

Web 前端开发 实战教程

HTML5+CSS3+JavaScript | 微课版

Web Front-end Development



蜗牛学院 邓强 主编

- ◆ 作者有十余年的软件开发经验和项目管理经验，知识讲解鞭辟入里
- ◆ 全书重点难点配有微课视频，书网结合，帮助读者融会贯通
- ◆ 利用真实网站开发案例贯穿全书，帮助读者了解网站开发真实过程

官方微信平台



384053806

QQ群



Web 前端开发 实战教程

HTML5+CSS3+JavaScript | 微课版

Web Front-end Development



蜗牛学院 邓强 主编

人民邮电出版社

北京

图书在版编目 (C I P) 数据

Web前端开发实战教程：HTML5+CSS3+JavaScript：
微课版 / 蜗牛学院，邓强主编. — 北京：人民邮电出
版社，2017.9

(互联网+职业技能系列)

ISBN 978-7-115-46664-8

I. ①W… II. ①蜗… ②邓… III. ①超文本标记语言
—程序设计—教材②JAVA语言—程序设计—教材 IV.
①TP312

中国版本图书馆CIP数据核字(2017)第196346号

内 容 提 要

本书共分 11 章，其中第 1~2 章主要讲解了 HTML 标签及属性的应用，结合两个实战项目帮助读者更好地理解相关知识点。第 3~6 章主要讲解 CSS 的应用，通过引入 4 个项目实战并讲解开发思路，对各知识点的应用进行深入分析，同时也提供完整的代码实现。第 7~11 章则讲解 JavaScript 程序设计，包括 JavaScript 的基础语法、JavaScript 内置对象的应用、JavaScript 对 DOM 和 BOM 的操作，以及如何响应用户的鼠标和键盘操作等，均附有大量的实例应用代码，结合 5 个项目实战案例，帮助读者深入理解程序设计的原理与应用。同时，本书完全利用原生的 JavaScript 来实现常见的 Web 页面功能，不借助于任何第三方框架，也是希望能够帮助读者从原理上对 Web 页面的功能实现有所理解，进而帮助大家打下扎实的基本功。

本书可以作为高校计算机相关专业的教学用书，也可以作为 Web 前端开发爱好者的实战宝典。书中利用大量实例和项目实战对最核心的知识点进行了深入剖析，可以更加有效地帮助读者提升 Web 前端开发的能力。

◆ 主 编 蜗牛学院 邓 强

责任编辑 左仲海

责任印制 马振武

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京鑫正大印刷有限公司印刷

◆ 开本：787×1092 1/16

印张：16.75

2017 年 9 月第 1 版

字数：439 千字

2017 年 9 月北京第 1 次印刷

定价：49.80 元

读者服务热线：(010)81055256 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

前言

Foreword

随着 Web 前端开发技术的快速发展，一方面归功于移动互联网的大规模普及及厂商对用户体验的高度关注；另一方面，由于移动端操作系统主要集中于 iOS 和 Android，而每一家移动互联网公司都必须要为其 App 产品开发两套系统，去适配数百款移动终端，研发和维护成本居高不下。这些前提都让基于移动互联网的 HTML5 技术得以快速发展。由近年的趋势基本可以判断，将会有越来越多的厂商把前端开发的重心移向 Web。

在这样一个时代背景下，越来越多的从业者将加入 Web 前端开发工程师的队伍当中。同时，随着 HTML、CSS 和 JavaScript 等技术规范的进一步升级及浏览器厂商的大力支持，Web 浏览器扮演的角色将会越来越重要，甚至不可或缺。所以，写一本书普及一下 Web 前端开发最核心的技术，对于作者来说是一件非常有价值、有意义的事。

那么，通过怎样的写作方式，能够更好地传播这些技术呢？作者通过 16 年实际研发经验及近 10 年的讲师经验得到一些启发。要有效地传播知识，最好的方式并非眉飞色舞地演讲，而是让大家实战，进而总结思路，优化思路，进一步突破。所以本书在写作之初，就定了一个基本前提——“全程实战”，一切知识点的讲解和思路的梳理，都是为书中的实战案例做准备的。软件开发本来就是一门实战出真理的手艺，过多的口水战显得毫无意义，当然，单纯抄写代码，没有形成一套有效解决问题的思路和方法，那也注定只是一些假把式，离高手之路只能越来越远。

作为“蜗牛学院”的 Web 前端开发工程师训练营的核心教材，本书通过知识点讲解、实例讲解、项目实战三大步骤，可以有效地帮助读者把 Web 前端开发的三大核心技术 HTML、CSS 和 JavaScript 进行系统性的梳理，并且将其纳入真实的项目场景中，帮助读者更好地理解和应用这些技术。其中，HTML 部分主要讲解标准的标签及属性的应用及在排版内容上的优化，CSS 部分重点讲解目前通用的 Web 页面样式设置方式，通过将 HTML 和 CSS 进行有效的结合，可以制作出一个专业的 Web 页面甚至开发一个站点。但是 HTML 和 CSS 主要用于内容的美化与展现，还无法做到与用户进行有效的交互，这便是 JavaScript 之所长。

本书可以作为高校计算机相关专业学生的教学用书，也可以作为 Web 前端开发爱好者的实战宝典。如果作为高校教材，建议授课时间在 64 课时以上，且优先考虑在机房进行授课。如果是 Web 前端开发爱好者使用，也同样建议将书中的每一个练习和项目都完整地做完一遍，甚至两遍。这样才能具备了一个 Web 前端开发工程师的核心能力，剩下的只是解决更加具体的业务和应用场景而已。

在本书的写作过程中，我的同事和家人给予了很大的理解和支持。在此对我的同事李懿、陈南、陈华、胡平等表示真诚的感谢。最后，非常感谢蜗牛学院的学员们，是我们无数个日夜的教与学以及师生之间的大量讨论，成就了本书案例和思路的成型。

读者可以通过蜗牛学院的在线课堂（网址为：<http://www.woniuxy.com>）进行学习，下载配套视频和源代码。相应资源也可登录人民邮电出版社教育社区进行下载。如果需要与作者进行技术交流或商务合作，可添加微信或 QQ15903523，或添加 QQ 群 645859048，或访问作者原创学习网站：<http://www.bossqiang.com> 均可，当然，也可以直接发送邮件至 dengqiang@woniuxy.com。

由于作者经验和水平有限，书中疏漏之处在所难免，欢迎读者朋友们批评指正。

编 者

2017 年 5 月

目录

Contents

第1章 HTML 核心基础 1

1.1 了解 Web 系统	2
1.1.1 网络体系结构	2
1.1.2 B/S 架构的特点	2
1.1.3 页面渲染引擎	3
1.2 HTML 开发基础	3
1.2.1 HTML 简介	3
1.2.2 第一个 HTML 页面	5
1.2.3 WebStorm 开发工具	6
1.2.4 其他开发工具	11
1.3 HTML 常用标签	11
1.3.1 文本	11
1.3.2 超链接	12
1.3.3 图像	14
1.3.4 表格	17
1.3.5 表元素	22
1.3.6 列表	23
1.3.7 滚动字幕	24
1.3.8 音频/视频播放	26
1.3.9 其他标签	28

第2章 HTML 项目实战 31

2.1 在线计算器	32
2.1.1 项目介绍	32
2.1.2 开发思路	32
2.1.3 代码实现	33
2.2 蜗牛学院官网	35
2.2.1 项目介绍	35
2.2.2 开发思路	35
2.2.3 代码实现	36

第3章 CSS 核心基础 40

3.1 CSS 基础	41
------------	----

3.1.1 CSS 简介	41
3.1.2 CSS 的特点	41
3.1.3 CSS 的使用	42
3.2 CSS 选择器	43
3.2.1 标签选择器	43
3.2.2 ID 选择器	45
3.2.3 Class 选择器	47
3.2.4 组合选择器	49
3.2.5 伪类选择器	50
3.3 CSS 元素样式	52
3.3.1 文本与图像	52
3.3.2 表格	54
3.3.3 超链接	56
3.3.4 列表	57
3.3.5 表单	58

第4章 盒模型 60

4.1 盒模型基础知识	61
4.1.1 盒模型简介	61
4.1.2 盒模型属性	61
4.1.3 盒模型基础使用	62
4.2 盒模型浮动	64
4.2.1 浮动的作用	64
4.2.2 盒模型左浮动	65
4.2.3 盒模型右浮动	66
4.2.4 禁止浮动	67
4.3 盒模型内容	67
4.3.1 宽度与高度	67
4.3.2 水平居中	69
4.3.3 垂直居中	71
4.3.4 内容溢出	73
4.4 盒模型嵌套	76
4.4.1 嵌套的作用	76
4.4.2 嵌套的排版	76

4.5 盒模型定位	78	7.2 基础编程练习	144
4.5.1 定位简介	78	7.2.1 练习题目	144
4.5.2 固定定位	79	7.2.2 循环求和	145
4.5.3 绝对定位	82	7.2.3 字符判断	145
4.5.4 相对定位	83	7.2.4 密码验证	147
4.5.5 其他属性	84	7.2.5 水仙花数	148
第 5 章 布局项目实战	85	7.2.6 次方计算	149
5.1 九宫格布局	86	7.2.7 统计字符	150
5.1.1 项目介绍	86	7.2.8 抓球问题	150
5.1.2 开发思路	86	7.2.9 九九乘法表	151
5.1.3 代码实现	87	7.2.10 字符串判断	152
5.2 在线计算器布局	88	7.3 数组	154
5.2.1 项目介绍	88	7.3.1 定义与使用	154
5.2.2 开发思路	89	7.3.2 核心算法	155
5.2.3 代码实现	89	7.3.3 排序算法	156
5.3 蜗牛学院官网布局	92	7.3.4 多维数组	157
5.3.1 项目介绍	92	7.3.5 数组对象	158
5.3.2 开发思路	92	7.3.6 数组练习	159
5.3.3 代码实现	94	7.4 函数	163
5.3.4 代码优化	105	7.4.1 函数定义	163
第 6 章 CSS 高级应用	106	7.4.2 可选参数	165
6.1 动画效果	107	7.4.3 匿名函数	165
6.1.1 animation	107	7.4.4 函数应用	166
6.1.2 transform	113	7.5 字符串	172
6.1.3 transition	119	7.5.1 字符串的属性	172
6.2 响应式布局	121	7.5.2 字符串的方法	172
6.2.1 响应式布局简介	121	7.5.3 字符串的应用	173
6.2.2 媒体查询	122	第 8 章 文档对象模型	177
6.3 项目实战	126	8.1 Document 对象	178
6.3.1 实现一个立方体	126	8.1.1 对象集合	178
6.3.2 实现立方体运动	128	8.1.2 对象属性	180
第 7 章 JavaScript 程序设计	131	8.1.3 对象方法	181
7.1 语法基础	132	8.2 查找 DOM 元素	182
7.1.1 JavaScript 简介	132	8.2.1 DOM 简介	182
7.1.2 输入与输出	133	8.2.2 通过 ID 查找	183
7.1.3 变量与数据类型	135	8.2.3 通过标签查找	185
7.1.4 分支语句	138	8.2.4 通过 Class 查找	186
7.1.5 循环语句	142	8.2.5 通过 Name 查找	188
		8.2.6 DOM 操作练习	188
		8.3 操作 DOM 元素	192
		8.3.1 DOM 的属性与方法	192

8.3.2	DOM 的新增	194	11.1.1	项目介绍	231
8.3.3	DOM 的删除	195	11.1.2	开发思路	231
8.3.4	DOM 的修改	196	11.1.3	代码实现	233
8.3.5	针对表格的操作	197	11.1.4	思维拓展	236
第 9 章 JavaScript 对象		204	11.2	在线计算器	237
9.1	Window 对象	205	11.2.1	项目介绍	237
9.1.1	BOM 简介	205	11.2.2	开发思路	238
9.1.2	窗口对象	206	11.2.3	代码实现	239
9.1.3	弹出窗口	208	11.2.4	思维拓展	241
9.1.4	弹出框对象	209	11.3	在线时钟	242
9.1.5	定时器对象	209	11.3.1	项目介绍	242
9.1.6	其他对象	210	11.3.2	开发思路	243
9.2	异常处理机制	211	11.3.3	代码实现	243
9.2.1	异常处理机制简介	211	11.4	倒计时程序	245
9.2.2	使用 onerror 处理异常	211	11.4.1	项目介绍	245
9.2.3	使用 try...catch 处理异常	212	11.4.2	开发思路	246
9.3	正则表达式	213	11.4.3	代码实现	246
9.3.1	正则表达式简介	213	11.4.4	思维拓展	251
9.3.2	正则表达式语法	213	11.5	图片轮播	252
9.3.3	RegExp 对象	215	11.5.1	项目介绍	252
第 10 章 事件处理		217	11.5.2	开发思路	252
10.1	鼠标事件	218	11.5.3	代码实现	253
10.1.1	鼠标事件列表	218	11.5.4	思维拓展	258
10.1.2	鼠标单击实例	218			
10.1.3	鼠标悬停实例	220			
10.2	键盘事件	221			
10.2.1	键盘事件及属性	221			
10.2.2	检测按键实例	222			
10.2.3	只接受数字输入	224			
10.3	表单事件	224			
10.3.1	表单事件简述	224			
10.3.2	密码对比实例	225			
10.3.3	动态城市联动	226			
10.4	多媒体事件	227			
10.4.1	多媒体事件列表	227			
10.4.2	记录播放时间	228			
第 11 章 JavaScript 项目实战		230			
11.1	随机飘雪	231			

第1章

HTML核心基础

本章导读：

■ 本章主要介绍 Web 页面中，HTML4 和 HTML5 里面最常用的标签及常用属性，即构成一个基本网页的核心要素，以及开发 Web 页面的一些常识。

■ 作为重要媒介的 HTML 标准，目前已经发展到 HTML5 版本。与之对应的，浏览器也在不停地进步，可以说，Web 页面的功能之所以越来越强大，离不开各个浏览器厂商的大力支持。同时，随着移动互联网的飞速发展，HTML5 也更加具备实用价值，在增强用户体验方面以及优化前端交互方面，起到了至关重要的作用。

学习目标：

- (1) 充分理解HTML页面的基本结构。
- (2) 充分理解HTML常用标签及其属性的应用。
- (3) 熟练运用HTML常用标签完成网页的布局及美化。
- (4) 熟练运用WebStorm前端开发工具开发HTML页面。



V1-1 网络体系与
BS 架构

1.1 了解 Web 系统

1.1.1 网络体系结构

在当今互联网如此发达的时代，其背后的系统架构无非以下 3 种，这三种架构各有其不同的适用场景，本书将重点探讨 B/S 架构。

(1) B/S (Browser/Server) 架构：典型应用包括如谷歌、百度这种搜索引擎，或者是 Taobao、eBay 这种电子商务网站，或者是新浪、雅虎这种门户网站，又或是 ITPub、CSDN 这种论坛等，各类应用数不胜数。但是无论是哪种应用，都是通过网页浏览器进行访问，通过浏览器与服务器进行通信来完成的，所以这一类系统统称为 B/S 架构的系统。

(2) C/S (Client/Server) 架构：典型应用包括如 QQ，MSN 这类即时通信工具，或者魔兽世界、传奇等这类大型网络游戏，或者是 Outlook、Foxmail 这类邮件客户端等。在移动设备上，也有很多 C/S 架构的应用程序，比如智能手机里安装的各类新闻阅读器、天气查询软件、在线视频播放等软件。C/S 架构的系统都有一个共同的特点，那就是客户端是定制的，是为完成各类功能和与服务器通信而专门开发的。针对不同的应用，有不同的客户端，没有统一的标准和规范。

(3) P2P (Point-to-Point) 点对点系统：这类系统的典型代表有局域网聊天工具飞秋、BT 下载软件等。这类系统的特点是不需要服务器中转，客户端与客户端之间彼此直接通信。



V1-2 浏览器工作
过程

1.1.2 B/S 架构的特点

事实上，B/S 架构的浏览器本身就扮演着一个 Client 的角色，所以完全可以将 B/S 和 C/S 统称为 C/S 架构，这完全没有任何问题。只不过 Browser 这个 Client 与传统意义上的 C/S 架构中的 Client 是有区别的，B/S 可以看作是对 C/S 架构的一种改进，最主要的区别表现在如下几个方面。

(1) B/S 架构的浏览器是规范的、标准的。其核心引擎由几大软件厂商提供，如微软、谷歌、火狐等，并且都支持 W3C (万维网协会) 制定的各类 Web 标准，如 JavaScript、HTML、CSS 和标准的 HTTP 协议。所以，使用 B/S 架构的应用程序可以轻易实现 Any Time、Any Where、Any One 的访问方式，只需要输入一个正常的 URL 地址即可，非常灵活。

(2) B/S 架构的浏览器部署更方便。所有操作系统都内置标准浏览器，它们大同小异，对不同系统的兼容性非常强 (因为浏览器只要能正常解析 HTML 标签，处理 HTTP 协议数据包即可)。如果系统需要升级，只需要对服务器端进行升级即可，客户端不需要做任何修改，因为浏览器访问服务器时会自动获取服务器最新的内容 (使用客户端缓存除外)，所以对于 B/S 架构的系统部署起来是非常方便快捷的。

(3) 在系统的设计与开发方面，B/S 也有优势。如果使用 B/S 架构，可以花更多的精力来关注业务逻辑，客户端的处理由浏览器完成，服务器端的处理由标准 Web 服务器 (如 Apache、IIS、Tomcat 一类) 来完成。

(4) 在系统性能方面，B/S 架构的优势不再明显。采用 B/S 结构的客户端只能完成浏览、查询、数据输入等简单功能，绝大部分处理工作由服务器承担，这使得服务器的负担很重。当然，目前的云计算平台可以很好地处理服务器负担重的问题。

总而言之，B/S 架构的优点很多，目前已经得到广泛运用。而且浏览器的功能越来越强，已经可以用于完成很多复杂的处理，用户可以通过体验得到极大的提升。

1.1.3 页面渲染引擎

网页浏览器的页面渲染引擎负责取得网页的内容、整理排版以及计算网页的显示方式,然后输出至显示器或打印机。所有网页浏览器、电子邮件客户端以及其他需要编辑、显示网络内容的应用程序都需要页面渲染引擎,当前比较流行的页面渲染引擎有如下几种。



V1-3 页面渲染过程

1. Trident 页面渲染引擎

Trident 是微软视窗操作系统 (Windows) 搭载的网页浏览器——Internet Explorer 的页面渲染引擎的名称,它的第一个版本诞生于 1997 年 10 月发布的 Internet Explorer 第四版中,目前是互联网上非常流行的排版引擎。目前使用 Trident 渲染引擎的浏览器有 Internet Explorer、360 安全浏览器等。

2. Gecko 页面渲染引擎

Gecko 是开放源代码的、以 C++ 编写的页面渲染引擎。Gecko 是跨平台的,能在 Windows、Linux 和 Mac OS X 等主要操作系统上运行。使用 Gecko 页面渲染引擎的浏览器有 Firefox、Mozilla 等。

3. KHTML 页面渲染引擎或 WebKit 框架

KHTML 拥有速度快捷的优点,但对错误语法的容忍度则比 Firefox 产品所使用的 Gecko 引擎小。苹果电脑于 2002 年采用了 KHTML,作为开发 Safari 浏览器之用。WebCore 及 WebKit 引擎均是 KHTML 的衍生产品,目前使用 KHTML 页面渲染引擎的浏览器有 Safari、Konqueror、Google Chrome 等。

4. Presto 页面渲染引擎

Presto 是一个由 Opera Software 开发的浏览器页面渲染引擎,应用于 Opera 浏览器。

可见,浏览器的页面渲染引擎丰富繁杂,这对 Web 前端开发提出了很大的挑战,特别是兼容性以及稳定性方面。用户不单要熟悉每种渲染引擎和相应浏览器的特性,还需要了解它们之间的各种差异,便于设计出适合的、兼容性好的、稳定性高的系统。

1.2 HTML 开发基础



V1-4 HTML 简介

1.2.1 HTML 简介

1. 什么是 HTML

HTML 即“超文本标记语言”,其英文全称为 Hyper Text Markup Language,是用来描述网页的一种语言规范。通过 HTML 的全称可以看出,HTML 不是一种编程语言,而是一种标记语言,由诸多不同的标签来完成网页内容的描述。

超文本标记语言是标准通用标记语言下的一个应用,也是一种规范,一种标准。它通过标记符号来标记想要显示的网页中的各个部分。网页文件本身是一种文本文件,通过在文本文件中添加标记符,可以告诉浏览器如何显示其中的内容(如文字如何处理、画面如何安排、图片如何显示等)。浏览器按顺序阅读网页文件,然后根据标记符解释和显示其标记的内容,对书写出错的标记将不指出其错误,且不停止其解释执行过程,编制者只能通过显示效果来分析出错原因和出错部位。但需要注意的是,不同的浏览器,对同一标记符可能会有不完全相同的解释,因而可能会有不同的显示效果。

2. HTML 的特点

HTML 文档制作不是很复杂,但功能强大,支持不同数据格式的文件嵌入,这也是万维网(WWW)盛行的原因之一,其主要特点如下。

(1) 简易性: HTML 版本升级采用超集方式,从而更加灵活方便。

(2) 可扩展性: HTML 的广泛应用带来了加强功能、增加标识符等要求, HTML 采取子类元素的方式为系统扩展带来保证。

(3) 平台无关性: 虽然个人计算机大行其道, 但使用 Mac 等其他机器的也大有人在, HTML 可以使用在广泛的平台上, 这也是万维网盛行的另一个原因。

(4) 通用性: 另外, HTML 是网络的通用语言, 是一种简单、通用的全置标记语言。它允许网页制作者建立文本与图片相结合的复杂页面, 这些页面可以被网上任何其他他人浏览到, 而且无论使用的是哪种类型的电脑或浏览器。

3. HTML 标签

HTML 标签 (HTML Tag) 即 HTML 标记标签, 是由尖括号包围的关键词, 比如 <html>。HTML 标签通常成对出现, 比如 和 , 其中的第一个标签是开始标签 (也称为开放标签), 第二个标签是结束标签 (也称为闭合标签)。

<标签>内容</标签>

4. HTML 版本

从初期的网络诞生后, 已经出现了许多 HTML 版本, 详情如表 1-1 所示。目前最为流行的, 特别是对移动互联网支持最好的是 HTML5 版本。

表 1-1 HTML 版本历史

版本	发布时间	备注
HTML	1991	初始版本, 非标准
HTML+	1993	作为互联网工程工作小组 (IETF) 工作草案发布
HTML 2.0	1995	作为 RFC 1866 发布
HTML 3.2	1997	W3C 推荐标准
HTML 4.01	1999	W3C 推荐标准, 目前的 PC 端网页规范
XHTML 1.0	2000	W3C 推荐标准, 后来经过修订于 2002 年 8 月 1 日重新发布
HTML5	2012	W3C 推荐标准, 基于移动终端进行优化
XHTML5	2013	从 XHTML 1.x 的更新版, 基于 HTML 5 草案

其中, XHTML 是指严格按照开始标签与结束标签的方式来书写的 HTML 页面, 并且符合 XML 规范。关于 XML (可扩展标记语言) 的细节本书不作详细介绍, 需要了解的读者可自行搜索。

HTML5 的设计目的是在移动设备上支持多媒体, 所以新的语法特征被引进, 如 Video、Audio 和 Canvas 标记。HTML5 还引进了如下所述新的功能, 可以真正改变用户与文档的交互方式。

(1) 优化: 新元素, 新属性, 完全支持 CSS3, 对网页动画支持更好; 同时, 对 HTML4.0 一些较为过时的标记和属性进行了淘汰。

(2) 新增: Video 视频播放, Audio 音频播放, 2D/3D 制图, 本地存储, 本地 SQL 数据。

(3) 增强: Web 应用体验更好, 充分支持移动设备; 引入新的 JavaScript 脚本引擎, 对 AJAX 支持更好; 新的解析规则更加灵活, 更加明确。

支持 HTML5 的浏览器包括 Firefox (火狐浏览器)、IE9 及其更高版本、Chrome (谷歌浏览器)、Safari、Opera 等; 傲游浏览器 (Maxthon) 以及基于 IE 或 Chromium (Chrome 的工程版或称实验版) 所推出的 360 浏览器、搜狗浏览器、QQ 浏览器、猎豹浏览器等国产浏览器同样具备支持 HTML5 的能力, 如图 1-1 所示。



图 1-1 HTML5 浏览器支持

V1-5 第一个
HTML 页面-1

1.2.2 第一个 HTML 页面

HTML 页面是由纯文本构成的，可以使用记事本来编辑一个 HTML 页面。下面介绍一个最基本的 HTML 页面如何实现，如图 1-2 所示。

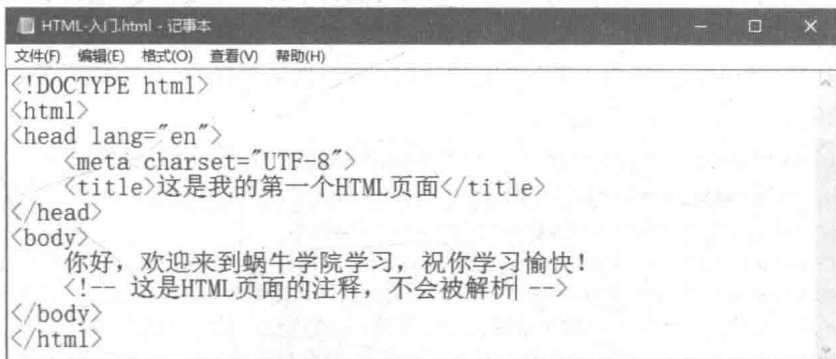


图 1-2 第一个 HTML 页面

先打开记事本，将如上文内容输入，然后保存为后缀名为.html 的文件，这样，HTML 页面即开发完成。双击这个 HTML 文件，用默认的浏览器打开，即可看到运行结果如图 1-3 所示，一个最简单且结构完整的 HTML 网页就此完成。

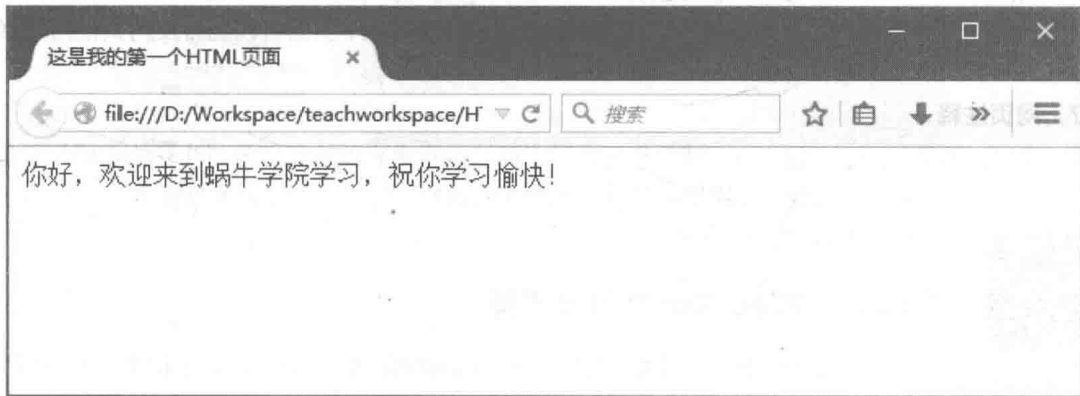


图 1-3 HTML 页面的显示效果

通过本章内容的学习，能够让读者尽快找到学习的乐趣，同时开发出比较专业的网站，要达到这样的效果，有必要了解上述代码代表的意思。

1. <!DOCTYPE> 声明

<!DOCTYPE>声明有助于浏览器正确显示网页。网络上有很多不同的文件，只有正确声明 HTML 的版本，浏览器才能正确显示网页内容。DOCTYPE 声明是不区分大小写的以下书写方式均可。

```
<!DOCTYPE html>
<!DOCTYPE HTML>
<!doctype html>
<!Doctype Html>
```

2. <html>标签

标签 `<html>` 与 `</html>` 表明了文档类型, 表明这是一个按照 HTML 规范来书写的 HTML 文档。

3. <head>标签

标签<head>与</head>之间的内容主要用于描述页面的头部信息，包括但不限于页面设置、标题、CSS 样式属性、JavaScript 代码等。

4. <meta>标签

<meta>标签主要用于设置页面的基础元信息，比如此处设置页面的编码格式为 UTF-8（即全球统一文本编码），如果不明确指定编码格式，浏览器很有可能将中文处理为乱码。除了设置为 UTF-8 编码外，也可以设置为 GB2312、GBK、GB18030 等中文国标编码，这三者都对中文有很好的支持。但是通常现在的网站都比较国际化，设置为 UTF-8 更加通用，兼容性也更好。<meta> 标签除了可以设置编码格式外，还可以设置更多其他属性，具体如下。

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta http-equiv="cache-control" content="no-cache" />
<meta name="viewport" content="width=device-width,
initial-scale=0,maximum-scale=1,user-scalable=yes">
<meta name="description" content="蜗牛学院-移动互联网人才孵化基地" />
<meta name="keywords" content="蜗牛学院,Java开发,在线课堂,软件测试" />
<meta name="author" content="蜗牛学院是成都顶尖的IT培训及研发机构" />
```



V1-6 第一个
HTML 页面-2

5. <title>标签

<title>标签只用于设置页面的标题。

6. <body>标签

<body>标签应该说是网页中最为重要的标签，主要用于展示网页的内容，理论上来说，网页的全部内容都必须包含在<body>标签中。<body>标签的作用相当于浏览器的内容展示区域，算是一个最大的内容容器，里面可以包含任何其他用于展示网页内容的标签。

7. 网页注释

在 HTML 页面中，`<!--`和`-->`是特殊标记，用于标注网页内容的注释部分。注释的主要作用是对代码进行解释。注释不会被浏览器解析和执行，主要是给开发人员做参考，所以注释部分的内容不会显示在浏览器中。



V1-7 WebStorm
开发环境

1.2.3 WebStorm 开发工具

通过上一节的学习可以发现，使用记事本编辑 HTML 其实是非常消耗时间的，必须要一个一个字符输入。本节将介绍一款非常专业的网页开发工具——WebStorm。

1. 下载安装

访问 WebStorm 官方网站 <http://www.jetbrains.com/webstorm/> 可以获取最新版本的下载安装包, 整个下载及安装过程较为简单, 本书不再赘述。

2. WebStorm 主界面

WebStorm 开发工具软件主界面, 如图 1-4 所示。

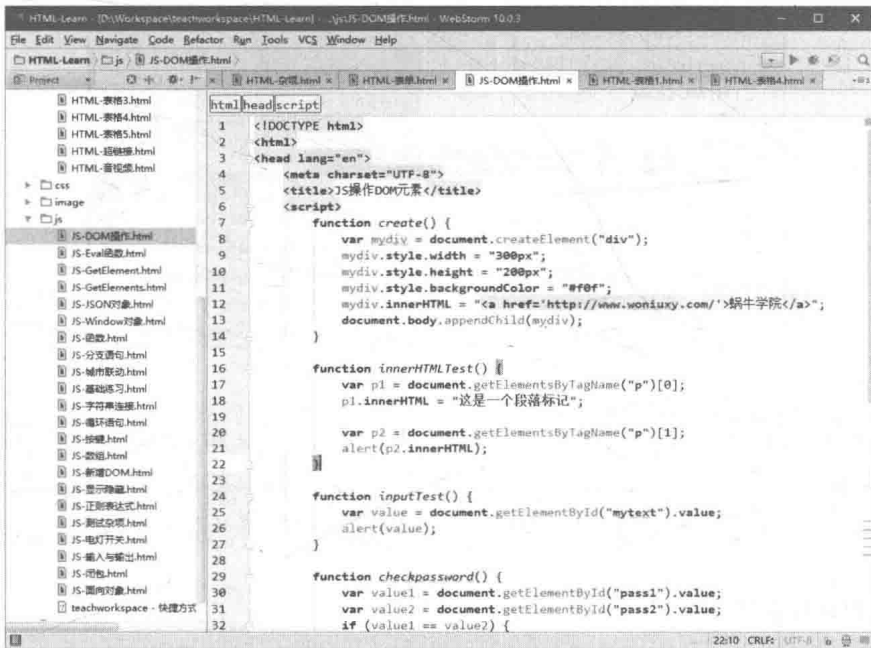


图 1-4 WebStorm 主界面

WebStorm 主界面主要分为三大区域, 顶部为菜单栏, 左边为项目结构, 右边为代码窗口。其基本操作与绝大部分 IDE 工具是一致的, 此处不再赘述。

3. 创建一个 HTML 项目

选择 “File” → “New Project” 命令, 打开新建项目设置对话框, 输入项目的名称及路径即可完成项目创建, 如图 1-5 所示。

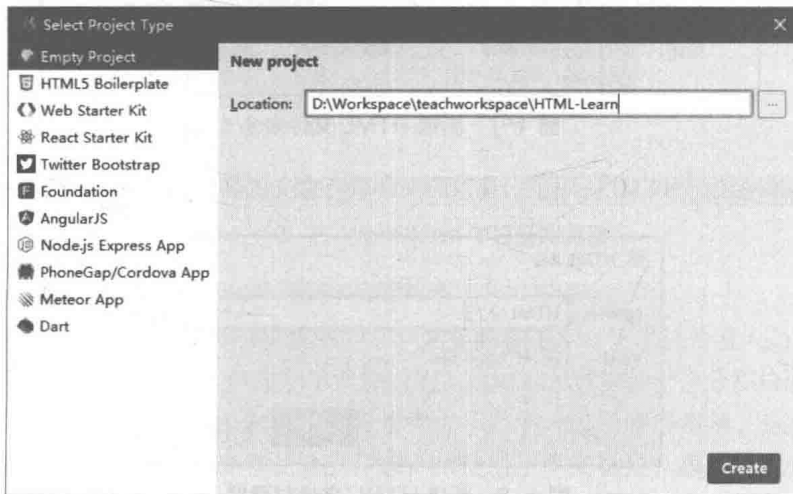


图 1-5 新建 WebStorm 项目

需要注意的是，这里选择创建一个 Empty Project (空项目) 即可，文本框中的 HTML-Learn 是项目名称。

4. 创建项目目录结构

在项目根目录下继续再创建几个目录用于后续开发使用。右击项目名称，选择 “New” → “Directory” 命令，输入目录名称即可。

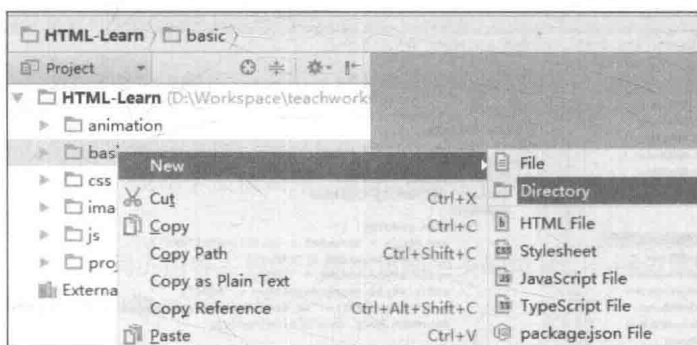


图 1-6 新建项目目录

如图 1-6 所示，这里创建了 animation、basic、css、image、js 和 project 六个目录，后续设计开发过程中会陆续使用到这些目录。创建项目目录时，名称可以使用中文或英文，但是建议使用英文，且使用标准的英文单词或缩写，不建议使用全拼或中文或其他无任何意义的字母或单词。这个习惯在程序设计过程中具有良好的便利性。

5. 创建 HTML 文件

项目目录创建完成后，在 basic 目录下创建一个 html 源文件，如图 1-7 所示。

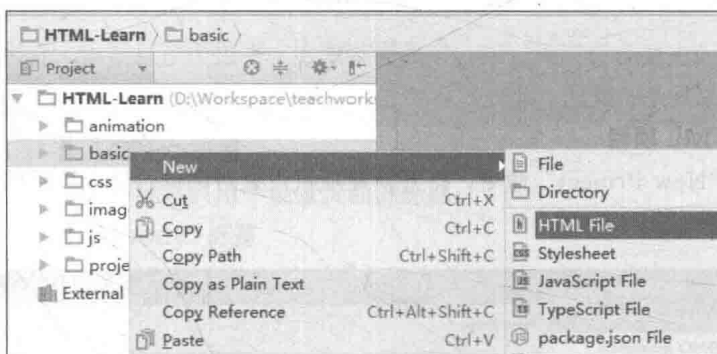


图 1-7 新建 HTML 文件命令

设置 HTML 文件名为 “HTML-入门”，建议给文件取有意义的名称，以便于后续查阅，如图 1-8 所示。

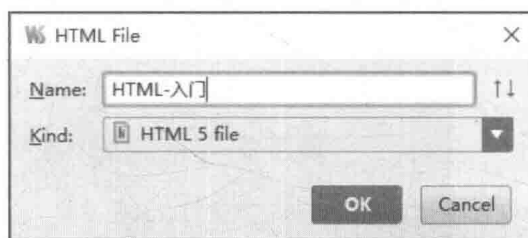


图 1-8 新建 HTML 文件对话框

6. 快速完成内容填充

创建完成一个 HTML5 文件后, WebStorm 开发工具已经完成了很多工作, 把页面的基本结构已经自动生成, 开发人员只需要填充网页内容和标题即可, 如图 1-9 所示。

```
1  <!DOCTYPE html>
2  <html>
3  <head lang="en">
4      <meta charset="UTF-8">
5      <title>这是我的第一个HTML页面</title>
6  </head>
7  <body>
8      你好, 欢迎来到蜗牛学院学习, 祝你学习愉快!
9      <!-- 这是HTML页面的注释, 不会被解析 -->
10 </body>
11 </html>
```

图 1-9 WebStorm 源代码窗口

7. 运行 HTML 页面

即使是使用 WebStorm 开发的网页, 同样可以利用资源管理器打开项目所在目录, 双击该 HTML-入门.html 文件用浏览器打开运行即可。

另外, WebStorm 提供了非常方便的运行方法, 在打开的网页编辑器中, 当鼠标指针移动到右上角时, 会出现几个浏览器图标, 单击任意一个即可用相应的浏览器打开该页面 (前提是操作系统中已经安装该浏览器), 如图 1-10 所示。

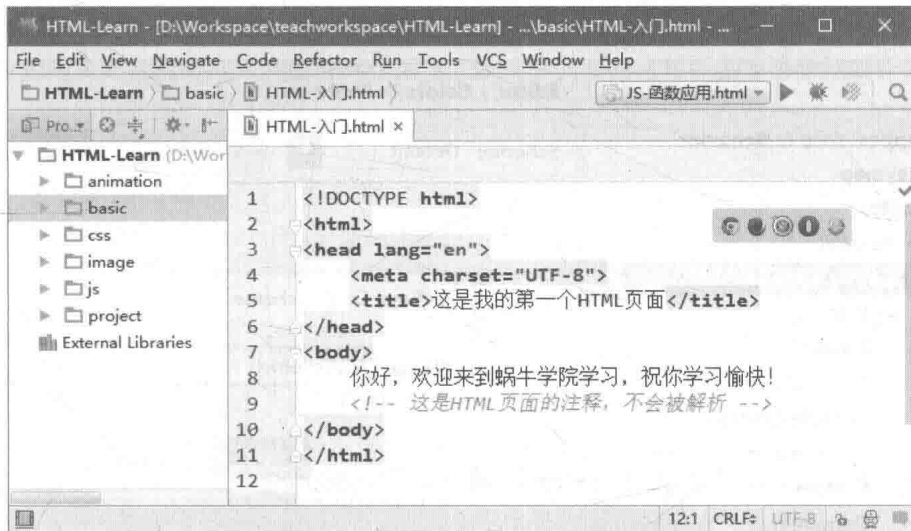


图 1-10 在 WebStorm 中打开浏览器

8. 修改编辑区格式

利用 WebStorm 工具可以显著提升开发效率, 它自带智能提示, 可以让开发人员的输入效率大幅提高; 同时, 用不同的颜色帮助开发人员显示不同类型的文本内容, 更加便于区分和快速查阅。

但是, WebStorm 工具默认的编辑器字体设置得太小, 不太容易看得清楚, 这样很容易导致一些错误, 比如应该使用英文半角的双引号、单引号等, 却使用了中文全角, 很难发现这种类似的失误, 但是这样会导致网页无法正常解析。所以开发过程中需要将编辑器的字体大小调整得更加容易识别,