

始于功能，终于代码，拿来即用
语法、规范、理念、工具、技巧，尽在代码中



超实用的 CSS代码段

赵荣娇 任建智 编著

将最有用的CSS代码汇总在一起，提供各种解决实际问题的跨终端方案、兼容、盒子模型、定位、自适应、流和浮动、内核特殊属性、布局差异……



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

TP393.092.2
122



超实用的 CSS代码段

赵荣娇 任建智 编著

电子工业出版社
Publishing House of Electronics Industry

内 容 简 介

本书精选 400 余段 CSS 代码, 覆盖网页上所有的设计元素, 堪称史上最有用的 CSS 书籍, 是网站建设和网页设计人员不可或缺的解决方案、技巧和模板。本书的代码跨平台、跨设备、跨浏览器, 充分向读者演示了如何使用 CSS 的各项技术, 实现令人炫目的网页布局 and 效果。

本书从网页效果的不同类型和使用场景, 对常用的 CSS 代码段进行了全方位的介绍和演示。全书分为 11 章, 包含文字、字体、边框、图片、按钮、链接、背景、颜色、动画、页面布局、美化、盒子、3D、CSS Hack 等网页设计和交互技术, 对那些客户要求高、工作节奏快的网站开发人员和设计人员有着尤其重要的指导作用。

本书内容简洁明了、代码精练、重点突出、实例丰富、语言通俗易懂、原理清晰明白, 是广大网页设计入门者和提高者的良好选择, 同时也非常适合大中专院校学生学习阅读, 也可作为高等院校非计算机专业, 以及计算机非网络工程和相关专业的辅助读物。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

超实用的 CSS 代码段 / 赵荣娇, 任建超编著. —北京: 电子工业出版社, 2014.9
(代码逆袭)

ISBN 978-7-121-23948-9

I. ①超… II. ①赵… ②任… III. ①网页制作工具 IV. ①TP393.092

中国版本图书馆 CIP 数据核字 (2014) 第 173063 号



责任编辑: 董 英

印 刷: 北京京科印刷有限公司

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱

邮编: 100036

开 本: 787×1092 1/16

印张: 24.25

字数: 602 千字

版 次: 2014 年 9 月第 1 版

印 次: 2014 年 9 月第 1 次印刷

定 价: 59.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件到 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前言 →

CSS 是网站一件美丽的外衣，没有 CSS，我们的网页不会如此丰富多彩。所以，要搭建网站和设计网页，就必须学会 CSS。本书是一本研究代码实践的书，它为读者全面深入地讲解了针对各种屏幕大小设计和开发现代网站的 CSS 技术。400 多段代码给读者带来的不仅仅是网页设计的提速，更是教会读者如何应对跨浏览器兼容，如何处理语义化、无障碍访问、搜索引擎优化、创建高性能网页等时时刻刻困扰网页设计人员的问题。

CSS 中的那些槛儿

有些网页设计人员做了多年 CSS，依然在面对问题时束手无策，下列问题很常见，你又能了解多少呢？

- 浏览器的兼容
- 盒子模型
- 绝对定位和相对定位
- 流和浮动
- 自适应的文字、DIV、图片
- Webkit 内核的浏览器的特殊属性
- 动态宽度的布局和固定宽度的布局差异
- IE 6 的兼容

以上所有内容在本书的代码中都有讲解，除这些常见 CSS 门槛外，本书力求将最有用的 CSS 代码汇总在一起，提供各种解决实际问题的跨浏览器方案。

如何学习 CSS

11 个字就能帮助我们更好地学习 CSS。

- **多看、多练：**观摩成功的网页设计，分析并练习网页设计常用的代码。
- **多想、多问：**思考设计实现的原理，提出自己的问题并通过各种渠道来找答案。
- **多总结：**记录前人已经探索出来的 CSS 技巧，总结实战中碰到的问题及解决方案。

只要真正能做到勤思考、勤动手、勤总结，CSS 学习定能一马平川。

本书的编写特点

1. 独特的 CSS 切入点

与市面上其他 CSS 有关的书不同，本书从最常见的网页效果出发，直接应用 CSS 代码实现设计效果，没有用的例子不给，只讲事实不讲废话！

2. 内容丰富，知识全面

本书以网页设计的各个分类和使用场景作为基础，立体式全方位地解释各种场景下的 CSS 代码段应用，场景丰富、实例丰富，并拥有良好的可扩展性、可复用性。

3. 去中心化，分布式学习

本书的代码实例都是独立的，你可以从中间开始学，也可以从头开始学。代码跨平台跨设备，你可以在 iPad 上学，也可以在 PC 上学，如果可以，手机上学代码也完全可能。

4. 解释清晰，原理结合实践

由于 CSS 是网页描述性语言，虽然语法简单，但是很多读者可能不知效果从何做起，本书通过清晰的实现原理分析，配备简单易懂的代码和解释，从效果的实现原理方面进行了剖析，使读者不仅能知其然，更知其所以然。

5. 多种代码方式的实现

本书的实例从纯 CSS 代码、简单的原生 JavaScript 配合、jQuery 框架和自己搭建框架等多种方式实现不同的效果，配合原理的说明，可在不同的方式间自由切换。

6. 自发式学习

在学习代码前，先让读者练习实际上最基础却最容易做错的 CSS 考题和面试题，激发读者的学习斗志。

本书的设计始于功能、终于代码，是网页设计人员的案头必备。

本书的内容安排

本书共 11 章，各章节为不同类别效果的 CSS 实现。

第 1 章为文字与字体，介绍常用的网页文本样式的设计，包括自定义字体、文本缩进与首字符下沉、文本对齐、文字间距、文本溢出、文字阴影、毛玻璃效果、金属质感等装饰、隐藏文本、文字旋转、现代字体栈等。

第 2 章为边框和图片，介绍网页中常见的边框设计和各种图片的展示方式，包括各种边框、黑白图片、图片水印、细节放大、瀑布流、图片墙、图片轮播图（焦点图）、幻灯片（带缩略图）、图片自适应、图片原地放大、图片翻转、图像地图等。

第 3 章为按钮和链接，介绍常见的网页按钮和链接相关的设计，包括圆角按钮、导航菜单、下拉菜单、右键菜单、标签云、文字分享、iPhone 开关、按钮式单选框与复选框、自定义播放器、文字变链接等。

第 4 章为背景和颜色，介绍网页中背景和颜色的使用，包括高光效果页面顶部阴影、多背景、全屏背景、斑马线背景、棋盘背景、易拉罐效果等。

第 5 章为变换与动画的相关内容，是一些网页中动态效果的集合，包括纸张边角动画、气泡式提示、对联广告、页面 loading 效果、进度条、模拟时钟、苹果系统的 DOCK 栏和 Stack 特效、扇形展开等。

第 6 章介绍页面的布局，包括图文混排、几种不同的居中布局方法、绝对定位与相对定位、适配 iPad 屏幕的布局、Clearfix、滚动条的控制、CSS 3 文本分列、Metro 和 Flexbox 布局风格等。

第 7 章为美化与装饰，学习如何在细节方面美化网页，包括文本的装饰、锚链接装饰、自定义滚动条、分隔线、列表符号、跨浏览器的透明度、鼠标指向特效、翻页页码、顶部阴影、页面卷曲、针线缝合效果等。

第 8 章主要介绍 CSS 中的盒子效果，包括 CSS 3 盒模型、内层 CSS 3 盒阴影、外层 CSS 3 盒阴影、纯 CSS 3 透明水晶盒、投影发光效果。

第 9 章内容与 3D 相关，包含 3D 文字、3D 图片立体效果、3D 按钮、3D 下拉菜单和 3D 旋转动画，使网页更加立体化。

第 10 章内容为 CSS Hack，解决因不同浏览器性能的不同带来的效果不一致和效果出错的问题，包含让网站在所有浏览器下显示一致、解决 IE 6 的浮动元素的双倍边距问题、识别不同浏览器、背景与图片透明、IE10 CSS Hack、CSS 3 滤镜。

第 11 章为其他常用代码，既包含某些 CSS 代码技巧，也包含一些不便于分类的代码，还包含某些 JavaScript 与 CSS 结合实例中的 JavaScript 解决方案，内容包含 CSS 块引用模版、一般媒体查询与响应式设计、字符编码、手机 APP 设计等。

本书面对的读者

- 网页设计入门者
- 网页开发入门者
- CSS 学习爱好者
- 由 CSS 2 向 CSS 3 转型的开发人员
- 网页美工人员
- 中小型企业网站开发者
- 大中专院校的学生
- 各种 IT 培训学校的学生
- 网站后台开发人员
- 网站建设与网页设计的相关威客兼职人员

本书的思维导图



编者推荐

本书摒弃传统的说教模式，每段代码都是单独的功能型页面，读者可以从全书的任意一点开始线性阅读。本书的目的就是将最有用的代码与读者分享，包含了网页设计人员在实战中必须具备的所有技巧和方法，读者可以拿来就用。本书的 400 段代码也许并不是最优的代码，但笔者提供了 Github 地址，与世界 CSS 工程师一起优化这些代码，并实现了更新迭代，以保证读者始终能看到最好的、最高效的、最实用的 CSS 代码段。这是一本市场上绝无仅有的 CSS 实战书，是一本值得拥有的 CSS 设计书。

本书的服务

笔者能力有限，如果写作过程中有什么疏漏，或者读者有什么疑问，可通过以下方式与我们沟通。

- QQ 群：296811675，作者在线答疑。
- 扫描封底的微信二维码，时刻参与我们的图书互动和本书的考题答案。
- @博文视点的微博，了解我们发布的信息和各种前端流行技术。
- 博文视点官方网站 <http://www.broadview.com.cn/>，下载本书所有实例源代码。
- Github，<https://github.com/yinqiao/supercss>，了解代码的实时更新和迭代过程，并可以在每章代码下参与讨论；也可以观看其他读者提出的问题，还可以随时随地下载代码。

很多读者在学习过程中苦于无法交流，小故障无法及时解决，加入我们的服务方阵，我们将为您提供终身免费的服务。一本书、一段情、一辈子。

本书主要由赵荣娇、任建智编写，参与编写的人员还有李勇、周敏、王铁民、张兴瑜、马新原、薛淑英、殷龙、席新亮、谢郁、于健、周洋、李春城、李兰英。

序 1 CSS 的前世今生



本书的案例允许读者没有一丁点 CSS 基础，以实践代码为主，帮助读者解构网页中的每个元素，包括文本、链接、图片、层、边框、背景、颜色、布局、美化等所有你能在网页中看到的视觉元素。

这不是一本普通的 CSS 书，它涉及网页设计的技巧、网页性能的优化、拿来主义的应用。它不是你的老师，它只是你的左右手，拥有这些代码，你将会在前端工作中战无不胜！

CSS 1、CSS 2、CSS 3、CSS 4

CSS（Cascading Style Sheets），翻译为层叠样式表。运用 CSS，不仅可以设计出精美的页面效果，而且还能提高网站的可访问性、可维护性以及用户体验。CSS 已经被绝大多数浏览器使用的有 3 个版本：CSS 1、CSS 2 和 CSS 3。

CSS 1 提供有关字体、颜色、位置和文本属性的基本信息，该版本已经得到了目前解析 HTML 和 XML 的浏览器的广泛支持。CSS 1 还未与 HTML 结构脱离，即结构与样式混合嵌套。

CSS 2 推荐的是一套内容和表现效果分离的方式，HTML 元素可以通过 CSS 2 的样式控制显示效果，可完全不使用以往 HTML 中的 table 和 td 来定位表单的外观和样式，只需使用<div>和此类 HTML 标签来分割元素，之后即可通过 CSS 2 样式来定义表单界面的外观。

CSS 3 是 CSS 技术的升级版本，CSS 3 语言开发是朝着模块化发展的。以前的规范作为一个模块实在是太庞大而且比较复杂，所以，把它分解为一些小的模块，更多新的模块也被加入进来。这些模块包括盒子模型、列表模块、超链接方式、语言模块、背景和边框、文字特效、多栏布局、边框、阴影、过渡、3D 动画等。

CSS 的发展可以用表 1 来说明。

表 1 CSS 的发展

时间	特点
很久以前	浏览器不支持任何的CSS属性，所以它们的顺序并不重要
过去一段时间	只有带供应商前缀的属性被支持，所以这些顺序也不重要

续表

时间	特点
现在	带供应商前缀的属性和标准属性都被支持，所以顺序很重要，要把标准属性写在最后，以便让其在被支持时能覆盖供应商属性
将来	只有标准属性被支持，所以顺序又将变得不重要

W3C 于 2011 年 9 月 29 日开始设计 CSS 4，直至现时只有极少数的功能被部分网页浏览器支持，如使用在 HTML 而非 SVG 上的 `pointer-events`。CSS `pointer-events` 属性允许控制在一定情况下（如果有的话）一个特定的图形元素可以成为鼠标事件的目标，当这个属性未指明时，相同可见特征适用于 SVG 内容。CSS 4 增加了一些更方便的方法以选择不同元素，如：

```
01 $ul li.clicked {
02     background: white;
03 }
```

CSS 4 里一项新功能是能够定义于选择器链中哪一个是目标元素。在上一段代码中，`ul` 前面加了一个 `$` 符号，代表它就是要改变的目标，即把里面有 `li`（`class` 是 `clicked`）的 `ul` 的底色转为白色。

CSS 4 里另一项新功能就是能够使用 `:matches()` 来简化选择器，例如：

```
04 :matches(div, p, nav) span{
05     font-size: 18px;
06 }
```

此段代码等同于：

```
07 div span, p span, nav span{
08     font-size: 18px;
09 }
```

事实上现在已有浏览器支持类似的功能，如 Firefox 和 Webkit：

```
10 /*Firefox 使用-moz-前缀*/
11 -moz-any(div, p, nav) span{
12     font-size: 18px;
13 }
14 /*Webkit 使用-webkit-前缀*/
15 -webkit-any(div, p, nav) span{
16     font-size: 18px;
17 }
```

尽管 CSS 4 的内容并不为浏览器所广泛支持，但是，CSS 的更高标准仍然在进一步的推广中。

CSS 的学习目标

在学习 CSS 之前，需要对 HTML、XHTML 有基本的了解。学习 CSS 的目标是如何使

用 CSS 同时控制多重网页的样式和布局，通过使用 CSS 来提升工作效率。

CSS 的学习目标是精通 HTML/XHTML、CSS，熟悉页面架构和布局，对 Web 标准和标签语义化有深入理解，对目前互联网流行的网页制作方法，以及各大浏览器兼容性有极大的了解。

本书的目的就是帮助你：

- 掌握 CSS 基础语法，包括 CSS 的引用方式、id 选择器、类选择器等，如：
`selector {declaration1; declaration2; ... declarationN }`
- 掌握 CSS 的引用方式，包括外部样式表、内部样式表、内联样式、多重样式等方式。
- 掌握 CSS 选择器、CSS 样式、CSS 框模型、CSS 定位等技巧。
- 掌握 CSS 高级语法的使用，包括选择器分组、派生选择器、后代选择器、子元素选择器、相邻兄弟选择器、属性选择器、继承及其他问题，如：

```
h1,h2,h3,h4,h5,h6 { color: green; }
body { font-family: Verdana, sans-serif; }
p, td, ul, ol, li, dl, dt, dd { font-family: Verdana, sans-serif; }
```

- 解决各种浏览器兼容性问题。
- 学习网站性能优化方法。
- 掌握前沿技术（HTML 5+CSS 3）的组合。

30 段必须拥有的 CSS 代码

在开始介绍本书实例前，笔者摘选了 30 段最通用的代码段，不解释，你值得拥有！这些代码穿插在本书的代码实例中，读者如果不太明白，可以通过目录找到具体实例亲自动手练习。

（1）清除浮动的方法

```
.clearfix:before, .container:after { content: ""; display: table; }
.clearfix:after { clear: both; }
/* IE 6/7 */
.clearfix { zoom: 1; }
```

（2）跨浏览器设置透明度

```
.transparent {
    filter: alpha(opacity=50); /* IE */
    -khtml-opacity: 0.5;      /* 老版本的 Safari */
    -moz-opacity: 0.5;        /* Mozilla, Netscape */
    opacity: 0.5;             /* fx, Safari, Opera */
}
```

（3）设置圆角

```
#container {
    -webkit-border-radius: 4px 3px 6px 10px;
```

```
-moz-border-radius: 4px 3px 6px 10px;
-o-border-radius: 4px 3px 6px 10px;
border-radius: 4px 3px 6px 10px;
}
```

(4) 一般媒体查询方法 (针对 PC、智能设备、大屏幕等)

```
/* 智能设备(横屏和竖屏) ----- */
@media only screen
and (min-device-width : 320px) and (max-device-width : 480px) {
    /* 样式设计 */
}
/* 智能设备(竖屏) ----- */
@media only screen and (min-width : 321px) {
    /* 样式设计 */
}
/*智能设备(横屏) ----- */
@media only screen and (max-width : 320px) {
    /* 样式设计 */
}
/* iPads (横屏和竖屏) ----- */
@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) {
    /* 样式设计 */
}
/* iPads (竖屏) ----- */
@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) and
(orientation : landscape) {
    /* 样式设计 */
}
/* iPads (横屏) ----- */
@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) and
(orientation : portrait) {
    /* 样式设计 */
}
/* 台式机和笔记本 ----- */
@media only screen and (min-width : 1224px) {
    /* 样式设计 */
}
/* 超大屏 ----- */
@media only screen and (min-width : 1824px) {
    /* 样式设计 */
}
/* 苹果 iPhone 4 ----- */
@media only screen and (-webkit-min-device-pixel-ratio:1.5), only screen and (min-device-
pixel-ratio:1.5) {
```

```
/* 样式设计 */
}
```

(5) CSS 字体属性的简写（或缩写）

```
body {
    font: font-style font-variant font-weight font-size line-height font-family;
}
```

(6) 自定义文本选择的高亮效果

```
::selection { background: #e2eae2; }
::-moz-selection { background: #e2eae2; }
::-webkit-selection { background: #e2eae2; }
```

(7) 锚链接伪类的设置

```
a:link { color: blue; }
a:visited { color: purple; }
a:hover { color: red; }
a:active { color: yellow; }
```

(8) 全屏背景

```
html {
    background: url('images/bg.jpg') no-repeat center center fixed;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    background-size: cover;
}
```

(9) 垂直居中内容

```
.container {
    min-height: 6.5em;
    display: table-cell;
    vertical-align: middle;
}
```

(10) 强制垂直滚动条

```
html { height: 101% }
```

(11) 设置自定义字体

```
@font-face {
    font-family: 'MyWebFont';
    src: url('webfont.eot'); /* IE9 Compat Modes */
    src: url('webfont.eot?#iefix') format('embedded-opentype'), /* IE6-IE8 */
    url('webfont.woff') format('woff'), /* Modern Browsers */
    url('webfont.ttf') format('truetype'), /* Safari, Android, iOS */
    url('webfont.svg#svgFontName') format('svg'); /* Legacy iOS */
}
body {
    font-family: 'MyWebFont', Arial, sans-serif;
}
```

(12) 自定义段落首字母

```
p:first-letter{
    display: block;
    margin: 5px 0 0 5px;
    float: left;
    color: #ff3366;
    font-size: 5.4em;
    font-family: Georgia, Times New Roman, serif;
}
```

(13) CSS 3 盒子模型内部阴影

```
#mydiv {
    -moz-box-shadow: inset 2px 0 4px #000;
    -webkit-box-shadow: inset 2px 0 4px #000;
    box-shadow: inset 2px 0 4px #000;
}
```

(14) CSS 3 盒子模型外部阴影

```
#mydiv {
    -webkit-box-shadow: 0 2px 2px -2px rgba(0, 0, 0, 0.52);
    -moz-box-shadow: 0 2px 2px -2px rgba(0, 0, 0, 0.52);
    box-shadow: 0 2px 2px -2px rgba(0, 0, 0, 0.52);
}
```

(15) 固定宽度的居中布局

```
#page-wrap {
    width: 800px;
    margin: 0 auto;
}
```

(16) 禁用移动 Webkit 浏览器中的高亮显示

```
body {
    -webkit-touch-callout: none;
    -webkit-user-select: none;
    -khtml-user-select: none;
    -moz-user-select: none;
    -ms-user-select: none;
    user-select: none;
}
```

(17) CSS 省略号

```
div{
    width:200px;                /*设置宽度*/
    white-space:nowrap;         /*设置不折行*/
    text-overflow:ellipsis;     /*这就是省略号喽*/
    -o-text-overflow:ellipsis; /*兼容 Opera*/
    overflow: hidden;           /*将超出的部分设置为隐藏*/
}
```

(18) CSS 3 斑马条纹

```
tbody tr:nth-child(odd) {
    background-color: #ccc;
}
```

(19) 给浏览器的滚动条加上颜色

```
body{
    scrollbar-face-color:#666666;
    scrollbar-shadow-color:#FFFFFF;
    scrollbar-highlight-color:#FFFFFF;
    scrollbar-3dlight-color:#3366cc;
    scrollbar-darkshadow-color:#666666;
    scrollbar-track-color:#EEEEEE;
    scrollbar-arrow-color:#666666
}
```

(20) 文本动画

```
ul {
    animation:6s linear 0 normal none infinite change;
    -webkit-animation:6s linear 0 normal none infinite change;
    -moz-animation:6s linear 0 normal none infinite change;
    -o-animation:6s linear 0 normal none infinite change;
}
@-webkit-keyframes change {
    0%    {margin-top:0;}
    15%   {margin-top:0;}
    25%   {margin-top:-40px;}
    40%   {margin-top:-40px;}
    50%   {margin-top:-80px;}
    65%   {margin-top:-80px;}
    75%   {margin-top:-40px;}
    85%   {margin-top:-40px;}
    100%  {margin-top:0;}
}
@keyframes change {
    0%    {margin-top:0;}
    15%   {margin-top:0;}
    25%   {margin-top:-40px;}
    40%   {margin-top:-40px;}
    50%   {margin-top:-80px;}
    65%   {margin-top:-80px;}
    75%   {margin-top:-40px;}
    85%   {margin-top:-40px;}
    100%  {margin-top:0;}
}
```

(21) 文字阴影

```
text-shadow: 5px 5px 5px #6600ff;
```

(22) 文字的水平居中

```
.container{  
    text-align:center;  
}
```

(23) 文字的垂直居中

```
.container {  
    height: 35px;  
    line-height: 35px;  
}
```

(24) 让子元素相对于父元素垂直居中

```
<div id="big">  
    <div id="small">  
    </div>  
</div>  
#big {  
    position: relative;  
    height: 480px;  
}  
  
#small {  
    position: absolute;  
    top: 50%;  
    height: 240px;  
    margin-top: -120px;  
}
```

(25) 图片宽度的自适应

```
img {  
    max-width: 100%  
}  
_img {  
    width: 100%  
}
```

(26) 简单 3D 按钮

```
#button {  
    background: #888;  
    border: 1px solid;  
    border-color: #999 #777 #777 #999;  
}
```

(27) 用图片美化列表标志

```
ul {  
    list-style: none  
}
```



```
ul li {  
    background-image: url("path-to-your-image");  
    background-repeat: none;  
    background-position: 0 0.5em;  
}
```

(28) 用 CSS 画三角形

```
.triangle {  
    border-color: transparent transparent green transparent;  
    border-style: solid;  
    border-width: 0px 300px 300px 300px;  
    height: 0px;  
    width: 0px;  
}
```

(29) 禁止自动换行

```
h1 {  
    white-space: nowrap;  
}
```

(30) 使用图片替换<h1>标签的元素又不影响 SEO

```
h1 {  
    text-indent: -9999px;  
    background: url("h1-image.jpg") no-repeat;  
    width: 200px;  
    height: 50px;  
}
```

本书浏览器约定

本书涉及的浏览器测试基准如图 1 所示，可以看到内核和外壳的占有情况，内环为内核，外环为外壳，这也是接下来要提到的浏览器同类项的数据基础。

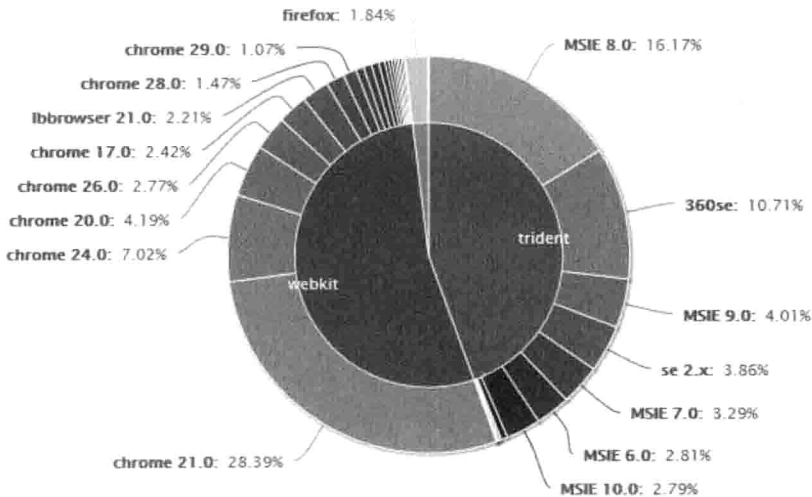


图 1 浏览器基准

不同的浏览器可能会采用相同的内核（渲染引擎）。一方面，外壳实现的差异性会影响整个网站功能，这也是为什么内核无法完全代表所有浏览器的原因。另一方面，某些浏览器具有相同的血统，是可以归为同一类的，同一类中，如果测试通过其中一个，那么其他与之同类的浏览器也基本上可以通过测试。图 1 是根据相同的内核或对 W3C 标准有类似的支持程度将浏览器进行归类的。之所以这样做，是因为测试基准除了要基本反映浏览器的市场占有率，还要考虑到开发测试成本。因此，测试基准中的浏览器应当具有典型代表性。

浏览器的不同版本的内核也不一样，因此通常要针对不同版本的浏览器作测试，开发者要了解内核对标准的支持，比如 IE 的渲染引擎 Trident 的不同版本差别较大，因此 IE 需要测试 6~10 的所有版本，然而由于 Firefox 和 Chrome 升级覆盖面广，因此基准中只保留其最新版本。

本书涉及的代码会根据需要，对不同浏览器进行针对性的测试。