1. **Web前端开发技术综述**

**学习重点**

了解什么是Web前端

学会区别理解前、后端技术

熟悉前端开发的三种语言

学会使用开发工具和调试工具

**知识结构图**

**本章介绍**

Web的存在已经有20多年了，随着网络不断的发展，网络不断的影响着智能手机、平板电脑等；懂得如何做好网页设计与制作的程序员迎来了绝佳的机会。从本章开始，将会带你走进Web前端开发的世界。打开这个世界的第一道门将会包括Web的概述、前后端技术、Web前端开发所用的知识、开发工具、调试工具等，将会给你带来一个全新的体验。

**1.1 Web概述**

Web是（World Wide Web，WWW）的简称，即全球广域网，也称为万维网，它是一种基于超文本和HTTP的、全球性的、动态交互的、跨平台的分布式图形信息系统，是建立在Internet上的一种网络服务，为浏览者在Internet上查找和浏览信息提供了图形化的、易于访问的直观界面，其中的文档及超级链接将Internet上的信息节点组织成一个互为关联的网状结构。

随着社会不断发展，网络技术日新月异，国内外信息化建设已经到了以Web应用为基础核心的阶段，越来越多的企业选择以Web来建立其应用系统，增强企业的管理，加大企业的影响力；Web开发的市场需求大，Web的开发也越来越复杂，这既是一个机遇，也是一个挑战。

**1.1.1 Web的起源**

Web这个Internet上最热门的应用架构是由Tim Berners-Lee发明的。1989年CERN（欧洲粒子物理研究所）中由Tim Berners-Lee领导的小组提交了一个针对Internet的新协议和一个使用该协议的文档系统，该小组将这个新系统命名为Word Wide Web，它的目的在于使全球的科学家能够利用Internet交流自己的工作文档。

这个新系统被设计为允许Internet上任意一个用户都可以从许多文档服务计算机的数据库中搜索和获取文档。1990年末，这个新系统的基本框架已经在CERN中的一台计算机中开发出来并实现了,1991年该系统移植到了其他计算机平台，并正式发布了Web技术标准。目前，与Web相关的各种技术标准都由著名的W3C组织（World Wide Web Consortium）管理和维护。

**1.1.2 Web的工作原理**

客户端（浏览器）

在了解Web的工作原理之前需要了解什么呢？Web服务器和Web客户端。

响应

请求

Web服务器：安装了web服务器软件的计算机就是web服务器。web服务器软件对外提供web服务，供客户访问浏览，接收客户端请求，然后将特定内容返回客户端。常见的Web服务器有：IIS、Apache、Tomcat等。

服务器

Web客户端：通常将那些凡是向Web服务器请求获取资源的软件称为Web客户端。Web浏览器是客户端最主要的应用软件，用户只需要安装一个浏览器，便能向服务器发送请求，并得到服务器的响应。常见的浏览器有：Chrome、Firefox、Internet Explorer等。

Web的工作流程：用户点击超链接或在浏览器中输入地址后，此时浏览器将该信息转换成标准的HTTP请求发送给Web服务器。当Web服务器接收到HTTP请求后，根据请求内容查找所需信息资源，找到相应资源后，Web服务器将该部分资源通过标准的HTTP响应发送回浏览器。最后浏览器接收到响应后将HTML文档显示出来。



**图1-1 Web的工作流程**

**1.1.3从网页制作到前端开发**

Web前端开发是从网页制作演变而来的，名称上有很明显的时代特征。在互联网的演化进程中，网页制作是Web1.0时代的产物，那时网站的主要内容都是静态的，用户使用网站的行为也以浏览为主。

（1）Web 1.0纪元

Web 1.0时代开始于1994年，其主要特征是大量使用静态的 HTML 网页来发布信息，并开始使用浏览器来获取信息，这个时候主要是单向的信息传递。市场需求主要是门户网站、企业网站、个人网站。

Web1.0的本质是聚合、联合、搜索，其聚合的对象是巨量、无序的网络信息。Web1.0 只解决了人对信息搜索、聚合的需求，但没有解决人与人之间沟通、互动和参与的需求。主要技术是HTML、CSS、Dreamweaver、Flash、Fireworks等。

Web1.0以静态主;

Web2.0增加了与用户的交互，使网页“动”起来了。

（2）Web 2.0纪元

Web 2.0 始于 2004 年 3 月。在Web2.0中，软件被当成一种服务，Internet从一系列网站演化成一个成熟的为最终用户提供网络应用的服务平台，强调用户的参与、在线的网络协作、数据储存的网络化、社会关系网络、RSS应用以及文件的共享等成为了Web2.0发展的主要支撑和表现。

Web2.0是互联网的一次理念和思想体系的升级换代，由原来的自上而下的由少数资源控制者集中控制主导的互联网体系，转变为自下而上的由广大用户集体智慧和力量主导的互联网体系。主要技术是：HTML、CSS、JavaScript、Ajax等。

（3）HTML5纪元

HTML5是2010年正式提出来的，HTML5时代指的是移动搜索的时代，不是大众用户的时代。HTML5让网页制作从布局到细节处理都更加的灵活，可以创建更好的网页结构，拥有更加丰富的标签，对媒体播放、编辑、存储等有更好的支持方式，兼容性更强。

近年来，越来越多的系统和应用平台选择使用HTML5作为制作标准，网页制作被纳入其中，从各种平台的总结来看，HTML 5两大特点，一是强化了 Web 网页的表现性能。二是追加了本地数据库等 Web 应用的功能，HTML网页制作技术的深入提升，强大的功能陆续的被应用到网页制作、网站建设当中。主要的技术是HTML、CSS、JavaScript、HTML5、CSS3、移动端Web、响应式设计、服务端脚本等。

从技术角度看：

Web1.0依赖的是动HTML和静态HTML网页技术；

Web2.0则JavaScript、XML、ajax等技术和理论为基础；

HTML5则是以HTML5、CSS3以及一些新的框架为技术基础进行开发；

Web3.0的技术特点是综合性的，语义Web、本体是实现Web3.0的关键技术。

（4）Web 3.0纪元

对web3.0的定义是网站内的信息可以直接和其他网站相关信息进行交互，能通过第三方信息平台同时对多家网站的信息进行整合使用；用户在Internet上拥有直接的数据，并能在不同网站上使用； Web3.0浏览器会把网络当成一个可以满足任何查询需求的大型信息库。

Web1.0特征是以静态、单向阅读为主，用户仅是被动参与；Web2.0则是一种以分享特征的实时网络，用户可以实现互动参与，但这种互动仍然是有限度的；Web3.0则以网络化和个性化为特征，可以提供更多人工智能服务，用户可以实现实时参与。

Web1.0是过去时，Web2.0和HTML5是现在时，Web3.0是将来时。从网页制作到前端开发，Web时代不断升级，所需要的技术也越来越多，未来的市场需求也将会越来越大。

**1.1.4前端技术**

进入Web2.0时代过后，网站的前端由此发生了翻天覆地的变化。网页不再只是承载单一的文字和图片，各种富媒体让网页的内容更加生动，网页上软件化的交互形式为用户提供了更好的使用体验，这些都是基于前端技术实现的。

前端三剑客：HTML、CSS、JavaScript。

什么是前端技术？前端技术包括Web页面的结构、Web的外观视觉表现以及Web层面的交互实现。简单来说，就是我们日常浏览网页时所呈现的内容，包括文字、图片、动画特效以及交互行为。现在的前端技术不仅仅包括“三剑客”（HTML、CSS、JavaScript），还需要掌握更多的技能。除此之外，还有HTML5、CSS3、响应式开发、移动端开发等，这些都是当下时代发展的产物，也是时代的潮流。想要成为一名优秀的前端工程师，这些技能都是必不可少的。

前端新技术：

HTML5、CSS3、响应式设计、移动端开发。

下面是前端开发的内容：

* 文本编辑
* 图像处理
* 界面设计
* 网页布局
* 网页样式设计
* 用户与网页的交互设计

**1.1.5后端技术**

后端是指运行在服务器端的程序，这些程序可以动态和互动的页面。一般来说，后端开发一般是由后台开发工程师来进行设计、开发，但是作为前端工程师了解一些后台功能是非常有益的。

后端更多的是与数据库进行交互以处理相应的业务逻辑。需要考虑的是如何实现功能、数据的存取、平台的稳定性与性能等等业务。常用的后端开发技术：Java、Php、Python、ASP.NET、Node.js等。

下面是后端开发的内容：

* 表单处理
* 服务器中信息组织方式的设计
* 数据库编程
* 内容管理系统
* 服务器端应用程序

**1.1.6从前端开发到后端开发的学习路线**

**1.2 Web前端开发技术**

HTML——结构层

CSS——表示层

JavaScript——行为层

简单来说：

HTML是肉身、CSS就是外貌、JavaScript就是灵魂。没有JavaScript，HTML+CSS就是植物人；没有Javascript和CSS，HTML就是个毁容的植物人。

Web前端主要掌握的基本技术——“三剑客”：HTML、CSS、JavaScript；在前端开发中，它们充当着各自的角色，发挥着自己各自的功能。HTML负责结构，CSS负责样式，JavaScript负责交互。本章会对这三门语言进行简单的介绍，后面的章节将会进行详细的学习。

**1.2.1 HTML**

HTML是一种超文本标记语言，超文本标记语言的结构包括“头”部分（head）和“主体”部分（body），其中“头”部提供关于网页的信息，“主体”部分提供网页的具体内容。

HTML并不是一种编程语言，它是一种标记语言，也就是说，它是用来识别和描述一个文件中各个组件的系统，比如标题、段落、列表等。通过HTML就能够搭起我们整个网页的骨架，即网页的整体结构。每个网页的骨架都是按照一定的顺序搭建的，并可以在这个骨架里填充一些内容，比如文字、表格、图像等。

**1.2.2 CSS**

HTML用来描述网页的结构，那些美观、漂亮的网页是怎样做成的呢？在此，CSS就要出场了，它便能使我们的网页变得更加美观。也就是说，我们用HTML搭建好了整个“身体”，那我们应该给她穿上“衣服”，画上“妆容”，这样才能够美丽动人。

字体、色彩、背景图片、行间距、页面布局等都是通过CSS来进行控制的。当然，如果需要给我们的网页添加一些特殊效果或者动画，我们可以通过CSS3来完成。CSS不仅能够提供页面在传统的浏览器和屏幕上展示，还能提供控制网页在其他媒介终端上展示，比如ipad、智能手机等。

CSS样式表是可以重复使用的，增大了代码的利用率，也就是说，多个标签或者元素都可以使用同一个CSS样式，后面章节将会详细介绍。

**1.2.3 JavaScript**

JavaScript是一种脚本语言，在网页中它可以用来添加交互和行为。JavaScript是常用于操纵网页元素或者浏览器窗口功能的一种语言，也是最标准和最普遍使用的脚本语言。

当一个网页有了“美丽的脸蛋”、“衣服”和“身材”，但是如果她不会与你交流，你们之间没有交互、互动，那么这个网页也就没有多大的吸引力，这时候就需要用到JavaScript来为你的网页增添行为。

编写JavaScript需要一定的编程经验，它和HTML、CSS比起来编写的难度就大大的提高了，但是它是网页不可或缺的，对于网站开发很有用处，只要通过不断的练习，便能熟练的掌握JavaScript的编写。

JavaScript常见的行为：

* 处理表单
* 控制元素
* 控制事件的发生
* 获取元素的内容

**1.3 Web前端开发工具**

VS Code、Atom、Sublime Text都是现在最为主流的开发工具，并且都是跨平台的，同时支持Windows、Linux、Mac OS X等操作系统。

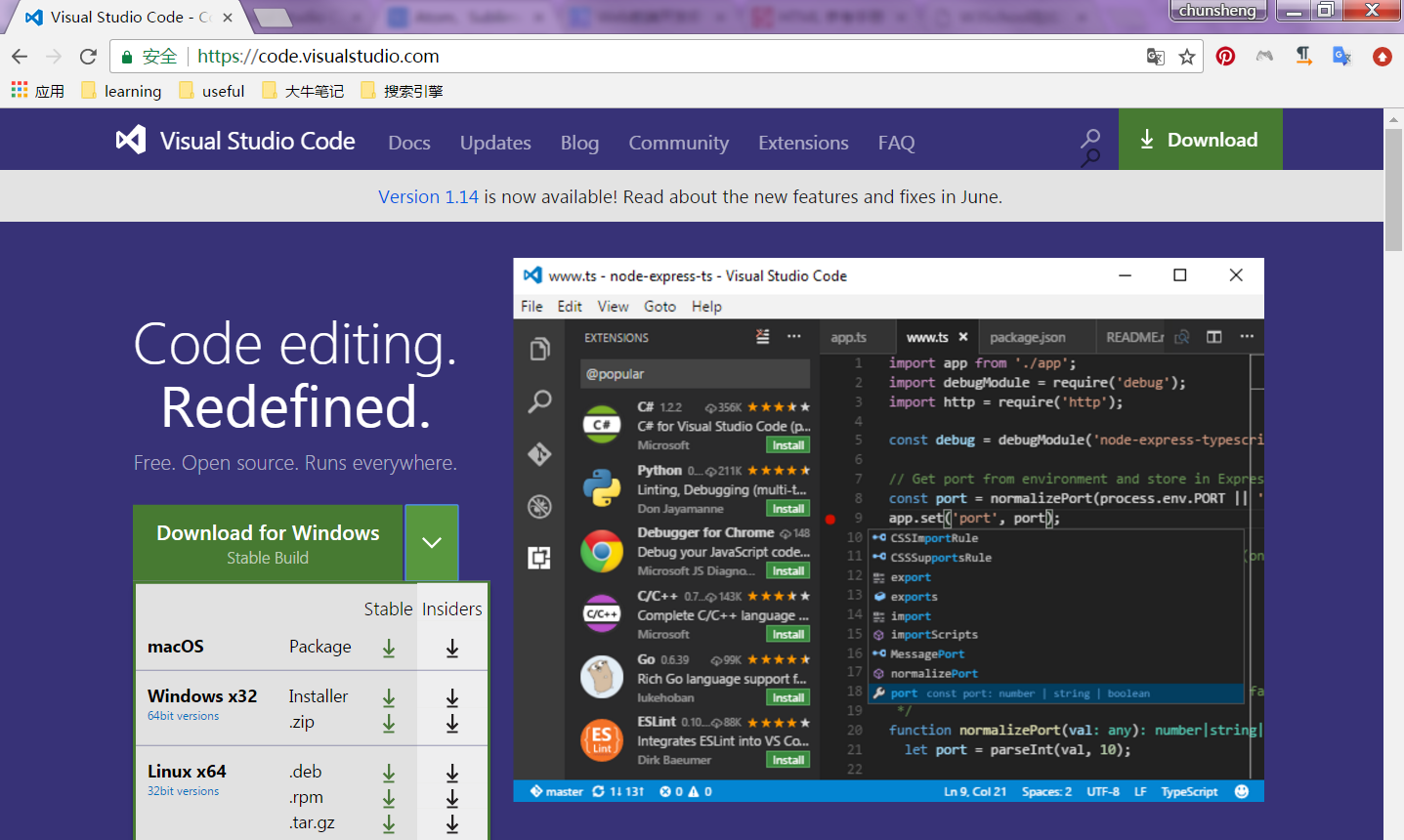
作为前端开发人员，使用一款自己上手且功能强大的开发工具是非常重要的，对于开发效率也会有大大的提高。对于现在的各种各样的开发工具，都有各自的优点，选择好适合自己的开发工具才是最好的开发工具。编者推荐以下3个编译器。

**1.3.1 VS Code**

VS Code（Visual Studio Code）是微软研发的一款免费、开源的跨平台（代码）编辑器，支持Windows、Linux和OS X操作系统。这款编辑器默认集成git（一款代码管理工具，后面章节会有具体介绍），支持git提交。同时也具有开发环境功能，例如代码补全、代码片段、[代码重构](https://zh.wikipedia.org/wiki/%E4%BB%A3%E7%A0%81%E9%87%8D%E6%9E%84)等，还支持扩展程序并在编辑器中内置了扩展程序管理的功能。本书主要代码编写都在VS Code上进行。

VS Code可以在官网上（https://code.visualstudio.com）选择符合自己电脑操作系统的版本进行下载安装，网站见图1-1。

三种编译器都支持插件功能，提高开发效率，安装方法简单快捷，详细的安装方法可参考各编译器的官方网站。



**图1-1 VS Code下载**

VS Code的安装如果是新手用户可以直接进行默认安装，有一定软件安装经验的用户可根据自己的安装经验进行一定的自定义安装。安装完成后，启动VS Code。由于此编辑器默认背景为黑色不利于书本打印工作，这里将把背景换为白色，主题切换方式为File=> preference=>Color Theme，或者使用快捷键Ctrl+T，编译器见1-2。

代码窗口

帮助

调式

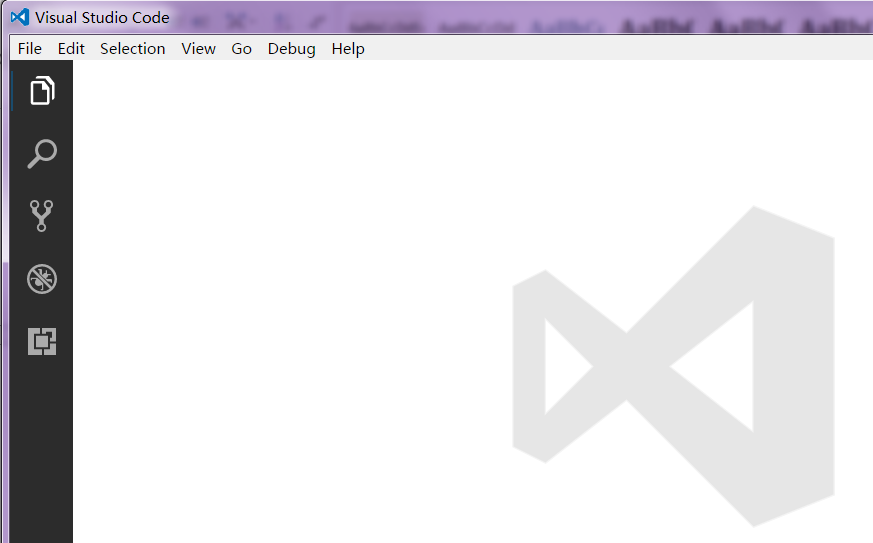
转到

选项

查看

编辑

文件



资源控制器

扩展

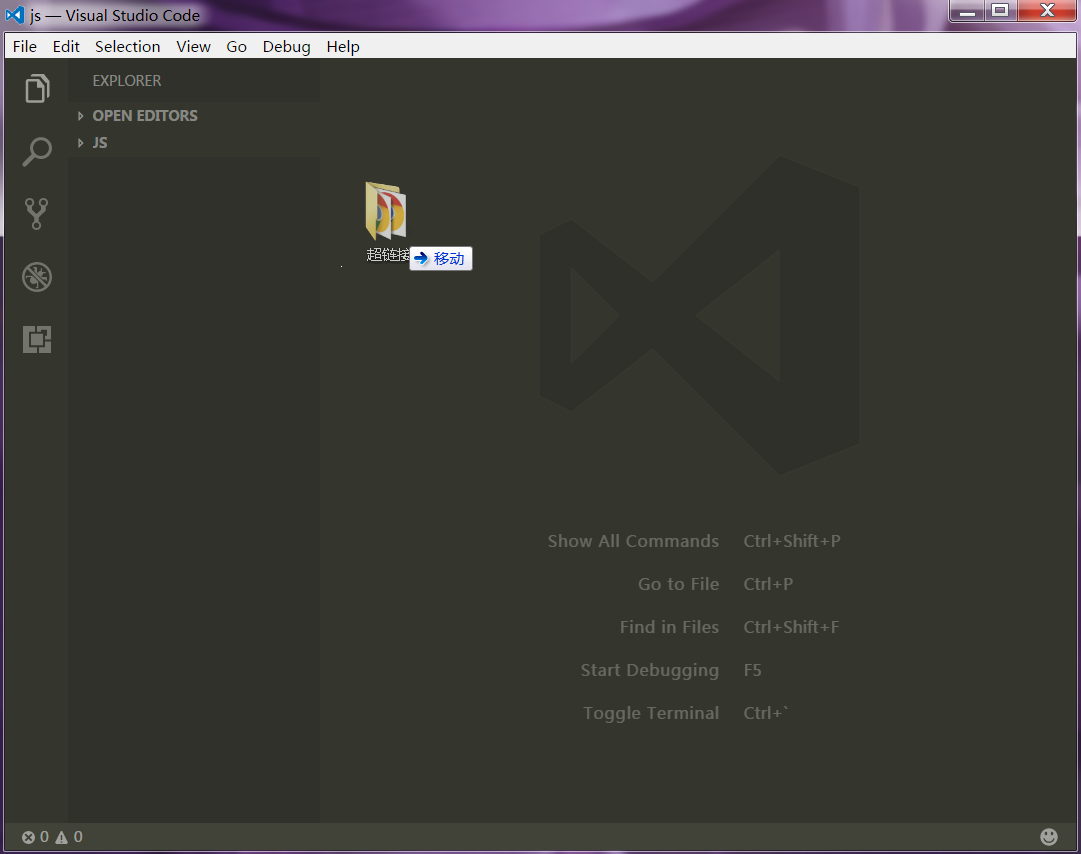
调试

搜索

文件

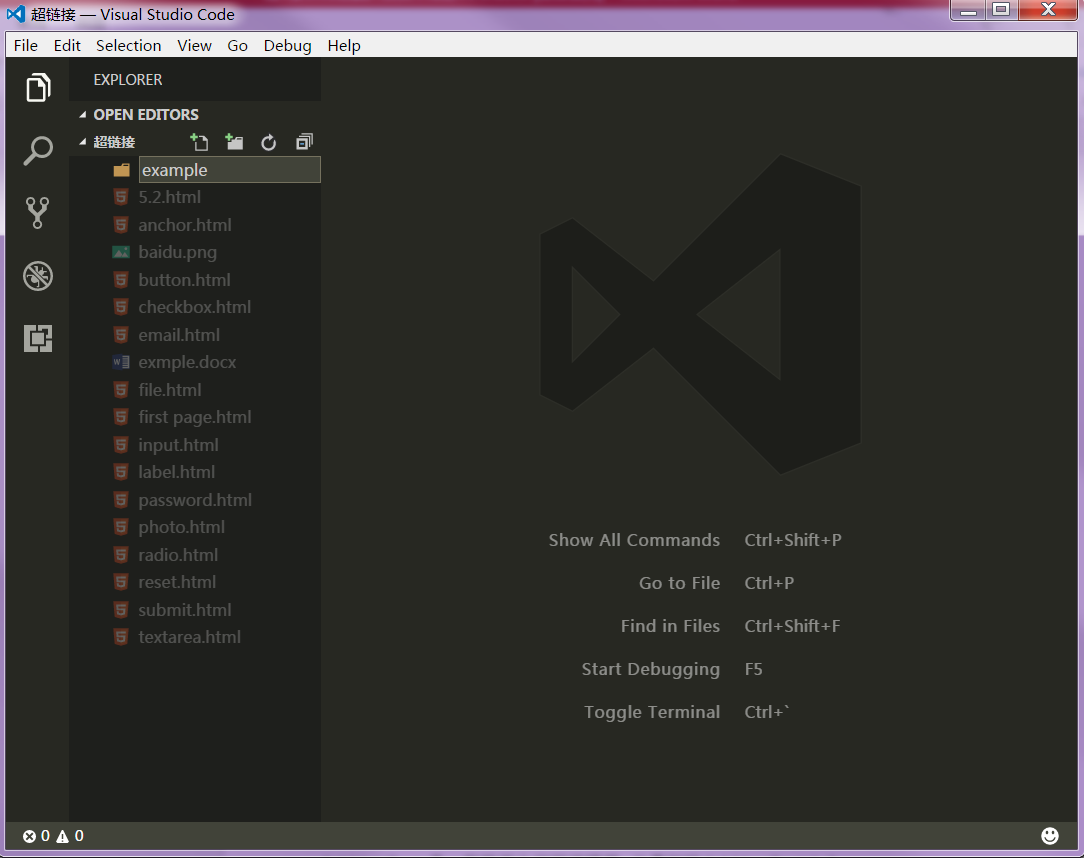
**图1-2 VS Code编辑器**

VS Code的项目文件是针对文件夹，它没有创建项目或者创建站点之类的概念。用户可以在编译器里面创建文件夹作为项目目录，也可以打开或者拖动文件到编译器中作为项目文件。



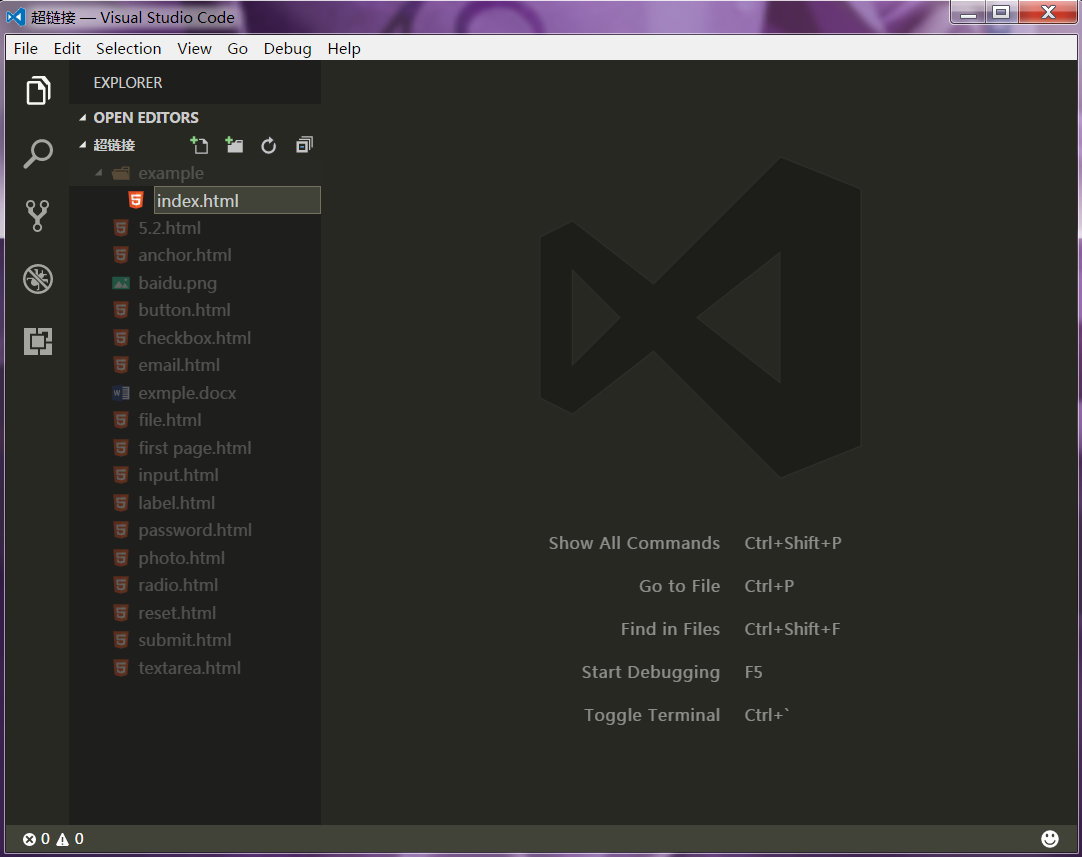
将一个文件夹或文件拖动到编译器中，编译器就会导入文件夹或者打开文件。

**图1-3 拖到文件创建项目**



当鼠标放入目录区域，根目录右侧会显示四个按钮，点击“添加文件夹”按钮再创建一个文件夹。

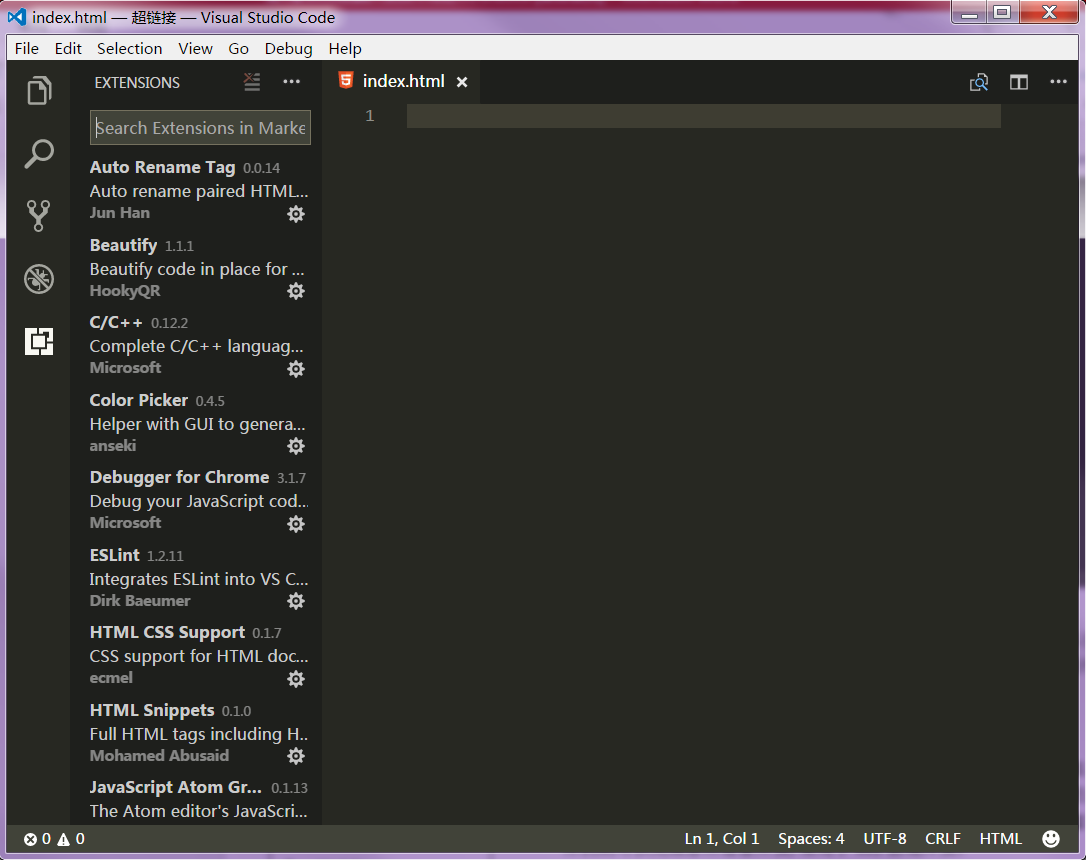
**图1-4 VS Code创建文件夹**



同样鼠标在目录中点击“添加文件”按钮可以直接在项目中增加一个文件。

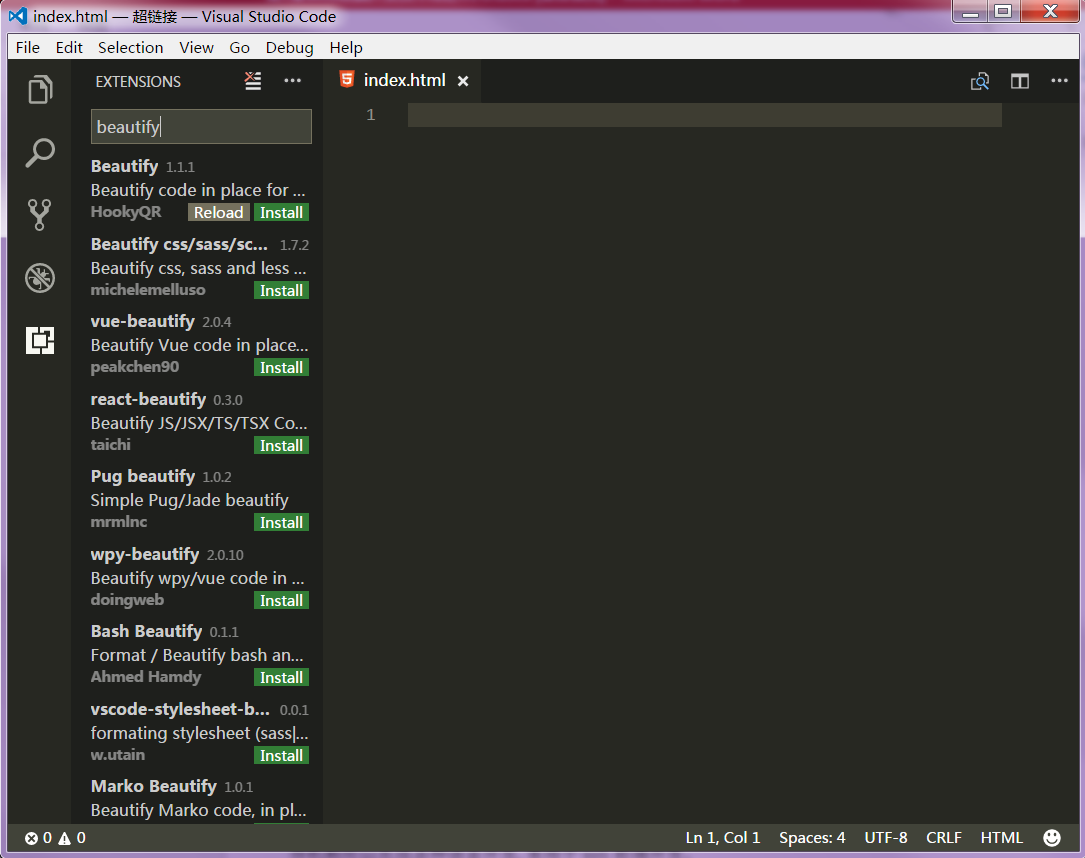
**图1-5 VS Code创建文件**

VS Code本身的功能具有很大的局限性，完全不能说它本身开发效率高。但是由于它可以安装许多的“扩展”，也就是常说的插件，这就使得VS Code能够更高效地进行项目开发。



点击扩展图标，左边默认显示已安装的扩展。

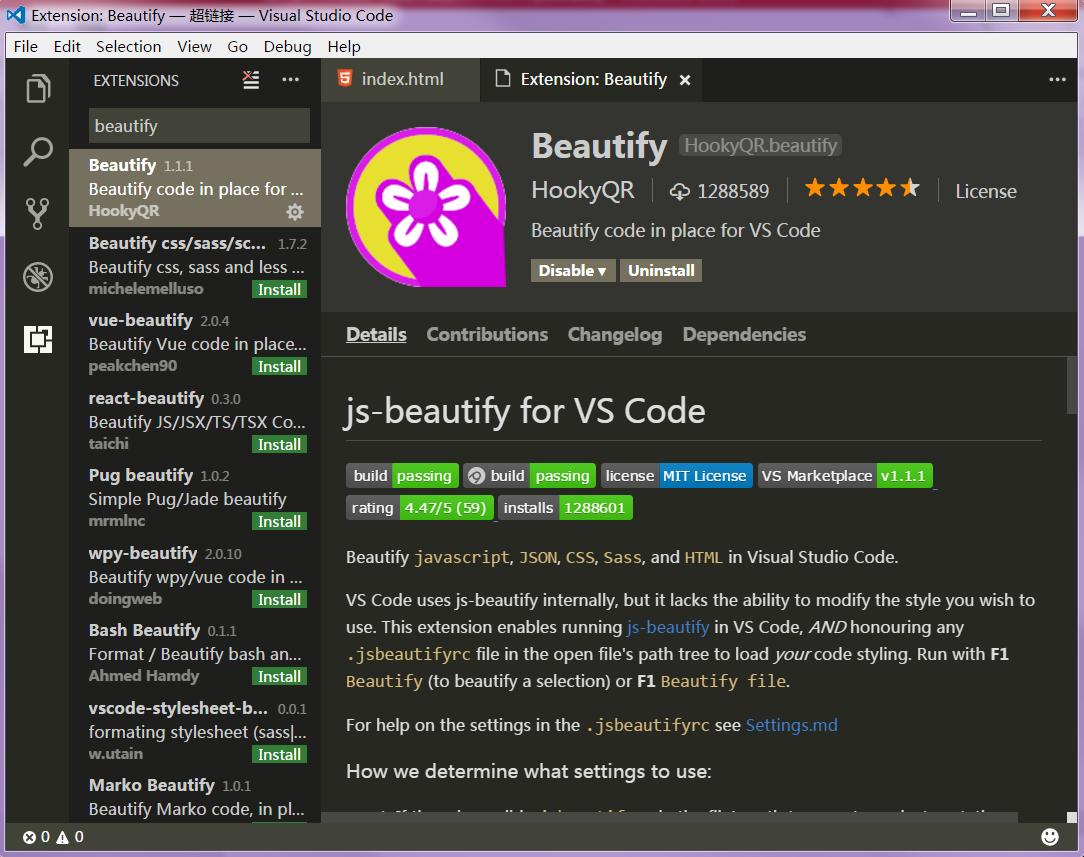
**图1-6 显示扩展程序**



在输入框中输入需要安装的插件名称，然后点击“安装”。安装完成之后点击“重载”之后，插件就可以使用了。

**图1- 7 扩展安装**

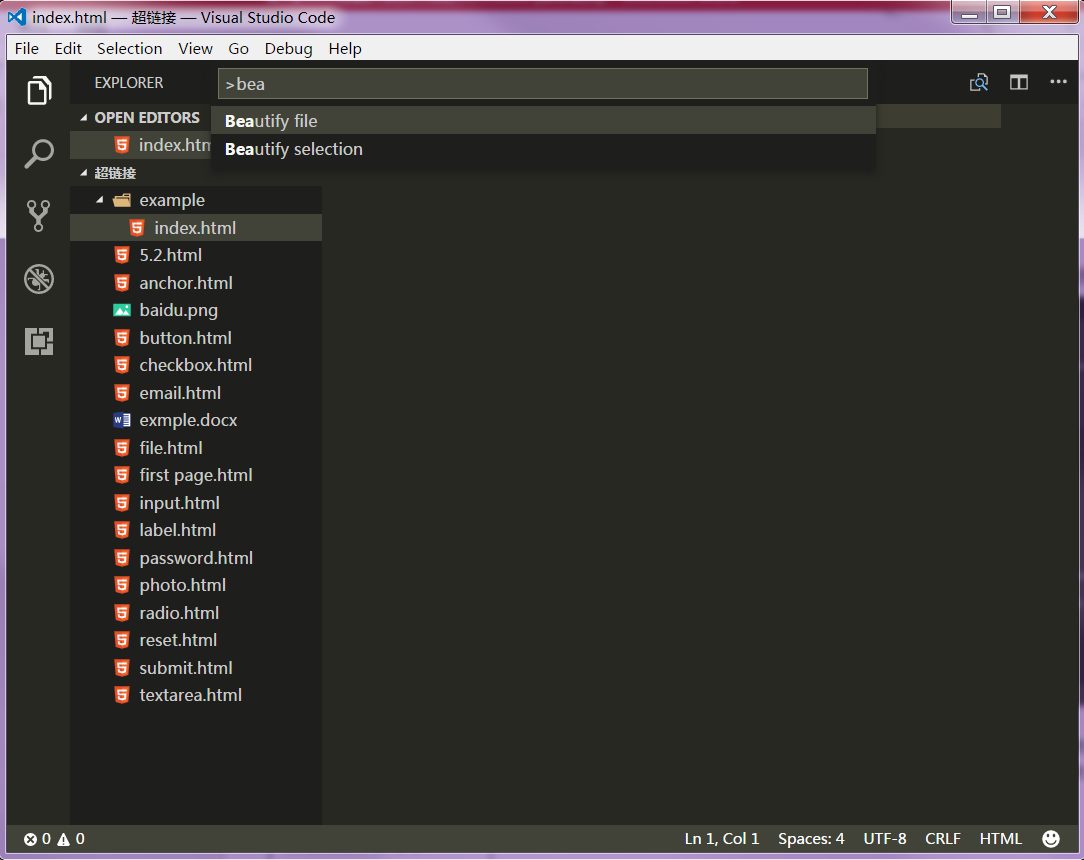
这里示例安装的是Beautify扩展，它的功能是格式化代码。使用该扩展会使你的代码按照一定的规范进行重新排版显示。这里将讲述如何使用扩展，以及用快捷键使用扩展。



双击已安装的扩展，右边就会出现扩展的描述及使用方法。这里是直接在代码文档中按快捷键“F1”。

**图1-8 扩展使用描述**

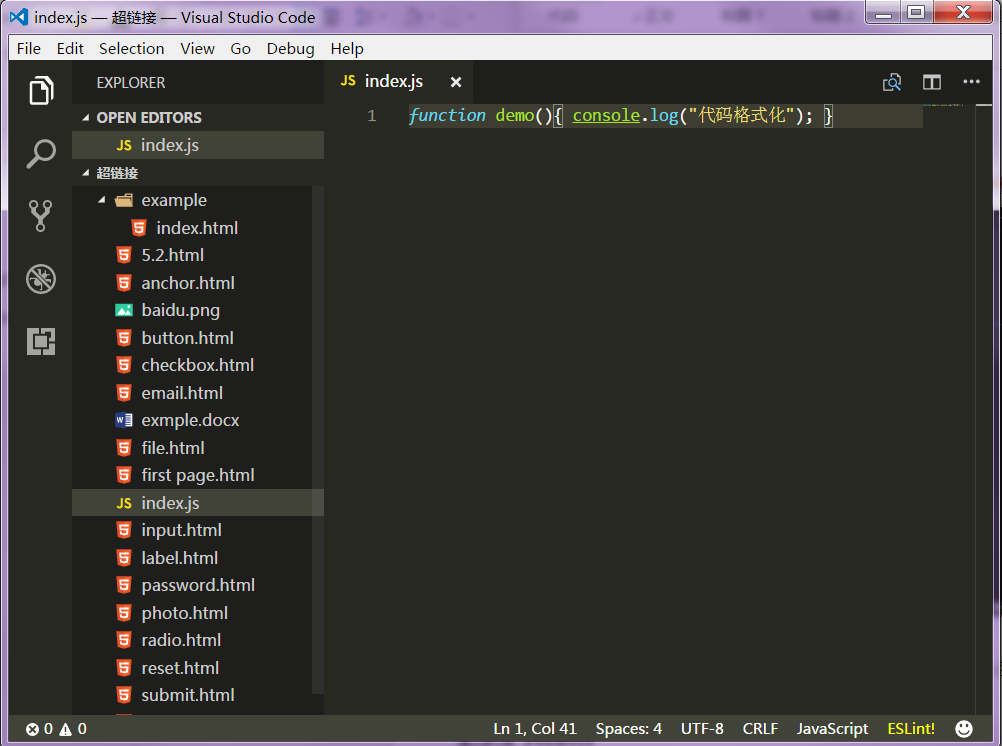
如果这里的快捷键不知道或者是忘记了，可以通过输入关键字来搜索扩展并使用它。通过快捷键Ctrl+Shift+P来打开ShowCommands，来进行输入关键字。



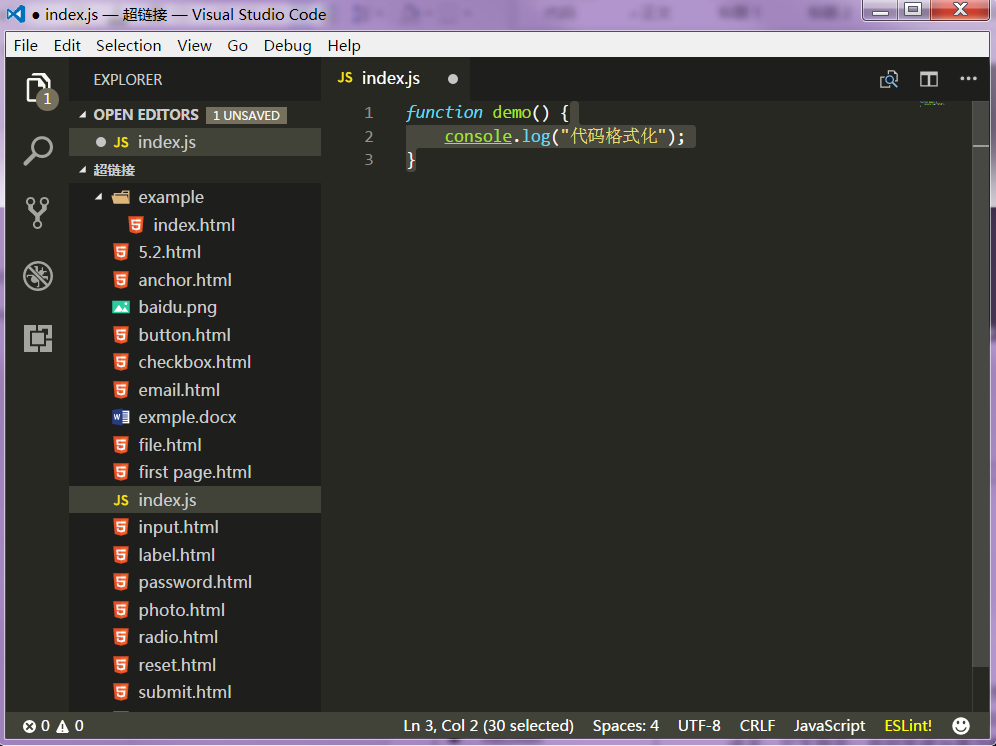
选择到扩展之后回车，扩展即生效。

**图1-9 ShowCommands查找扩展**

这里将进行一个简单的代码格式化演示，我们写一段简单的JavaScript代码然后再把它进行格式化。



**图1-10 代码格式化之前**



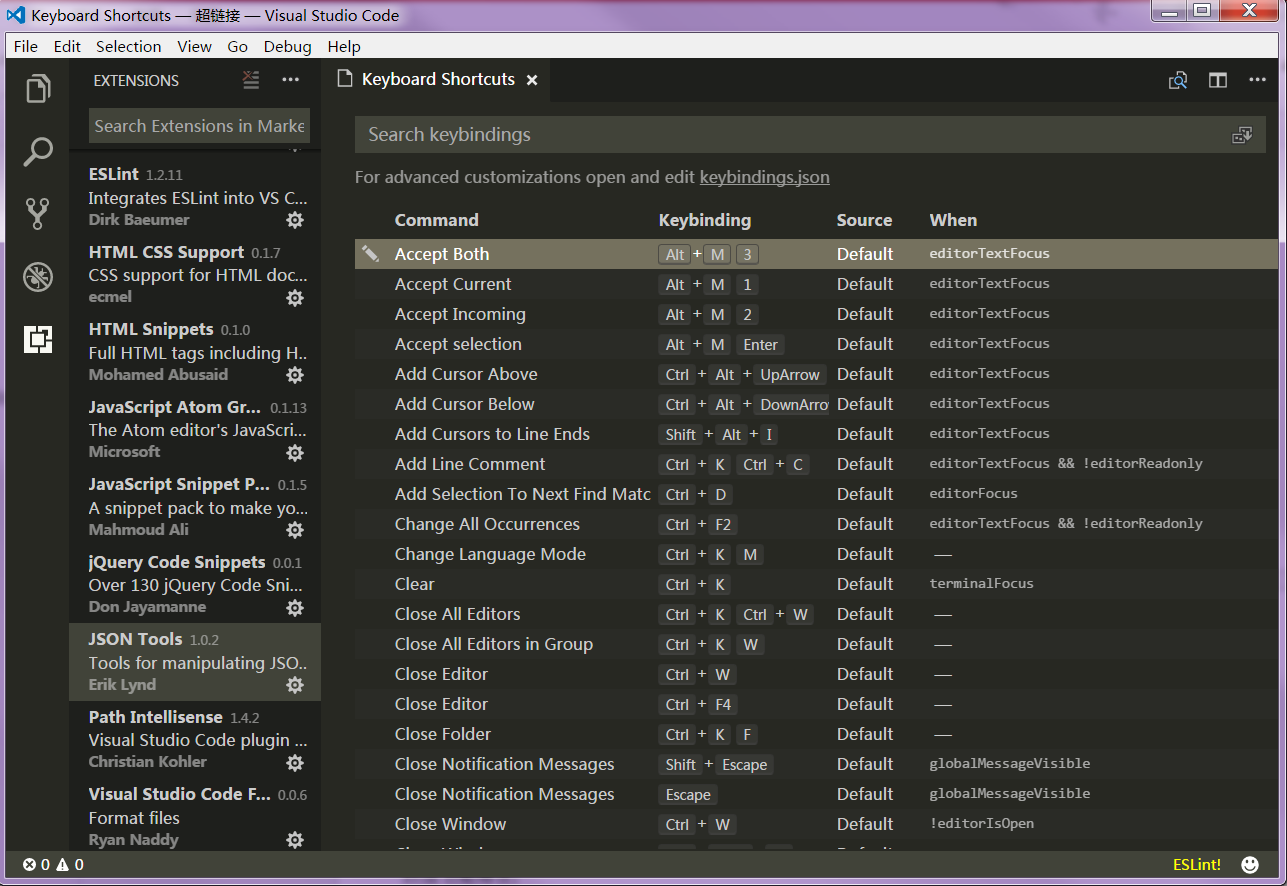
**图1-11 代码格式化之后**

一般开发人员使用一个扩展是远远不够，这里将给出一系列的常用插件安装，见表1-1。

**表1-1 VS Code常用扩展**

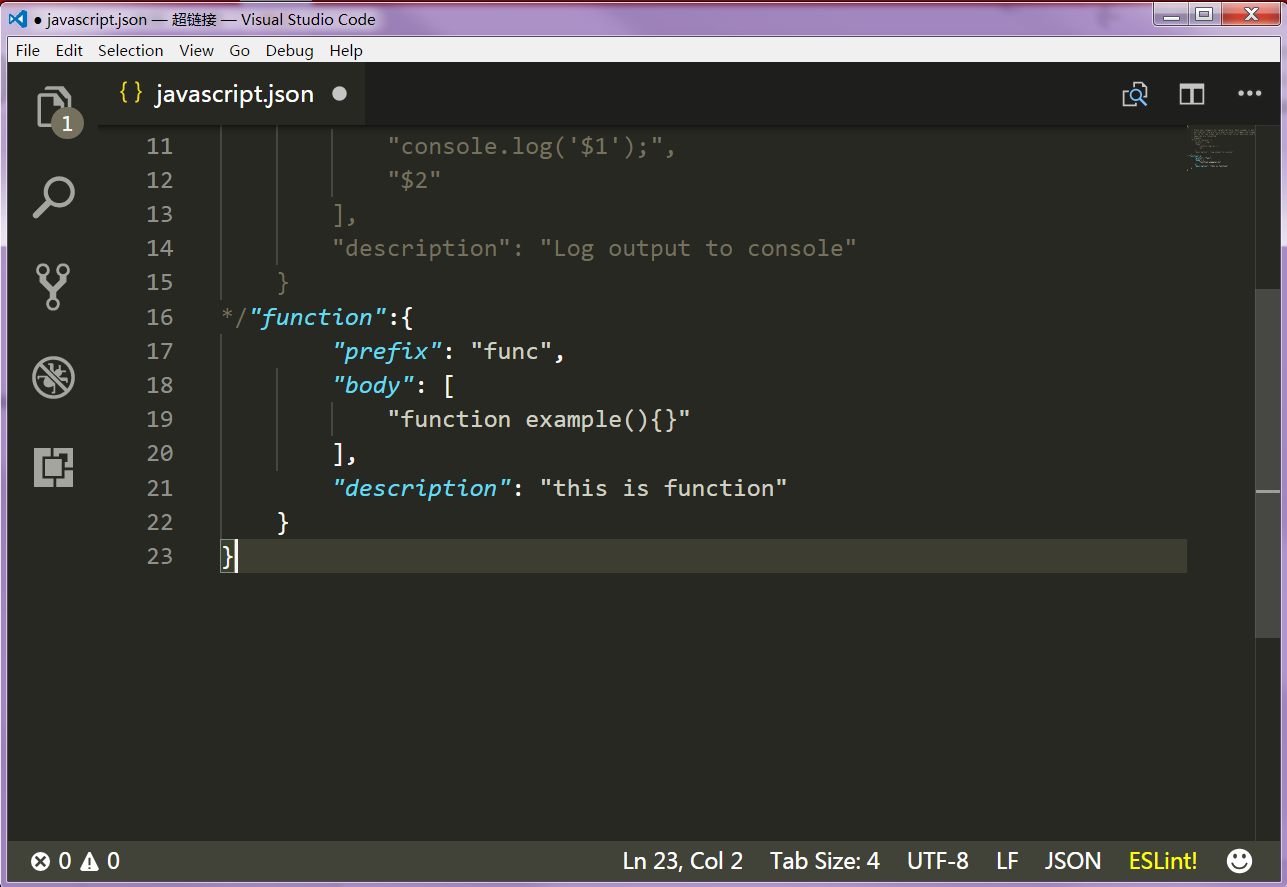
|  |  |
| --- | --- |
| 扩展名称 | 扩展功能 |
| HTML Snippets | HTML代码片段及提示 |
| HTML CSS Support | 写class时智能提示当前项目所支持的样式 |
| Debugger for Chrome | 让vscode映射chrome的debug功能 |
| Jquery Code Snippets | 提示jquery函数名称 |
| Vscode-icon | 为文件及文件夹加上相应文件的图标 |
| Eslint | 为JavaScript自制错误、警告提示规则 |
| Atuo Rename Tag | 修改标签时，同步修改闭合标签 |
| JS-CSS-HTML Formatter | 三种语法格式化 |
| Bootstrap 3 Snippets | 使用Bootstrap框架，自动引入BootStrap文件 |
| Color Picker | 颜色选择器 |

由于VS Code安装了很多扩展，可能会使用户忘记快捷键是什么，还有可能快捷键被其他扩展占用。所以这里将给出所有快捷键列表。用户可以根据自己的习惯进行修改使用。点击File=> preference=>Keyboard shortcuts就进入快捷键展示界面，如果快捷键过多，可根据关键词搜索。



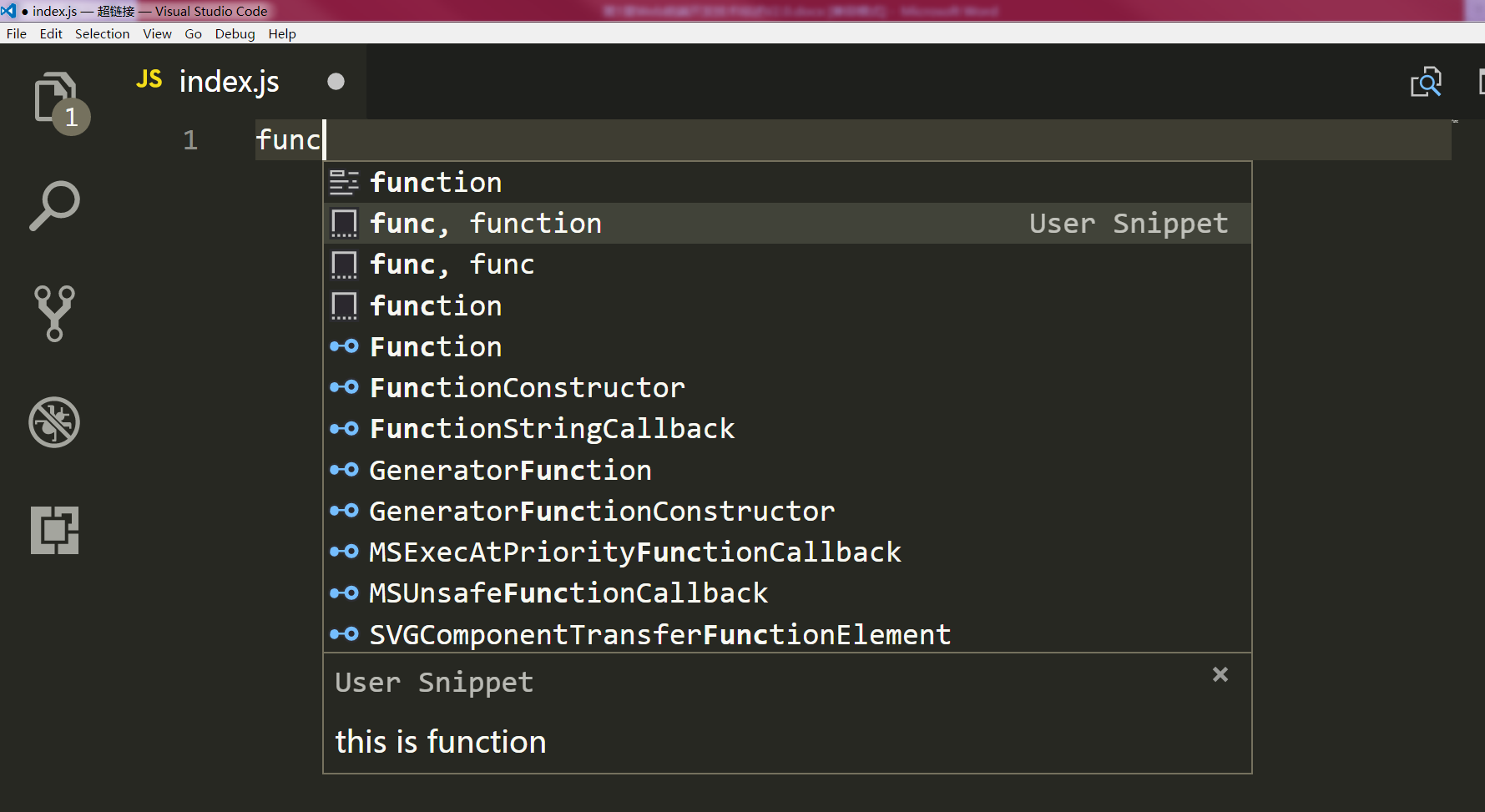
**图1-12 快捷键列表**

在一个经常会使用到同一结构内容的时候，一般的编程人员都会设定属于自己的代码块，以便于更加快捷地完成代码开发。VS Code同样可以轻松地实现代码块的编写。点击File=>preference=>User Snippets，然后选择一门自己经常编写的语法，进入代码块编写界面，过程见图1-13、图1-14及图1-15。

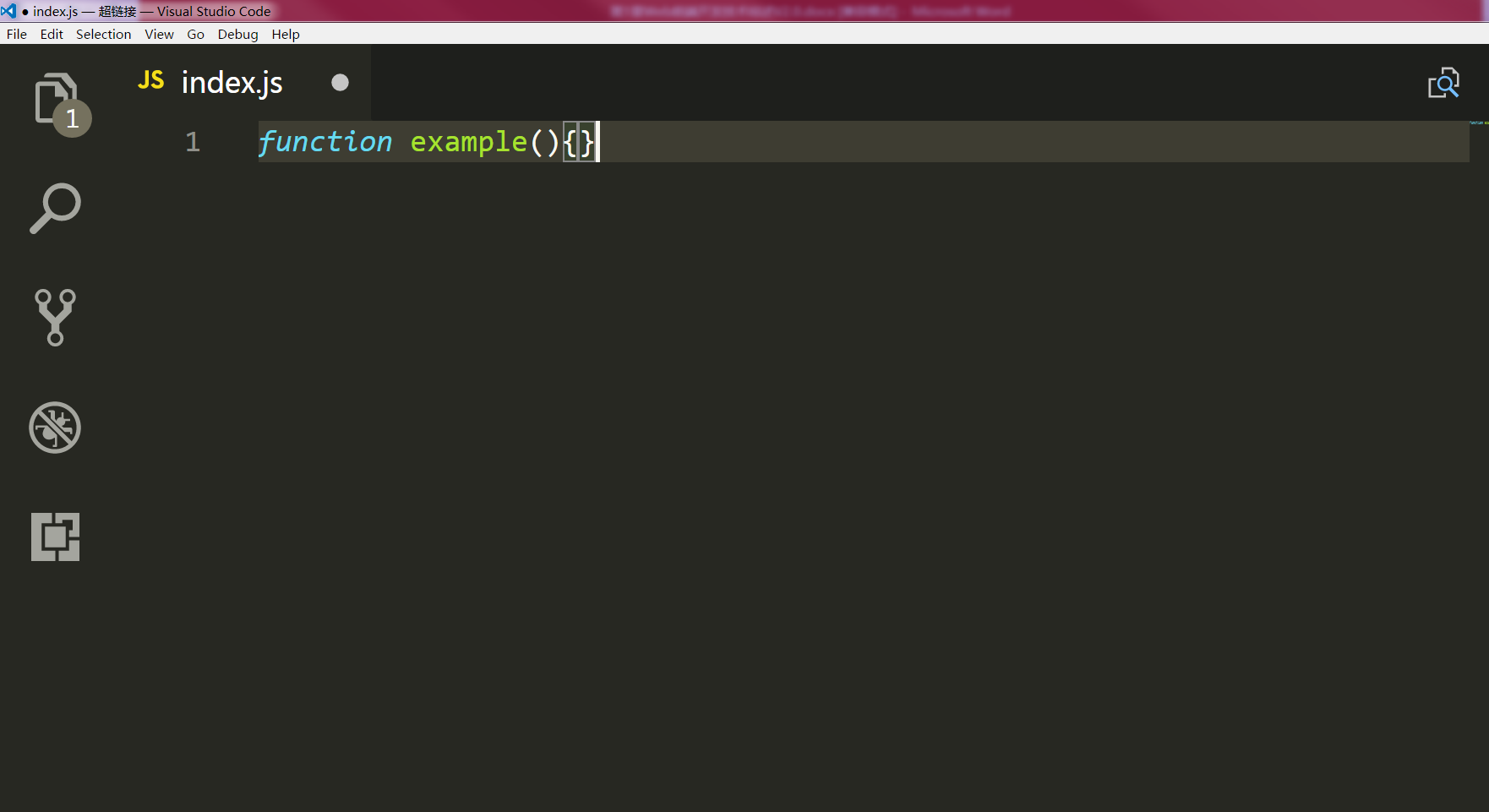


“function”是直接提示显示的字段；“prefix”参数是触发这个代码块的字段；“body”是选择完代码块，回车之后出现的的内容；“description”是这个代码块的描述。

**图1-13 设置代码块**



**图1-14 输入代码块字段**



**图1-15 回车出现代码块**

**1.3.2 Atom**

常见的编译器还有：

* Notepad++
* Editplus
* HBuilder
* Vim
* WebStorm
* Brackets
* Emacs

Atom 是github专门为程序员推出的一个跨平台文本编辑器，具有简洁和直观的图形用户界面，支持CSS、HTML、JavaScript等多种网页编程语言。它支持宏，自动完成分屏功能，集成了文件管理器。通过丰富的插件机制可以完成各种语言开发,常用于web前端开发。

Atom功能特点：

* 界面清新
* 丰富的插件
* git 原生支持
* 自定义界面
* 智能提示

Atom官网地址：<https://atom.io/>

Atom下载地址：<https://atom.io/>

**1.3.3 Sublime Text**

Sublime Text是一款跨平台代码编辑器，从最初的Sublime Text 1.0，到现在的Sublime Text 3.0，Sublime Text从一个不知名的编辑器演变到现在几乎是各平台首选的GUI编辑器，简洁、轻便是其一大特性。

Sublime Text功能特点：

浏览器怎样调试程序？

Chrome浏览器：

鼠标右键=>检查=>控制面板=>调试程序（可查看网页元素、CSS样式、控制台输出内容、查看网络请求等等）

Firefox浏览器：

鼠标右键=>审查元素=>控制面板=>调试程序（查看器、控制台、调试器、样式编辑器等等）

* 主流前端开发编辑器
* 体积较小，运行速度快
* 文本功能强大
* 支持编译功能且可在控制台看到输出
* 支持插件开发以达到可扩展目的

Sublime Text官网地址：<http://www.sublimetext.com/>

Sublime Text下载地址：<https://www.sublimetext.com/3>

**1.4 浏览器工具**

浏览器是指可以显示网页服务器或者文件系统的HTML文件内容，并让用户与这些文件交互的一种软件。它用来显示在万维网或局域网等内的文字、图像及其他信息。这些文字或图像，可以是连接其他网址的超链接，用户可迅速及轻易地浏览各种信息。

对于前端开发人员来说，它不仅仅是用来查看网页效果，还是用来调试代码、调试程序的一个利器，所以熟练使用浏览器是开发人员所必备的技能。常见的浏览器见表1-1.本章将介绍两种在开发过程中最常使用的两种浏览器。

**表1-1 常见的浏览器**

除此之外还有常见的浏览器360浏览器、搜狗浏览器等；

360浏览器和搜狗浏览器都是双内核的，同时支持Trident内核和Webkit内核，极速模式是基于Webkit开发的；兼容模式是基于Trident内核开发的。

|  |  |
| --- | --- |
| 内核 | 浏览器 |
| Trident（又叫IE内核） | IE浏览器 |
| The World浏览器 |
| Maxthon浏览器 |
| Gecko | Netcape 浏览器（Netcape6及以上版本） |
| **Firefox浏览器** |
| Presto | Opera浏览器（Opera7及以上版本） |
| Webkit | Safari浏览器 |
| **Chrome浏览器** |

**1.4.1 Chrome**

****

Chrome，又称谷歌浏览器，是一个由Google（谷歌）公司开发的免费网页浏览器。该浏览器基于其他[开源软件](https://baike.baidu.com/item/%E5%BC%80%E6%BA%90%E8%BD%AF%E4%BB%B6)撰写，包括[WebKit](https://baike.baidu.com/item/WebKit)，目标是提升稳定性、速度和安全性，并创造出简单且有效率的使用者界面。并且顺利通过W3C标准的HTML5和CSS3专业测评。

Chrome功能特点：

* 调试方便
* 不易崩溃
* 速度快
* 安全且支持插件扩展
* 搜索简单

Chrome下载地址：<http://www.google.cn/chrome/business/browser/>

**Chrome扩展安装**

Chrome扩展（插件）上网进行下载到本地进行安装，插件特别多，这里编者推荐一个下载地址（http://chromecj.com/）。除此之外，还可以在Chrome扩展商店进行下载。

* 第一步：启动Chrome浏览器，点击右侧小图标，如图1-17所示（方框圈中）。



**图1-17 选择图标**

* 第二步：点击【更多工具】=>【扩展程序】
* 第三步：如图1-18所示，这是Chrome扩展管理界面，将下载好的扩展文件拖到Chrome扩展管理界面中去，松开鼠标即可。



**图1-18 拖动安装插件**

* 第四步：如图1-19所示，Chrome会弹出安装扩展的提示，点击【添加扩展程序】。



**图1-19 添加扩展程序**

* 第五步：这样整个扩展就安装完毕，点击地址栏右侧小图标（图中方框圈中）即可使用该插件。如图1-20所示。



**图1-20 扩展安装成功**

**1.4.2 Firefox**

****

Firefox，又称火狐浏览器，是一个自由及开放源代码网页浏览器，使用Gecko排版引擎，支持多种操作系统，如Windows、Mac OS X及GNU/Linux等。Firefox支持非常多的网络标准，如标准通用标记语言下的子集HTML和XML、XHTML、CSS（除了标准之外，还有扩充的支持）、ECMAScript（JavaScript）、DOM、XPath和PNG图像文件。

Firefox扩展（插件）安装方法与Chrome类似，具体安装方法参见“1.4.1 Chrome扩展安装”。

Firefox功能特点：

* 插件检查
* 隐私浏览
* 保持同步
* 调试方便
* 硬件加速

Firefox下载地址：<http://www.firefox.com.cn/>

**1.4.3 Chrome与Firefox的区别**

两种浏览器都有各自的优缺点，但他们之间有什么区别呢？见表1-2。在开发过程中，没有绝对好的浏览器调试工具，适合自己的就是最好的。

**表1-2 Chrome与Firefox之间的区别**

|  |  |  |
| --- | --- | --- |
|  | Chrome | Firefox |
| 内核 | Webkit | Gecko |
| 启动速度 | 快 | 较慢 |
| 内存 | 占用内存多 | 占用内存少 |
| 界面 | 简洁 | 复杂 |
| 进程 | 多进程 | 单进程 |
| 定制性 | 弱 | 强 |
| 扩展（插件） | 分类清晰，容易查找 | 扩展丰富 |

* 1. **本章总结**

通过本章的学习，相信你一定对Web的开发有一个深刻的理解，对于它的工作原理、工作机制也有简单的理解。在本章，着重介绍了Web的基本的三门基础技术（“三剑客”）、前端的开发工具以及调试工具，把这些准备工作都做好了，就将正式开启前端开发的学习。