head></head>  
</html>
```

最后，我们在网页的头部，设置网页的标题，标题的内容为：`hello`。

```
<html>  
<head>  
  <title>hello</title>  
</head>  
</html>
```

此时我们使用浏览器打开这个网页，效果如下，可见我们已经成功设置了标题。



6. 小结

其实 HTML 语言的规则及其简单，标签几乎都是以 `<xxx>` 开头，对应的以 `</xxx>` 结束，而开始和结束中间的部分就是标签对应的内容。

常用的标签也就十几个，所以学习 HTML 就是熟悉这些常用的标签的过程。

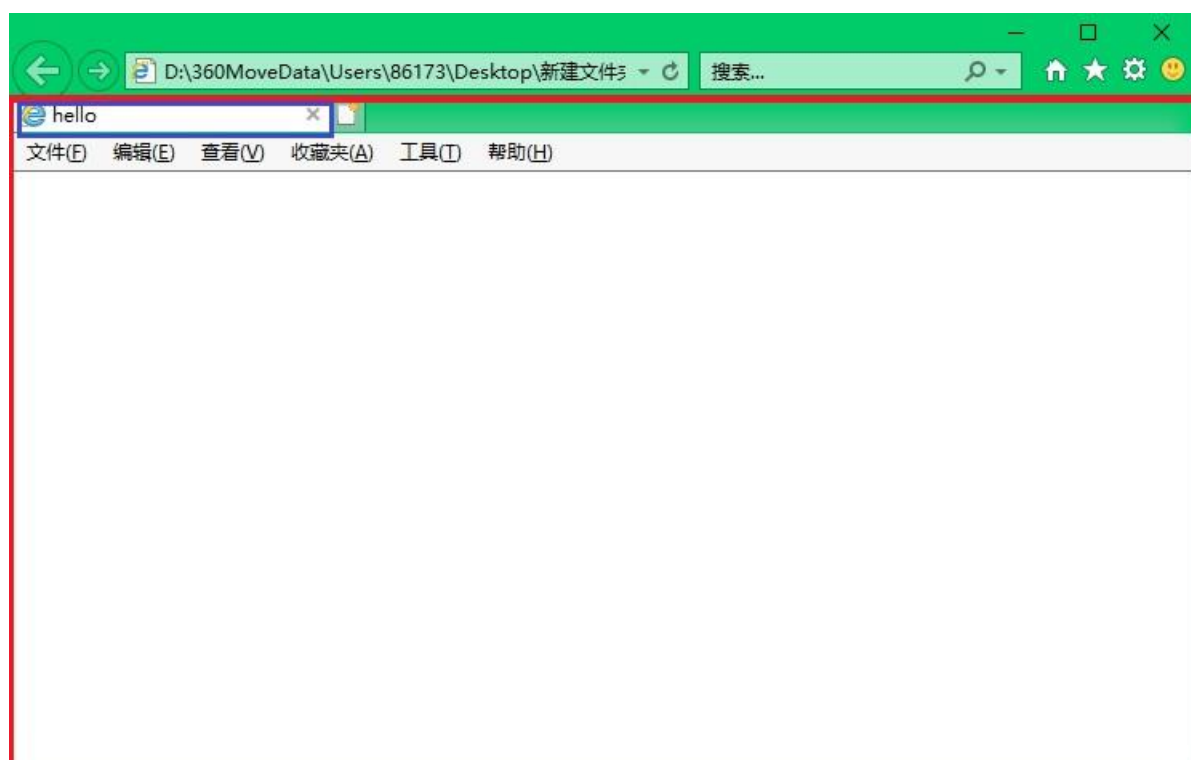
零基础 HTML 入门教程(6)--学习 body 标签

2. 关于标签

网页是 HTML 代码编写的，而 HTML 代码是由标签组成的。

标签的样式形如<xxx></xxx>，注意<xxx>是标签的开头部分，</xxx>是对应的结尾部分，开头和结尾中间夹着的就是标签的内容了。

所以，上节课我们讲的三个标签<html>、<head>、<title>分别对应的是整个网页，网页的头部，网页头部的标题。如下图，红线部分就是整个网页，蓝线部分就是网页的头部，而 hello 就是网页头部的标题。



这三个标签所代表的内容是包含关系，红框包含了蓝框部分，蓝框部分又包含了标题部分。

这是因为代码也是包含关系如下：

```
<html>
<head>
  <title>hello</title>
</head>
</html>
```

3. 学习一个重要标签<body>

在一个网页中，除了上面的标题，最重要的是网页的内容部分，如下图，蓝线部分是网页的标题，而下面主要的展示区域就是网页的内容部分了。



网页的内容部分通过<body>标签进行描述，由于网页内容是属于网页的，所以<body>标签要在<html>标签内部；然后由于内容部分在头部标题的下方，所以<body>标签要处于<head>标签下面。

所以一个完整的网页代码如下：

```
<html>
<head>
  <title>hello</title>
</head>
<body>
  this is a web page
</body>
</html>
```

4. <body>作用

<body>标签中间包含的内容，会显示在浏览器的内容区域，所以上面代码的效果如下：



大家可以随意修改 body 标签中间的部分，然后保存后，再打开网页，查看效果。

5. 小结

html 表示整个页面。

head 代表网页的头部，title 是头部的文字部分。

body 是网页的内容部分。

感觉还是比较简单的，多加练习吧！

零基础 HTML 入门教程(7)--支持中文的技巧

1. 网页中文会乱码

之前的章节中，我们陆续学习了 4 个标签：

- html：整个网页
- head：网页头部
- title：网页头部的标题
- body：网页内容

通过这四个标签，我们可以编写一个简单的网页：

```
<html>
<head>
  <title>我的网页</title>
</head>
<body>
  这是我的个人网页
</body>
</html>
```

打开后，效果如下，可以发现中文部分乱码了：



2. 为啥会乱码

那么为何会乱码呢？

这就要从 HTML 历史说起了，HTML 发明者蒂姆·伯纳斯·李是英国人，他会英语啊，不会中文。

所以人家在创立语言之处，肯定它就支持英语，你看这些标签其实都是英文单词或者英文单词的缩写。

所以在网页标准里面，天生就是支持英文的。

但是中文的话，就不是天生支持的那么好的，这就需要我们告诉浏览器——请支持中文吧！

3. 让浏览器支持中文

那么如何让浏览器支持显示中文呢，还是采用 HTML 惯用的方式——标签。

我们可以通过<meta>标签设置网页的一些特性，例如支持的语言编码。由于是设置网页的整个特性，所以它也是位于<head>标签之内的。

通过设置标签的 charset 属性为 utf-8 即可让网页支持中文，注意 utf-8 是最常用的中文编码。

```
<head>
  <meta charset="utf-8">
  <title>我的网页</title>
</head>
```

4. 标签的属性

在标签中间的部分，叫做标签的内容。内容一般会显示在网页上。

例如<title>我的网页</title>中，我的网页就是<title>标签的内容。

而在开始便签内部的，叫做标签的属性，属性一般用来给标签增加描述，是一种参数设置，但是一般不会在页面上显示。

例如<meta charset="utf-8">中，charset 就是标签的<meta>标签的属性，而 charset="utf-8"表示给该属性设定了值为 utf-8。

5. 小结

本篇介绍了如何在网页上正确的显示中文，如下：

```
<html>
<head>
  <meta charset="utf-8">
  <title>我的网页</title>
</head>
<body>
  这是我的个人网页
</body>
</html>
```

注意如果对网页的内容、属性还觉得不好理解的话，就暂时记住就是了，只要在网页头部添加<meta charset="utf-8">，即可让网页支持中文。

零基础 HTML 入门教程(8)--设置网页标题

1. 标题的作用

很多东西都得有标题、题目。

诗歌要有个题目，传达中心含义。书本要有个题目，揭示书籍内容。

网页也需要有标题，概括网页的内容。需要注意的是，本篇内容讲述的标题不是网页头部的 `<title>` 标题，而是网页 `<body>` 内容区域里面的标题。

从英文来讲，比较好区分，网页头部标题是 title，而内容区域的标题是 heading。

2. 如何编写 HTML 标题

HTML 的标题有 6 个级别，对应标签分别是 `<h1>`、`<h2>`、`<h3>`、`<h4>`、`<h5>`、`<h6>`。

注意 h1 最大，h6 最小。

我们以一首诗为例，编写一个带标题的页面。

```
<html>
<head>
  <meta charset="utf-8">
  <title>春晓</title>
</head>
<body>
```

```
<h1>春晓</h1>

春眠不觉晓，处处闻啼鸟。夜来风雨声，花落知多少。

</body>
</html>
```

效果如下，注意上面的红线部分是<title>标题，而下面的红线部分是<h1>标题。



3. 标题大小对比

我们通过一个网页来看下 h1-h6 的大小区别：

```
<html>
<head>
  <meta charset="utf-8">
  <title>标题大小对比</title>
</head>
<body>
```

```
<h1>这是 1 级标题</h1>  
<h2>这是 2 级标题</h2>  
<h3>这是 3 级标题</h3>  
<h4>这是 4 级标题</h4>  
<h5>这是 5 级标题</h5>  
<h6>这是 6 级标题</h6>  
</body>  
</html>
```

效果如下：



4. 小结

实际网页开发中，标题的样式效果要比 heading 标签复杂一些，这个在以后的学习中我们会慢慢接触。

本章掌握 h1-h6 的用法就 OK 了。

零基础 HTML 入门教程(9)--添加网页段落

1.何为段落

段落者，英文名 paragraph 也，表示一篇文章的一个分段。

一般来说，每个分段的最前面是要缩进 2 个字符的，同时不同的分段之间会换行。

2. HTML 中的段落

如果你以为 HTML 会使用 `<paragraph>` 作为段落的标签，那你就太天真了。

因为 paragraph 这个单词实在是有点长，太难记了，所以我严重的怀疑 HTML 创立者因为这个原因，将其简化为了 `<p>` 标签。

所以 HTML 中，使用 `<p>` 标签来描述一个段落。

有了段落之后，我们就可以做一个完整的网页了，可以轻松的实现分段功能。

```
<html>
<head>
  <meta charset="utf-8">
  <title>段落使用示例</title>
</head>
<body>
  <h1>假如生活欺骗了你</h1>
  <p>假如生活欺骗了你</p>
  <p>不要悲伤，不要心急</p>
  <p>忧郁的日子里须要镇静</p>
```

```
<p>相信吧，快乐的日子将会来临</p>
<p>心儿永远向往着未来</p>
<p>现在却常是忧郁</p>
<p>一切都是瞬息，一切都将会过去</p>
<p>而那过去了的，就会成为亲切的怀恋</p>
</body>
</html>
```

效果如下，可见有了段落，页面的排版更加的井井有条了。



3. 小结

段落，让诗歌更美好！

零基础 HTML 入门教程(10)--块级元素与行级元素

1. 块级元素

上一篇我们讲了<p>标签，它表示一个段落，在该便签后面会自动换行。

所以<p>标签是一个块级元素，它独自形成一块，跟后面的内容不会出现在同一行。

2. 行级元素

行级元素也可以称之为内联元素，就是在行的内部进行连接。所以行级元素是可以共同出现在同一行的，行级元素之后并不会换行。

3. div 标签

<div>标签是一个标准的块级元素，它的效果跟<p>标签类似，都会将内容作为单独的块显示。

如果说这两个标签有何区别的话，p 标签更多的表达了段落的意义。而如果写的一段代码想让它成为单独一块，但是这一块也不是一个段落，就可以采用 div。

例如，一首诗歌，每一句要断行，但是每一句其实不算是一个段落，此时可以用 div 标签。

```
<body>
  <h1>春晓</h1>
  <div>春眠不觉晓</div>
  <div>处处闻啼鸟</div>
  <div>夜来风雨声</div>
```

```
<div>花落知多少</div>  
</body>
```

效果如下：



再多说一些的话，其实用 p 和 div 都行，一般来说 div 用的更多一点。

4. span 标签

span 标签是行级元素，它包裹的内容不会换行，如下代码：

```
<body>  
  我是<span>中国</span>人  
</body>
```

效果如下，可以发现 span 包裹的内容和前后的内容同处一行，而且效果上也没啥差别。



那么此时可能就会有疑问了，要 `span` 标签有何用？就像《悟空》里面唱的，我要这铁棒有啥用捏？

设计了这个标签，那必然是得有点用的。其实 `span` 包裹起来的部分，可以单独为其设置效果。比如在后面我们学习了文字颜色设置之后，我们只想把中国二字设为红色，其余字颜色不变。

那么我们就需要 `span` 标签将中国包起来，然后对该标签应用特定的文字样式。

这就是 `span` 存在的意义，它可以区分不同的内容，以便对不同的内容应用不同的样式或者行为。

5. 小结

说了这么多，其实只需要掌握 `div` 换行、`span` 不换行。

零基础 HTML 入门教程(11)--初识 VSCode

1. 编辑器

在之前的章节中，我们一直使用最原始的记事本进行代码的编辑。然后将文件拓展名修改为.html后，通过浏览器进行查看。

随着我们学习的标签种类越来越多，大家可能已经发现了，用记事本写一段网页代码是非常费劲的。

所以本篇，我们就来给大家介绍一款超好用的代码编辑器，也是当前前端开发领域最流行的编辑器——VSCode。

2. 编辑器是干啥的

注意，如果是同样的一段代码，使用记事本编写，或者使用 VSCode 等编辑器进行编写，其实写出来的东西是一模一样的，都是代码。

但是，编辑器可以让编写、查看代码更加清晰、直观、高效。例如可以用颜色区分代码的不同部分，帮助大家理解代码；可以自动联想补全代码，提高大家的编写效率；可以自动为代码进行格式化，让代码更加井井有条，帮助大家更加直观的查看代码。

所以编辑器，实际上是一种提高效率的工具。

3. VSCode 编辑器的特长

网页编辑器有很多，为何 VSCode 能够独领风骚？

我认为 VSCode 成功的一个关键是插件机制，就是可以通过安装各种插件来拓展 VSCode 的功能。

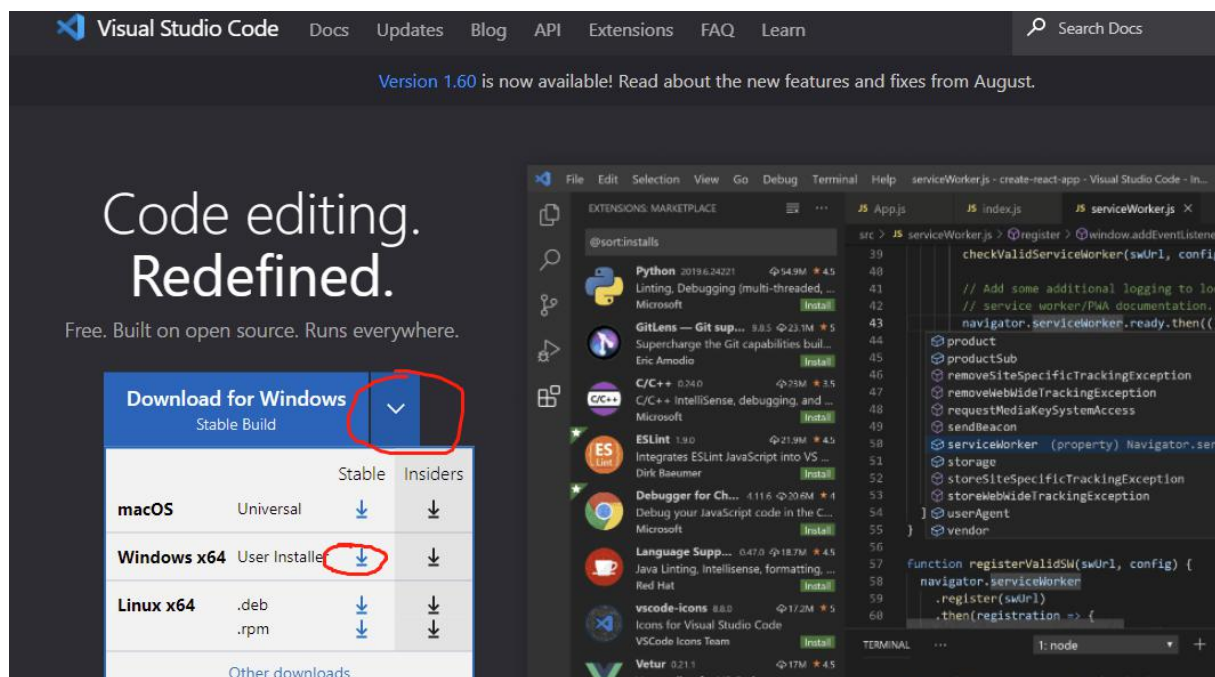
例如 VSCode 默认是英文版本的，可以安装一个中文插件，就能显示中文版本了。

通过插件机制，VSCode 具备了无限可能性！

4. VSCode 下载与安装

说了半天，我们来下载并安装 VSCode，其实 VSCode 也是一个软件，下载后双击安装包安装就是了。

首先打开下载地址：[VSCode 下载地址](#)，然后选择 Windows 版本下载，如下图：



点击上图中圈出的下载链接，然后稍等一会就会启动下载。

5. VSCode 安装

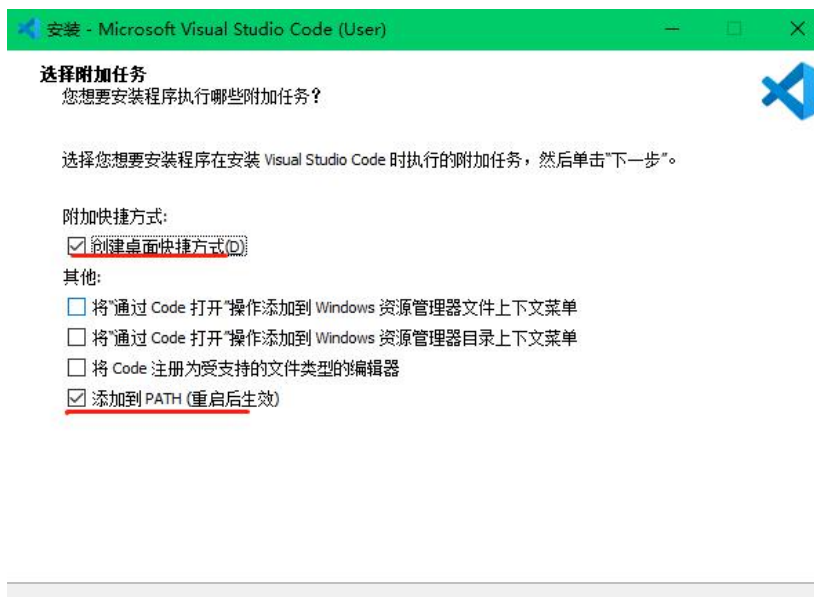
下载完毕后，我们得到了一个安装包，它是一个 exe 可执行程序。双击这个程序，即可开始安装 VS Code。

接下来，我通过截图的方式，详细给大家描述下安装过程，以便让大家明白我们安装过程的细节。

第 1 步，是要同意安装协议，相当于你要使用人家这个软件，就得同意人家的一些条件，这个嘛同意就行，然后点击下一步。



第 2 步，选择红线标出的两个复选框，创建桌面快捷方式很好理解，就是在桌面上放一个图标，可以双击打开 VSCode；添加到 PATH 的意思是将 VSCode 添加到系统环境变量，这样我们在使用命令行等工具的时候，可以轻松调用到 VSCode。注意添加到 PATH 这个如果不理解的话，也没啥影响，感兴趣的同学可以自己再去搜索了解。



第 3 步，直接点击安装就 OK 了。



最后，安装完毕后，可以双击桌面上的 VSCode 图标启动 VSCode 进行使用。

6. 小结

本篇介绍了 VSCode 的特点及安装过程，下一篇讲解下 VSCode 的常见用法。

零基础 HTML 入门教程(12)--VSCode 基础用法

1. 前言

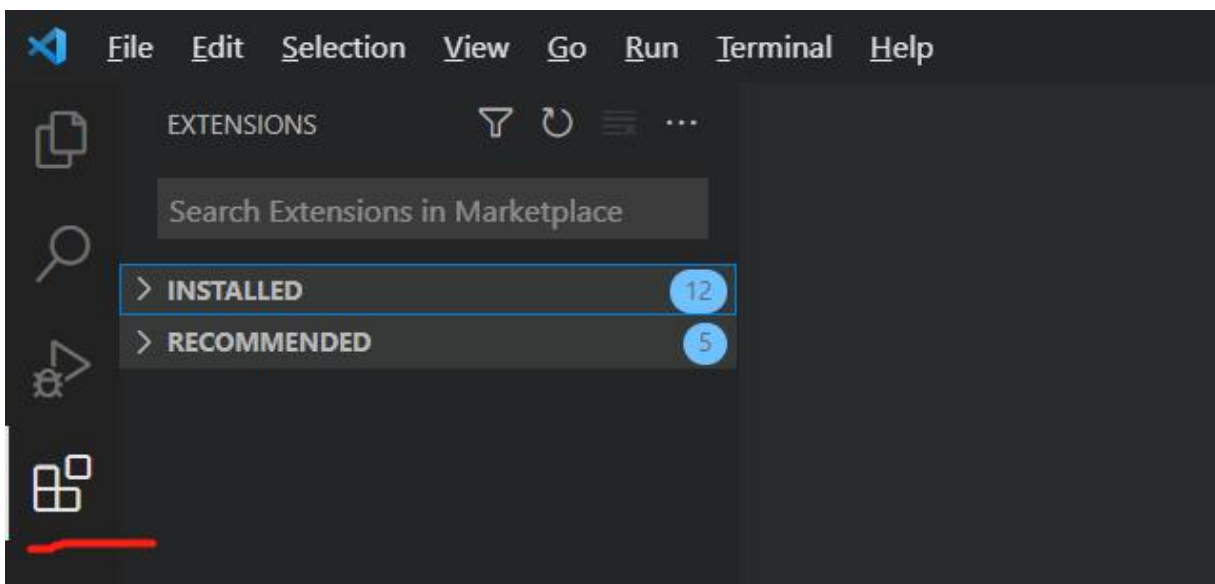
VSCode 虽然是一个代码编辑器，没有从本质上改变编写代码的含义。但是 VSCode 的使用还是比较复杂的，为啥复杂呢，因为它提供了各式各样的功能，功能多，自然就复杂了。

所以还是要给各位新同学介绍下 VSCode 的基本用法，便于大家快速的上手操作。

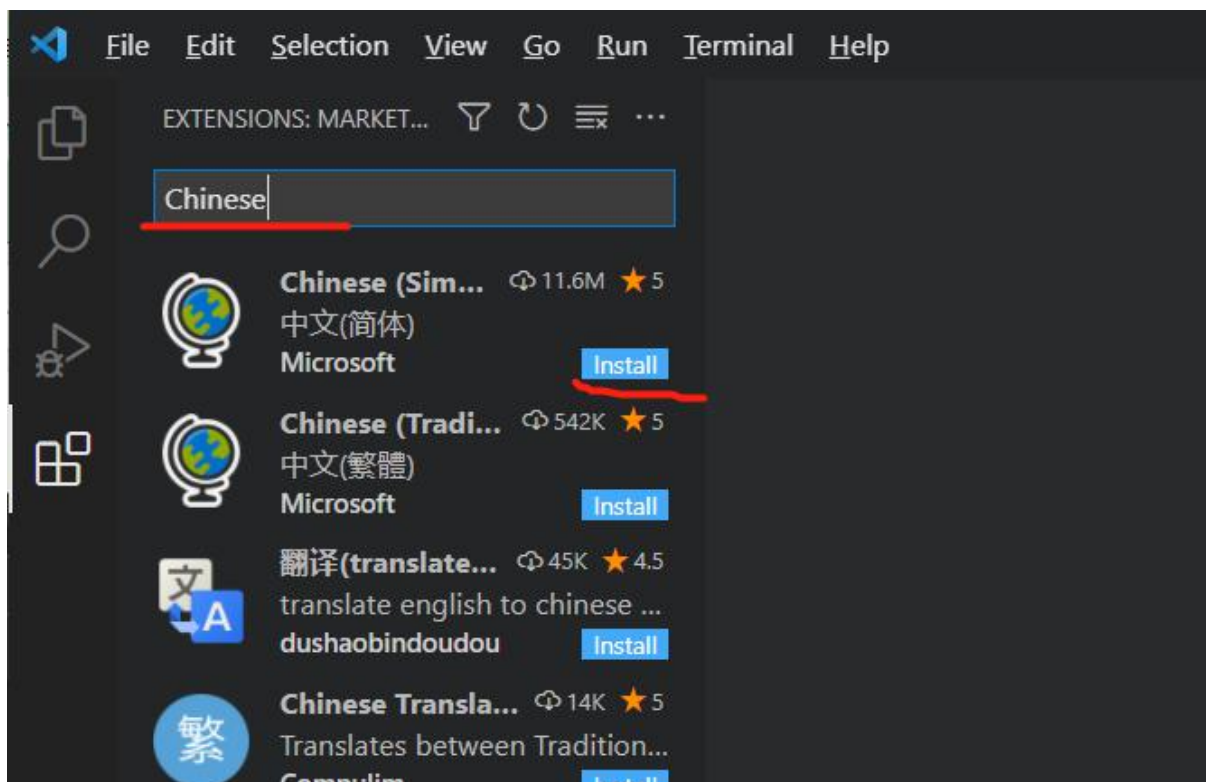
2. 界面修改为中文

首先就是要将默认的英文界面，修改为熟悉的中文界面了，这样起码英语并不是很好的同学上手起来就轻松多了。

首先点击插件拓展菜单，点击后，可以搜索想要的插件进行功能拓展。



然后在搜索框输入 Chinese，表示咱们要查找中文相关的插件，搜索结果如下，点击 Install 按钮安装即可。



安装完毕后，退出重启 VSCode，界面就是中文的了。

3. 管理项目文件夹

大家注意，不管是写什么代码，都得把代码文件管理好。

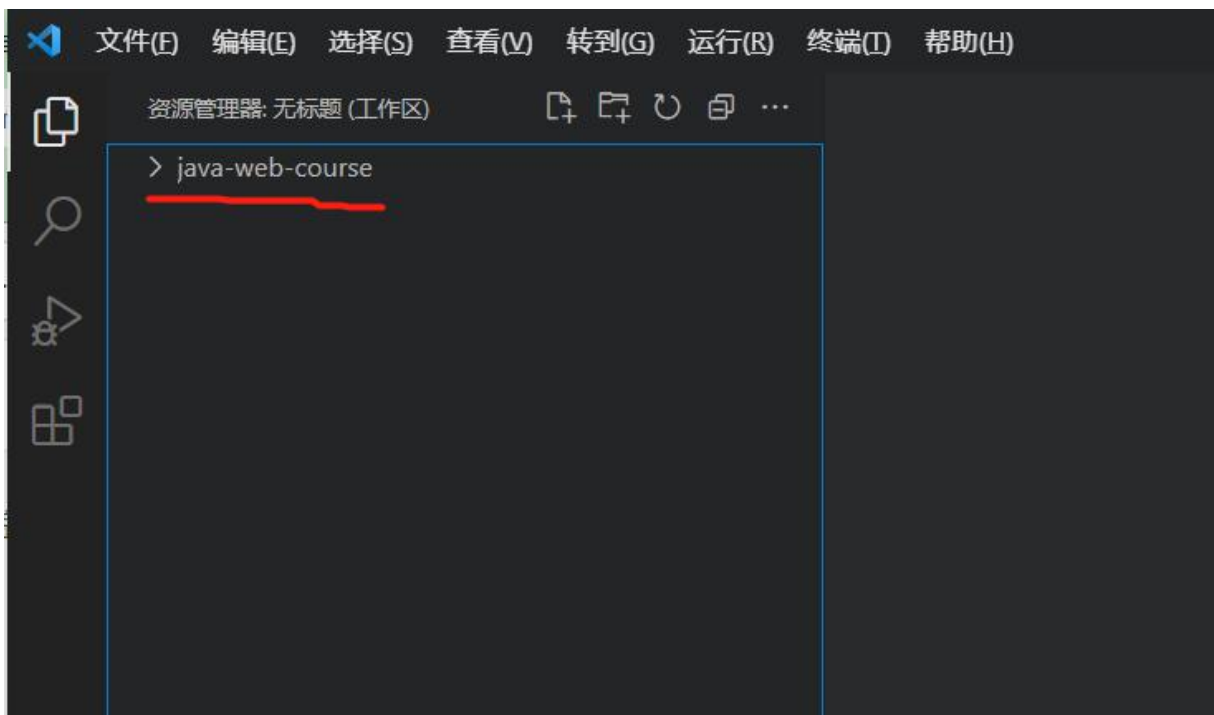
我见过太多的同学，编写的代码文件东一个、西一个，每次让我帮忙调试代码的时候，自己都找不到代码文件在哪。

其实，管理代码文件，就跟管理普通文件一样，应该按文件夹分门别类放好，同一个项目的代码文件应该都放到一个文件夹下，便于集中管理。

例如，我们现在学习的是 JavaWeb 教程，所以可以建立一个文件夹 java-web-course，学习过程中的相关文件都可以放到这个文件夹中。

然后可以使用 VSCode，将整个文件夹添加到工作区中，这样就可以在 VSCode 左侧工作区进行快捷操作了。

点击 VSCode 菜单栏的【文件】-【将文件夹添加到工作区】，然后选中刚刚建立的 java-web-course 文件夹，即可在工作区对其进行方便快捷的管理了。



4. 添加文件夹

JavaWeb 课程有很多内容，包括 html、css、js、bootstrap、java、数据库等等。

所以我们可以先在 java-web-course 文件夹下再建立一些子文件夹，便于对这些内容分类管理。

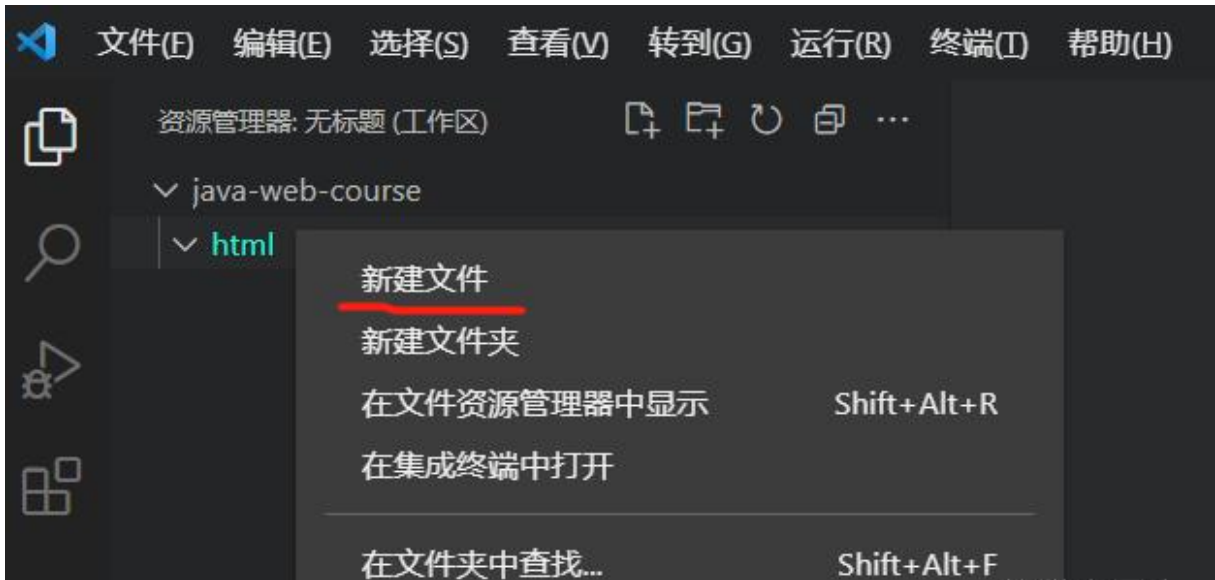
直接在工作区，右键点击 java-web-course，在弹出菜单中点击【新建文件夹】。



然后将新的文件夹命名为【html】，以后编写的 html 文件都可以放到这个文件夹下，便于统一管理。

5. 添加网页文件

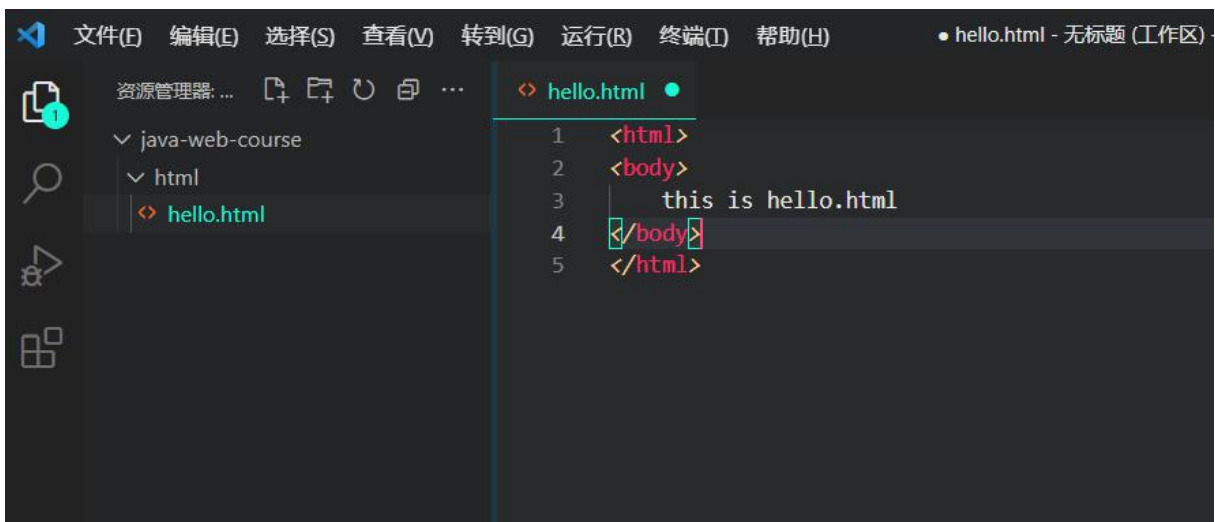
右键点击刚刚建立的【html】文件夹，在弹出的菜单中选择【新建文件】。



将新建文件命名为 `hello.html`，这样它就是一个网页文件了。

6. 编辑文件

单击【`hello.html`】文件，右侧内容区即可对网页代码进行修改，我们随便编写一段代码如下：



编辑完成后，注意保存，可以点击【文件】-【保存】，或者直接按【`Ctrl+S`】快捷键保存。

我们的代码只有保存后，才能在查看时生效。

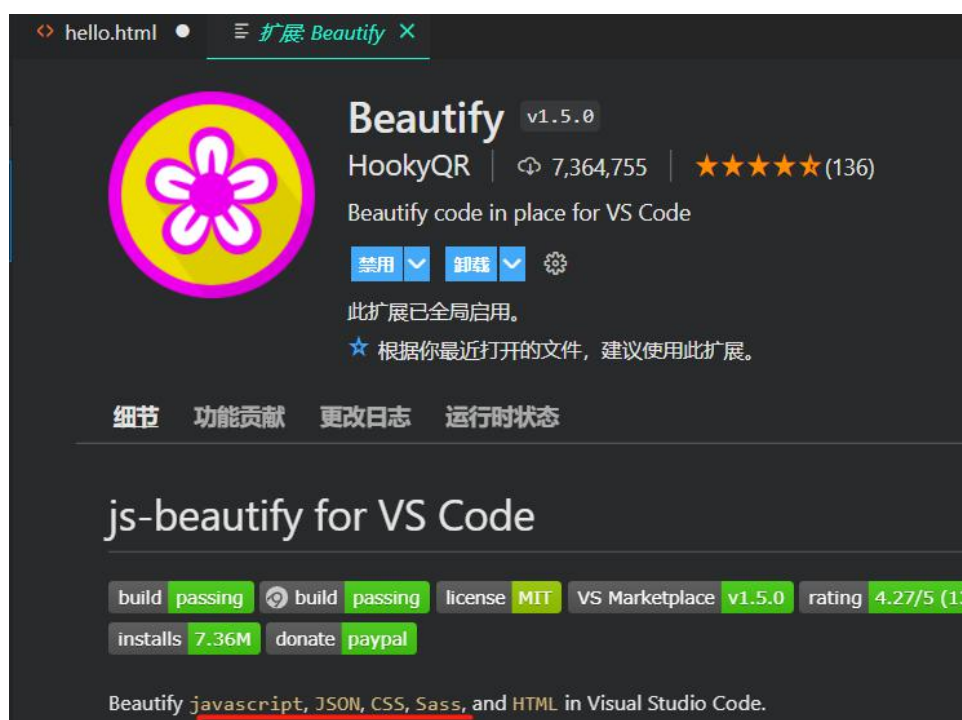
7. 代码格式化

代码格式化，是指让代码排版更加美观好看，一般我们会让子标签相对于父标签进行缩进，以便能看出代码的层次结构，更加利于对代码的理解。

VSCode 可以通过【Alt+Shift+F】快捷键对代码进行格式化，非常方便。

在这之前，我们还需要安装一个插件，以便让 VSCode 支持对 html 代码的格式化。（毕竟语言种类这么多，VSCode 也不可能上来就支持所有语言的格式化，所以还是通过安装对应插件的方式提供这一个功能）。

我们选择安装 Beautify 插件即可，通过该插件的说明，我们可以了解到该插件可以同时支持 javascript、JSON、CSS、HTML 等文件的格式化，可以说是非常强大了。至于如何安装该插件，与安装中文支持插件一样，搜索后点击【Install】安装即可。



8. 代码格式化说明

此时我们按【Ctrl+Alt+F】对刚刚的代码进行格式化，如下：

```
<html>

<body>
  this is hello.html
</body>

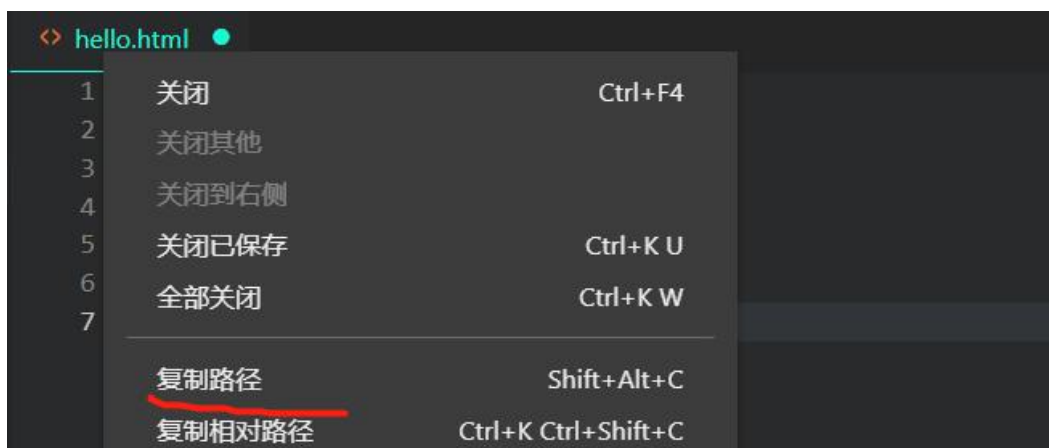
</html>
```

格式化后的代码，会非常规范，当然目前的例子中，代码很少，还看不出来。如果写上一大堆代码，然后格式化，会明显的发现这些代码会排的整整齐齐。

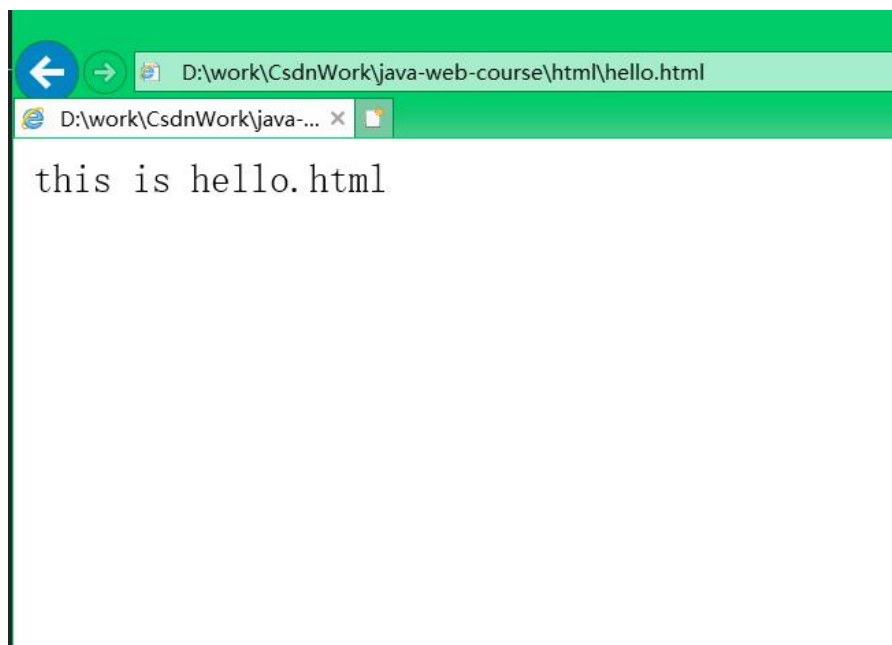
在初学者阶段，还是强烈建议大家多学习代码的格式化，看看插件是如何让代码井井有条的，然后大家在编写代码的时候，也提醒自己安装格式化的规范去编写漂亮的代码。

9. 预览网页

使用鼠标指向网页文件的标签，点击右键，选择【复制路径】。



然后打开浏览器，将复制的内容【粘贴】到地址栏，然后按回车即可查看网页效果了：



10. 快速预览网页

也可以通过安装【Open In Default Browser】插件后，在网页编辑页面点击【Ctrl+1】快捷键，则会自动弹出浏览器对当前网页进行预览，更加的方便快捷。

11. 小结

VSCode 用法千千万，还需要大家多多使用，熟能生巧。

编程，本质上是一种技能，需要反复的锤炼。

就好比开车，老司机是怎么练成的，无他，但手熟尔！

零基础 HTML 入门教程(13)--网页类型声明

1. 前言

如果我们买了一部苹果手机，然后要研究它如何使用，我们得先知道这部手机的版本，是 iPhone4，还是 iPhone5/6/7…。

同样，如果我们买了一辆汽车，也应该先知道它的型号，比如同一款车型，也得分上个 2019 款/2020 款/2021 款。

对于 HTML 来说，也是一样的，现在很火的一个名词 H5，其实就是指的 HTML5，这个 5 就是版本号。

2. HTML 版本号

目前来说，基本上浏览器对 HTML5 的支持都很到位了，所以绝大多数程序员编写的网页都是 HTML5 的。

但是由于历史原因，还有一部分网页是 HTML4 的。

好的，那么如何区分一个网页是 HTML4 还是 HTML5 的呢？浏览器如何知道程序员编写的 HTML 代码是想用 4 代的规范还是 5 代的规范呢？毕竟不同版本还是有区别的。

这就可以通过网页类型声明来解决。

3. 网页类型声明

常用的网页类型声明有两种，第一种是 HTML5 的：

```
<!DOCTYPE html>
```

第二种是 HTML4 的：

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

实际上我们只需要记住 HTML5 的声明就 OK 了，毕竟现在我们写网页肯定是用最新的版本，效果更好！

4. 一个完整的网页代码

所以，我们在正儿八经编写第一个网页的时候，网页的最开头处，是要写上网页类型声明的。

这样最起码，别人在看到咱们写的网页的时候，第一印象就是——嗯，挺专业的！

好的，一个完整的网页代码如下：

```
<!DOCTYPE html>  
  
<html>  
  
  <head>  
    <title>网页类型声明实例</title>  
    <meta charset="utf-8">  
  </head>  
  
  <body>
```

```
    你好，这是一个标准网页  
</body>  
  
</html>
```

5. 小结

其实对于大多数浏览器来说，不添加网页类型声明，照样正常运行。

但是咱们作为程序员，还是要尽量按照规范来写代码，写最规范最通用的代码，而不是满足于能用就行。

要有点追求的还是！

零基础 HTML 入门教程(14)--水平线

1. 水平线的概念

HTML 中有一个比较特别的标签，叫做水平线，写作`<hr>`。

该标签可以在网页上显示一条横线，一般用来分隔不同的网页内容。

2. 水平线的用法

使用方法很简单，在需要分割的地方，添加一个`<hr>`标签即可。

例如：

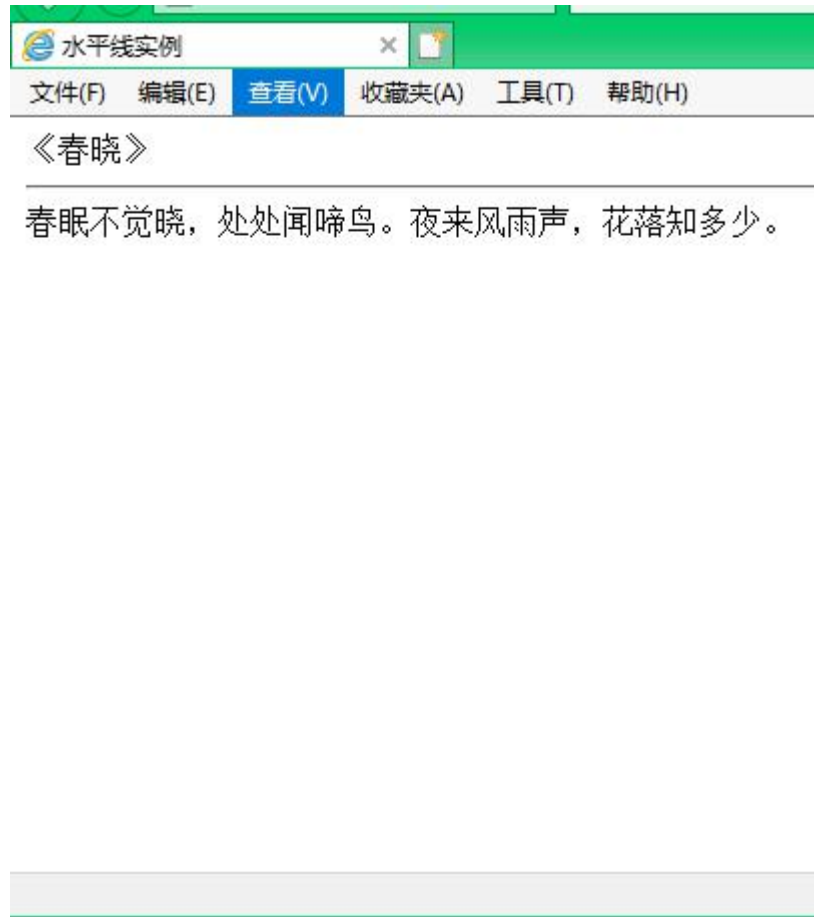
```
<!DOCTYPE html>
<html>

<head>
  <title>水平线实例</title>
  <meta charset="utf-8">
</head>

<body>
  《春晓》
  <hr>
  春眠不觉晓，处处闻啼鸟。夜来风雨声，花落知多少。
</body>

</html>
```

在诗歌的标题和内容直接，添加了一个水平线，所以效果如下：



3. 小结

水平线的使用非常简单，就不用再怎么总结了。

别小看水平线，在真实网页开发中，还经常会用到它。不错的标签！

零基础 HTML 入门教程(15)--注释

1. 注释

几乎所有的语言都有注释，例如我本人学习过的 C/C++/OC/C#/Java/Python/HTML/CSS/JavaScript/SQL，这些语言都提供了注释功能。

可见注释是非常重要的。

那么什么是注释呢，在我们学习古代文言文或者诗词的过程中，可能已经多次接触过注释这个词了，它一般是对原文的字句进行进一步的解释。

也就是说，注释其实不属于原文，它是额外增加的解释部分，用途是方便阅读者对原文理解。

2. 程序注释

那么程序中的注释是什么作用呢，其实跟文言文注释类似。

第一，注释的作用是解释，也就是说，程序中的注释的作用是对程序代码进行解释。

第二，注释不属于原文，也就是说，注释不属于程序代码，它对程序的运行没有任何影响，只是起到一个提示作用。

总结下，程序中的注释，其实是用来解释程序代码的，一般用于提醒程序员这段代码的作用、注意事项，但是注释对程序本身是没有影响的。

3. HTML 注释

HTML 语言也支持注释，它的格式为`<!-- xxxx -->`，中间的 xxxx 部分就是被注释的内容，注释的代码对网页毫无影响，是用来提示程序员的。

我们给出一个示例：

```
<body>
    《春晓》
    <!-- hr 表示一个水平分割线 -->
    <hr>
    春眠不觉晓，处处闻啼鸟。夜来风雨声，花落知多少。
</body>
```

在网页中预览，效果如下，大家能够发现，注释中间的内容并没有在网页上显示，不会对网页产生影响。

但是程序员在看这段代码的时候，还是能得到一些提示信息的。

《春晓》

春眠不觉晓，处处闻啼鸟。夜来风雨声，花落知多少。

4. 注释的作用

其实注释是非常重要的，尤其是大型项目。

大型的项目，比如大型的网页，里面的东西非常多，我们放眼望去，可能会比较迷惑。

此时如果有一些注释，指引我们去理解网页内容，肯定是非常有帮助的。

就好比说明书，虽然没有真正作用，但是起到了说明的效果。

5. 小结

注释很重要，大家以后会慢慢体会到的！

零基础 HTML 入门教程(16)--理解标签属性

1. 前言

都说了，HTML 是超文本传输语言，之前讲述的内容还都在文本范围内，马上就要讲到一个超出文本的标签了——图像。

但在讲述图像标签的使用之前，我们先来了解下内容和属性的区别。

2. 标签的内容

内容是比较好理解的，就是标签中间包裹的部分，例如<h1>我的博客</h1>，中间的我的博客就是 h1 标签的内容。

内容是直接显示到网页上的东西，所以我的博客会在网页上直接显示。

3. 标签的属性

但是还有一种情况，就是一些信息并不会在网页上显示，但是也是需要提供的辅助信息。

例如之前接触到的设置网页为中文 meta 标签。

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

在这个例子中，utf-8 表示一种编码方式，它是一个参数信息，用户告诉浏览器当前页面采用中文编码。

但是这个信息是没必要显示到网页上的，没必要让浏览器的用户看到，这种就叫做属性，属性一般是放到标签的开头部分的。

4. 标签的格式

所以，了解属性和内容之后，我们可以知道标签的格式，无非如下：

```
<label property="xxx">yyy</label>
```


对上面的内容，label 是标签，yyy 是内容，property 是属性名，xxx 是属性值。

5. 小结

本章主要是让大家了解下标签的属性与内容的区别，内容一般是显示到页面上的内容，而属性一般是提供一些附加的描述信息。

零基础 HTML 入门教程(17)--图像

1. 前言

图像标签是, 作用是在网页上显示图片。

2. 显示一张图片

图片没法用文字描述, 很难把图片放到 img 标签的内容部分。

但是图片一般是一个文件, 我们可以通过源属性, 将图片显示到 img 标签中。

首先将图片文件 1.jpg 和我们的网页 hello.html 放到一个文件下, 然后编写代码如下:

```
<!DOCTYPE html>
<html>

<head>
  <title>图像实例</title>
  <meta charset="utf-8">
</head>

<body>
  <p>
    显示一个图片
    
  </p>
</body>

</html>
```


当浏览器在处理到 `img` 标签时,发现该标签 `src` 属性的值为 `1.jpg`,所以该标签就会加载该图片文件。

所以效果如下:



3. 图片的可替换文本


有时候图片不小心丢失了,当然一般情况下是不会发生的,但是如果发生了,则图片无法正常显示,效果如下:

显示一个图片 

这样看起来就不是很友好了,此时我们可以给图片添加一个可替换文本,当图片不能正常显示时,可以显示一段文本内容替换。

```
<p>  
    显示一个图片  
      
</p>
```

其实效果也不是很好,所以还是尽量要保证图片是正确的、存在的。

显示一个图片 java图片

4. 将图片放入文件夹

如果图片比较多，那么图片都跟网页混在一起，会很杂乱。

我们可以建立一个 images 文件夹下，将 1.jpg 放入其中，此时 hello.html 与 imgs 处于同一目录。所以代码更改为：

```
<p>
  显示一个图片
  
</p>
```

解释下，./表示当前目录，然后./images/表示当前目录的 images 文件夹下，最后./images/1.jpg 就表示跟当前网页同目录的 images 文件夹下的 1.jpg 文件。

另外，当访问当前目录的时候，可以省略./，所以上述写法等同于：

```

```

5. 小结与拓展

本篇讲述了如何在网页上显示图像，以及如何加载文件夹下的图像。

其实还可以加载当前文件上层目录下的图片，使用的语法为../，..就表示当前文件所在目录的上级目录。

这个比较复杂，暂时不需要大家掌握，感兴趣的同学可以自行去尝试。

零基础 HTML 入门教程(18)--超级链接

1. 前言

超级链接是 HTML 中极其精彩的设计。

网页之所以让人感觉功能无限，丰富多彩，就在于网页上的超级链接，当我们点击超级链接时，可以跳转到一个新的页面，查看该链接对应的内容。

在一层一层的链接中，网站的内容得到极大的延展，用户也得以从互联网上获取海量的信息。

本篇我们就来讲解下超级链接的使用。

2. 跳转指定网址页面

可以使用超级链接跳转指定网址的页面，例如下面的例子会跳转到百度：

```
<a href="http://www.baidu.com">百度</a>
```

效果如下，点击文本百度，则跳转到对应的网址：<http://www.baidu.com>。



3. 跳转当前网站其他网页

href 属性也可以指向当前网站的其他网页，例如：

```
<a href="1.html">网页 1</a>
<a href="pages/2.html">网页 2</a>
```

则点击【网页 1】，会跳转到当前网页同目录下的 1.html；而点击【网页 2】，会跳转到 pages 文件下的 2.html。

4. 在新窗口中打开网页

有时候，我们点击超级链接时，并不希望新网页替换当前网页，而是弹出一个新窗口加载新页面，这样的好处是还可以轻松返回查看老页面。

此时可以使用 target 属性，当 target 属性值为_blank 时，点击超链接会弹出新窗口打开网页，实例如下：

```
<a href="http://www.baidu.com" target="_blank">百度</a>
```

5. 小结

超级链接非常重要，要掌握本节的内容，需要自己编写网页逐一实现上面的要点。

零基础 HTML 入门教程(19)--列表

1. 列表的概念

一个元素其实就是一个，那么多个元素就可以组合为列表。既然叫列表，那么列表的元素一般是垂直排列的，所以才叫列表。

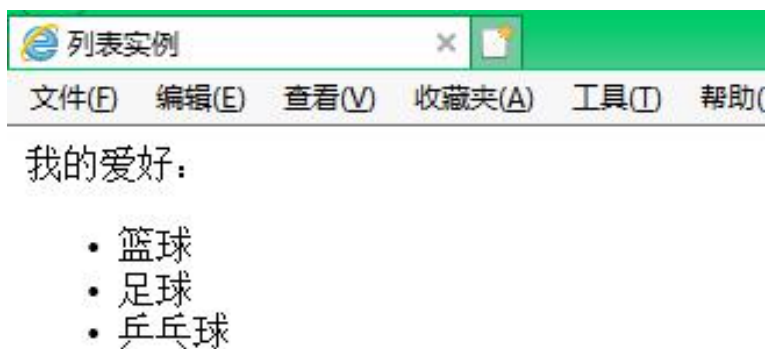
2. 无序列表

无序列表应用于列表的元素没有先后顺序的场景，例如我的爱好有：篮球、足球、乒乓球，这三者没啥先后顺序，则可以使用无序列表表示。

无序列表的标签为``，列表中的每个元素标签为``，所以上面的例子代码为：

```
<span>我的爱好：</span>
<ul>
  <li>篮球</li>
  <li>足球</li>
  <li>乒乓球</li>
</ul>
```

效果如下：



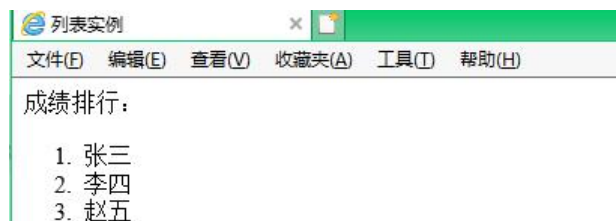
3. 有序列表

有序列表应用于元素有顺序的场景，例如学生成绩排行列表。

有序列表的标签为``，列表中的每个元素标签还是``，所以上面的例子代码为：

```
<span>成绩排行：</span>
<ol>
  <li>张三</li>
  <li>李四</li>
  <li>赵五</li>
</ol>
```

效果如下：



可见有序列表会自动为列表项添加序号。

4. 小结

看上面的例子中，无序列表和有序列表样式都不算好看，但是务必注意，HTML 主要是为了表达内容。

后续在美观部分，我们学习 CSS 后，可以将列表做的更加好看。

零基础 HTML 入门教程(20)--换行

1. 前言

我们之前学习过段落标签<p>, 该标签中间的内容会形成一个段落, 不同的段落之间会有换行的效果。

我们也可以直接使用换行标签
, 该标签的作用就是生成一个换行。

2. 换行标签 br

比如我们想写一首诗, 让诗歌的每一句都独占一行, 此时我们可以用<p>标签如下实现:

```
<p>
    假如生活欺骗了你
</p>
<p>
    不要悲伤, 不要心急
</p>
<p>
    忧郁的日子里需要寂静
</p>
<p>
    相信吧, 那快乐的日子即将来临
</p>
```

效果如下:

假如生活欺骗了你
不要悲伤，不要心急
忧郁的日子里需要寂静
相信吧，那快乐的日子即将来临

如果使用换行标签，代码如下：

```
假如生活欺骗了你<br>  
不要悲伤，不要心急<br>  
忧郁的日子里需要寂静<br>  
相信吧，那快乐的日子即将来临<br>
```

两者对比效果如下，可以发现，p 标签前后都换行，而 br 标签只是在当前位置生成一个换行。

假如生活欺骗了你
不要悲伤，不要心急
忧郁的日子里需要寂静
相信吧，那快乐的日子即将来临

假如生活欺骗了你
不要悲伤，不要心急
忧郁的日子里需要寂静
相信吧，那快乐的日子即将来临

3. 标签的闭合问题

想必大家也发现了，HTML 中有些元素是必须闭合的，比如：`<p></p>`，再比如`<html></html>`，这些标签有一个开头，然后有一个结束标签，中间就是内容部分。

但是有些标签，就没有结束标签，比如``，再比如`
`，因为这些标签中间也不需要内容。

这样就挺尴尬的，相当于有两种标准。

实际上，这些标签也可以结束，直接加一个斜杠即可，如``、`
`，这样写起来更加规范一些，都有头有尾嘛。

当然不写也行，直接写``、`
`也是符合 HTML 标准的，但是写上之后更规范，符合更高的标准！

这一点了解就行。

4. 小结

学习了一个新的标签 `br`，功能很简单就是换行。

零基础 HTML 入门教程(21)--表格

1. 前言

表格有多重要我就不需要赘述了吧，我们日常访问的网页处处都有表格，所以表格是非常非常重要的 HTML 元素。

本篇我们就来简单介绍下表格的用法。

2. 表格标签

我们可以使用 `<table>` 标签来定义一个表格，然后使用 `border` 属性来设定表格边框的宽度，如下。

注意此时表格里面还没有内容，所以网页也显示不出什么东西来。

```
<table border="1">

</table>
```

3. 表格的行与列

HTML 的表格是先编写行 `<tr>`，然后在行里面编写列 `<td>`，所以我们定义一个表格如下：

```
<table border="1">
  <tr>
    <td>张三</td>
    <td>男</td>
  </tr>
  <tr>
```

```
<td>李四</td>

<td>女</td>

</tr>

</table>
```

这个表格首先有 2 行，即 2 个 tr。然后每一行有两列，即两个 td。所以效果如下：



张三	男
李四	女

4. 页眉

表格一般需要有标题，我们可以把标题放到页眉里面。页眉标签为 `thead`，标题列的标签为 `th`。所以带页眉标题的代码如下。

需要注意的是，定义页眉后，我们一般将行列放到表格的主体 `tbody` 标签内部，这样更加规范。

```
<table border="1">

  <thead>

    <th>姓名</th>

    <th>性别</th>

  </thead>

  <tbody>

    <tr>

      <td>张三</td>

      <td>男</td>

    </tr>

    <tr>

      <td>李四</td>
```

```
        <td>女</td>
    </tr>
</tbody>
</table>
```

效果如下，可见标题列字体是加粗的。



姓名	性别
张三	男
李四	女

5. 页脚

表格还可以添加页脚，可以在页脚显示一些附加信息，例如学生的总数，页脚标签为 `tfoot`。添加页脚后代码如下：

```
<table border="1">
  <thead>
    <th>姓名</th>
    <th>性别</th>
  </thead>
  <tbody>
    <tr>
      <td>张三</td>
      <td>男</td>
    </tr>
    <tr>
      <td>李四</td>
      <td>女</td>
```

```
        </tr>
    </tbody>
    <tfoot>
        <tr>
            <td>
                学生总数:
            </td>
            <td>
                2 人
            </td>
        </tr>
    </tfoot>
</table>
```

对应效果如下：

姓名	性别
张三	男
李四	女
学生总数:	2人

一般页脚用的比较少。

6. 小结

HTML 表格主要表达了表格的内容，具体如何将表格做的好看，还得使用 CSS，后续课程我们会对 CSS 样式进行讲解。

零基础 HTML 入门教程(22)--表单

1. 前言

之前我们所讲的一些系列各种 HTML 标签，都是为了给用户展示一些东西。

实际上 HTML 是可以让用户输入信息的，例如登录时输入用户名密码；买东西时选择型号、颜色；发布博客时输入标题、内容。

HTML 通过表单及输入标签，来实现用户输入内容的收集。

2. 表单标签

表单标签，格式为 `<form>表单内容</form>`，中间的表单内容部分就是用来让用户输入内容的。

表单有两个重要的属性 `action` 和 `method`，如下：

```
<form action="#" method="get">

</form>
```

其中 `action` 属性表示表单提交后，哪个后台程序负责处理这个请求，这个涉及到我们 JavaWeb 后端学习的内容，大家到此大概知道这个参数的意思就行。`action="#"` 中 # 号并不是一个可以接受请求的地址，使用 # 的意思就是该表单目前不指定后台处理程序。

然后是 `method` 参数，这个也涉及网页前端和后台程序交互方式了，`method` 可以选择 `get/post` 值，这两个值代表着表单内容提交到后台的不同方式，具体也是等我们学习 JavaWeb 后端的时候具体了解，此时大体知道就行。

3. 表单输入标签

好的，现在我们知道如何编写表单了，还需要在表单内添加输入标签，输入标签的作用就是让用户输入东西的。

3.1 文本框

文本框用来让用户输入一段文本，格式如下：

```
<form action="#" method="POST">  
    请输入姓名: <input type="text" name="username"><br>  
</form>
```

其中<input>表示这是一个输入标签，type="text"表示这个输入标签的类型是文本框，最后 name="username"是给这个输入标签起了个名字，之所以要起名字，是当网页上有多个输入标签时，能把他们区分开。

文本框输入效果如下：



3.2 密码框

密码框也是用来输入文本的，但是密码框为了保护密码的私密性，输入的内容会被隐藏，别人是看不到的，例如：

```
请输入与密码: <input type="password" name="password"><br>
```

效果如下:



3.3 单选框

单选框用于从若干选项中选择其中一个, 例如性别只能从男、女中选择一个:

```
<form action="#" method="POST">
  请选择性别:
  <input type="radio" name="sex" value="male">男
  <input type="radio" name="sex" value="female">女
</form>
```

注意这两个单选框都是表示性别, 所以 name 相同。还有一个重要属性是 value, 当我们的表单提交给后台程序时, 会将选中项的 value 值提交到后台, 这样后台就知道用户从网页上选择了哪个选项了。

上面代码效果如下:



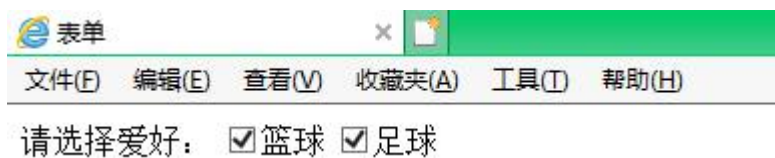
3.4 复选框

单选框只能选择一个，复选框可以选择多个，例如可以选择兴趣爱好：

```
<form action="#" method="POST">
    请选择爱好：
    <input type="checkbox" name="hobby" value="basketball">篮球
    <input type="checkbox" name="hobby" value="football">足球
</form>
```

由于两个复选框都是表示爱好，所以 name 相同。当表单提交到后台时，会将选中项的 value 值全部传递给后台程序。

上面的例子效果如下：



3.5 按钮

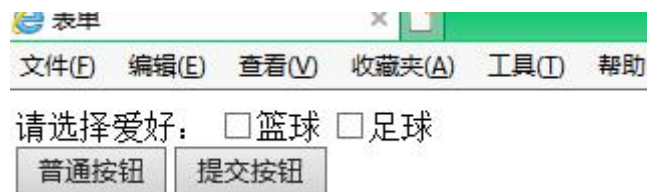
表单内还需要包含按钮，一般来说，点击按钮后会将表单提交给后台。

按钮其实分 2 大类，提交按钮点击后会提交表单，但是普通按钮点击后则不会提交表单，例如：

```
<form action="#" method="POST">
    请选择爱好：
    <input type="checkbox" name="hobby" value="basketball">篮球
    <input type="checkbox" name="hobby" value="football">足球
    <br>
```

```
<input type="button" value="普通按钮">
<input type="submit" value="提交按钮">
</form>
```

效果如下：



可以尝试下，点击普通按钮没反应，但是点提交按钮，页面会刷新一下。

这样可能不够直观，我们修改下表单提交的 action 如下：

```
<form action="http://www.baidu.com" method="POST">
  请选择爱好：
  <input type="checkbox" name="hobby" value="basketball">篮球
  <input type="checkbox" name="hobby" value="football">足球
  <br>
  <input type="button" value="普通按钮">
  <input type="submit" value="提交按钮">
</form>
```

也就是说，我们的表单提交给百度去处理(当然百度不会搭理我们，因为百度不是我们开发的，不会理会我们的请求)。

此时点击普通按钮，没反应，因为表单不会提交。而点击提交按钮，则会进入百度网页，表示当前信息交给百度处理了。

4. 小结

表单是用来让用户输入信息的，网页正是因为有了交互性，才变得生动有趣！

另外大家可能对前端、后台的概念不够熟悉，此处稍微解释下。

前端就是网页能看到的部分，而后台是网页的内容提交后进行分析处理的程序。我们目前学习的 HTML 属于网页前端开发，因为 HTML 的内容都会显示在网页中。

零基础 HTML 入门教程(23)--综合实例

1. 背景

通过前面 22 篇文章的学习，我们对 HTML 有了一个比较全面的了解。

本篇，我们编写一个网站实例，来看看通过已经掌握的 HTML 知识，能开发出什么样子的网站。

2. 开发流程

2.1 网站功能设计

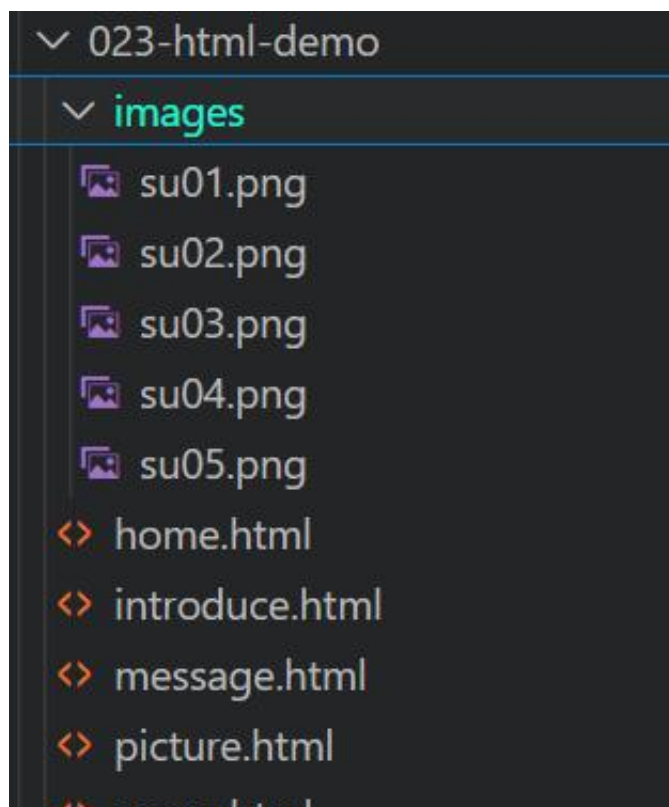
我们打算开发一个介绍苏轼的网站，没错，就是那个鼎鼎有名的大文人苏轼，网站我们就起名为【苏轼网】。

我们打算设计如下网页：

- 首页：展示网站标题【苏轼网】，及苏轼大图一张。
- 生平简介：展示一篇介绍苏轼生平的文章
- 经典诗词：通过表格展示苏轼经典的诗词
- 苏轼图集：展示苏轼相关的图片
- 留言板：提供网友填写留言的表单

2.2 建立网站目录结构

通过 VSCode 建立目录结构如下：



解释下，所有文件都放到项目文件夹 023-html-demo 下，表示第 23 篇博客-HTML 相关-实例的意思。

内有 5 个网页，home 是首页，introduce 是生平简介，message 是留言板，picture 是苏轼图集，poem 是经典诗词。

还有一个 images 文件夹，里面包含 5 张图片，是我从网上下载的苏轼相关图片，当做网站的图片资源，集中放到 images 文件夹下。

此处需要注意的是 images 文件夹和 5 个网页处于同一目录下，即项目文件夹 023-html-demo 下，而 5 张图片是处于 images 文件夹之内的。

2.3 开发首页

首页需要放置一个导航栏，导航栏可以展示所有页面的超级链接，然后需要放置网站标题和首页大图。

整体代码如下：

```
<html>

<head>
  <meta charset="utf-8">
  <title>首页</title>
</head>

<body>
  <!-- 导航栏 -->
  <div>
    <ul>
      <li><a href="home.html">首页</a></li>
      <li><a href="introduce.html">生平简介</a></li>
      <li><a href="poem.html">经典诗词</a></li>
      <li><a href="picture.html">苏轼图集</a></li>
      <li><a href="message.html">留言板</a></li>
    </ul>
  </div>
  <!-- 网站标题 -->
  <h1>苏轼网</h1>
  <!-- 首页大图 -->
  
</body>

</html>
```

效果如下：



2.2 生平简介页

生平简介页面，导航栏跟首页是一样的，其实所有页面的导航栏都相同，因为导航是用于跳转任意网页的。

然后生平简介页面在导航下面还需要放置生平简介文章，所以代码如下：

```
<html>

<head>
  <meta charset="utf-8">
  <title>生平简介</title>
</head>

<body>
  <!-- 导航栏 -->
  <div>
    <ul>
      <li><a href="home.html">首页</a></li>
      <li><a href="introduce.html">生平简介</a></li>
      <li><a href="poem.html">经典诗词</a></li>
      <li><a href="picture.html">苏轼图集</a></li>
      <li><a href="message.html">留言板</a></li>
    </ul>
  </div>
  <!-- 内容 -->
  <p>
    苏轼（1037 年 1 月 8 日—1101 年 8 月 24 日），字子瞻，一字和仲，号铁冠道人、东坡居士，
    世称苏东坡、苏仙 、坡仙
    ，汉族，眉州眉山（今四川省眉山市）人，祖籍河北栾城，北宋文学家、书法家、美食家、画
    家，历史治水名人。
    苏轼是北宋中期文坛领袖，在诗、词、散文、书、画等方面取得很高成就。
  </p>
</body>

</html>
```

效果如下：



2.3 经典诗词页

经典诗词页，上面同样是导航栏，下面通过表格展示一些经典诗词即可。

```
<html>

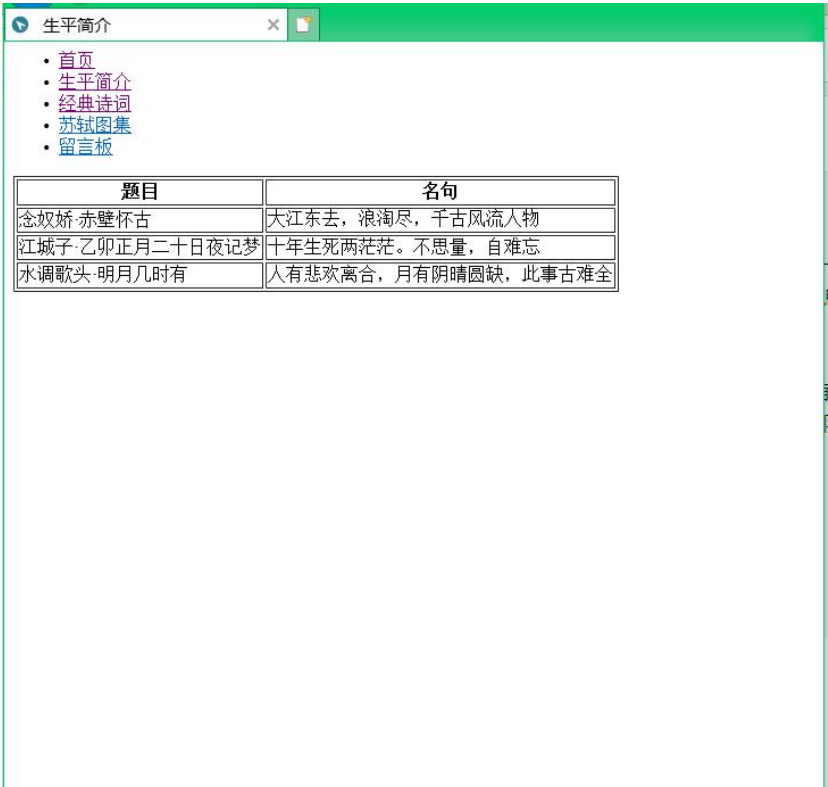
<head>
  <meta charset="utf-8">
  <title>生平简介</title>
</head>

<body>
  <!-- 导航栏 -->

  <div>
    <ul>
```

```
<li><a href="home.html">首页</a></li>
<li><a href="introduce.html">生平简介</a></li>
<li><a href="poem.html">经典诗词</a></li>
<li><a href="picture.html">苏轼图集</a></li>
<li><a href="message.html">留言板</a></li>
</ul>
</div>
<!-- 诗词表格 -->
<table border="1">
  <thead>
    <th>题目</th>
    <th>名句</th>
  </thead>
  <tbody>
    <tr>
      <td>念奴娇·赤壁怀古</td>
      <td>大江东去，浪淘尽，千古风流人物</td>
    </tr>
    <tr>
      <td>江城子·乙卯正月二十日夜记梦</td>
      <td>十年生死两茫茫。不思量，自难忘</td>
    </tr>
    <tr>
      <td>水调歌头·明月几时有</td>
      <td>人有悲欢离合，月有阴晴圆缺，此事古难全</td>
    </tr>
  </tbody>
</table>
</body>
</html>
```

效果如下：



2.4 苏轼图集页

除了导航栏外，添加一些苏轼相关图片即可。

```
<html>

<head>
  <meta charset="utf-8">
  <title>苏轼图集</title>
</head>

<body>
  <!-- 导航栏 -->
  <div>
```

```
<ul>
  <li><a href="home.html">首页</a></li>
  <li><a href="introduce.html">生平简介</a></li>
  <li><a href="poem.html">经典诗词</a></li>
  <li><a href="picture.html">苏轼图集</a></li>
  <li><a href="message.html">留言板</a></li>
</ul>
</div>
<!-- 图集 -->
<br>
<br>
<br>
<br>
</body>

</html>
```

效果如下:

- [首页](#)
- [生平简介](#)
- [经典诗词](#)
- [苏轼图集](#)
- [留言板](#)



2.5 留言板

除了导航栏之外，我们可以通过表单制作一个留言板。当然由于我们目前只学习了 HTML，所以留言板只能显示一个界面，并不能真正将留言提交给我们的后台，这个留待以后完善。

留言板代码如下：

```
<html>

<head>
    <meta charset="utf-8">
    <title>留言板</title>
</head>

<body>
    <!-- 导航栏 -->
    <div>
        <ul>
            <li><a href="home.html">首页</a></li>
            <li><a href="introduce.html">生平简介</a></li>
            <li><a href="poem.html">经典诗词</a></li>
            <li><a href="picture.html">苏轼图集</a></li>
            <li><a href="message.html">留言板</a></li>
        </ul>
    </div>
    <!-- 图集 -->
    <form action="#" method="get">
        请输入姓名：<input type="text" name="username"><br>
        请输入留言：<input type="text" name="content"><br>
        <input type="submit" value="提交留言">
    </form>
```

```
</body>
```

```
</html>
```

效果如下：



3. 小结

通过 23 篇文章的学习，并且编写了一个综合实例，如果认真学习的话，现在想必对 HTML 网页世界有了一个完全不一样的认识。

实际上这 23 篇文章的内容并不多，1 天学习两三篇也是很正常的事情，1 天学习 1 篇也不算慢。难的是持之以恒，凡事积小成多，努力努力再努力，直到量变引起质变。

哪有什么天赋异禀，谁的本事不是刻苦学来的。