



第6章 CSS3样式基础





学习要点

1

- 了解CSS3背景设置、阴影和渐变

2

- 掌握CSS3设置圆角边框的方法

3

- 熟悉CSS3设置文本效果的方法

4

- 掌握CSS3 transitions过渡效果设计

5

- 掌握CSS3 animation动画效果



CSS3 包含多个新的文本特性，其中Internet Explorer 10、Firefox、Chrome、Safari 以及 Opera 支持属性text-shadow和word-wrap。在CSS3中的属性应用，有些需要添加浏览器私有前缀，这是源于不同内核浏览器对新CSS属性的一个提前支持。

Internet Explorer(简称IE)浏览器IE9以下版本需添加-ms-;

firefox火狐浏览器需添加-moz-;

google chrome、、safari浏览器需添加-webkit-;

opera浏览器，添加前缀-o-;

例如:

-moz-text-shadow:6px 4px 4px #999;

-webkit-text-shadow:6px 4px 4px #999;

text-shadow: 6px 4px 4px #999;



6.1.1 文本阴影效果

`text-shadow` 属性向文本添加一个或多个阴影。该属性是逗号分隔的阴影列表，每个阴影有两个或三个长度值和一个可选的颜色值进行规定。省略的长度是 0。

通过`text-shadow`可以给文本添加阴影效果，其语法如下：

```
text-shadow:h-shadow v-shadow blur color;
```

其中`h-shadow`、`v-shadow`是必选项，

`h-shadow`指定水平偏移量，若是负值则阴影位于文字左边。

`v-shadow`指定垂直偏移量，**若是负值则阴影位于文字上面**。如果两者均为0，则阴影位于文字正后方。

`blur`和`color`为可选，`blur`如果没有指定，则默认为0，值越大，模糊半径越大，阴影也就越大越淡。 不设`color`参数，则由浏览器决定阴影颜色。



文本阴影效果



6.1.2 文本描边效果

文本描边效果可以通过给文本**四周添加差色阴影**的方式进行设置，关键代码如下：

```
text-shadow: -2px 0 white,2px 0 white, 0 2px white,0 -2px white;
```

代码通过设置左侧（**-2px 0**）、上侧（**0 -2px**）、右侧（**2px 0**）、下侧（**0 2px**）四个方向2个像素的白色阴影来呈现描边效果，坐标系向右向上方向为正，负值代表相反方向。



6.1.3 文本浮雕效果

文本浮雕效果可以采用颜色对比，加上阴影效果，使平面的文字看起来像浮雕，比如给文字右侧下侧加上深色阴影，左侧和上侧加上浅色阴影模拟光照，就可以显示浮雕的效果，关键代码如下：

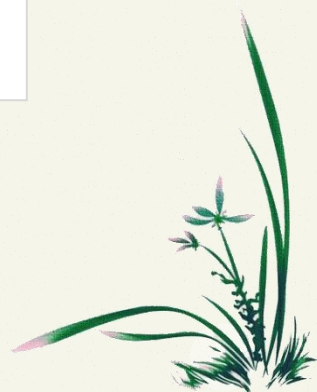
```
text-shadow: -1px -2px #fff,2px 2px #222;
```



案例6-1:

创建名为ch06的网站项目，再新建网页文件，命名为index6-1文本效果.html，分别在网页中编写文本阴影、描边效果和文本浮雕效果的样式，再在<body>标签内引用相关样式，实现不同的文本效果。关键代码如下：

```
/*文本阴影*/  
#yinying{  
    font-size: 40px;  
    color: #294f7b;  
    font-weight: bold;  
    -moz-text-shadow: 6px 4px 4px #999;  
    -webkit-text-shadow: 6px 4px 4px #999;  
    text-shadow: 6px 4px 4px #999;  
}
```



/*描边效果*/

#miaobian{

font-size: 40px;

background-color: #ccc;

padding: 10px;

width: 250px;

font-weight: bold;

color: red;

/*四个方向设置白色的框*/

-moz-text-shadow: -2px 0 white,2px 0 white, 0 2px white,0 -2px white;

-webkit-text-shadow: -2px 0 white,2px 0 white, 0 2px white,0 -2px white;

text-shadow: -2px 0 white,2px 0 white,0 2px white,0 -2px white;

}



```
/*文本浮雕*/
```

```
#fudiao{
```

```
font-size: 40px;
```

```
font-weight: bold;
```

```
color: #eee;
```

```
background:#c0c;
```

```
padding: 10px;
```

```
width: 250px;
```

```
text-shadow: -1px -2px #fff,2px 2px #222;
```

```
-moz-text-shadow: -1px -2px #fff,2px 2px #222;
```

```
/*根据需要添加浏览器兼容，该处兼容firework浏览器
```

```
*/
```

```
}
```



背景作为衬托网页主体内容的容器颜色，给网页增色不少，背景除了最基本的纯色和图像填充外，在CSS3增加了背景渐变效果，渐变是两种或者多种颜色之间的平滑过渡，渐变背景一直以来在web页面中都是一种常见的视觉元素，CSS3的渐变属性主要包括线性渐变、径向渐变和重复渐变。

渐变效果可以用在div,p,form,body,ul,ol等块标签元素上，起到修饰的效果。



CSS3中的线性渐变通过“background-image:linear-gradient（参数值）；”来设置，其基本语法如下：

background-image:linear-gradient([<angle> | <side-or-corner>],color stop, color stop[, color stop]*);

[] 中的参数为可选值，linear-gradient参数取值说明如表6-1 [] 中的参数为可选值，linear-gradient参数取值说明如表6-1



表6-1 linear-gradient参数取值说明

参数类型	取值说明
angle	表示渐变的角度，角度数的取值范围是0~365deg。这个角度是以圆心为起点的角度，并以这个角度为发散方向进行渐变。
side-or-corner	通过关键词来确定渐变的方向。默认值为top（从上向下），取值范围是 [left,right,top,bottom,center,top right,top left,bottom left,bottom right,left center,right center]，其中IE10只能取[left,top]，Chrome则没有[center,left center,right center]；
color stop	用于设置颜色边界，color为边界的颜色，stop为该边界的位置，stop的值为像素数值或百分比数值，若为百分比且小于0%或大于100%则表示该边界位于可视区域外，两个color stop 之间的区域为颜色过渡区。



CSS3中的径向渐变通过“background-image: radial-gradient（参数值）；”来设置，其基本语法格式如下：

background-image:radial-gradient（[圆心坐标（设一个值时为圆，二个值 为椭圆，不设为默认圆的中心）]，[渐变形状（不设默认为椭圆）]，[渐变大小（不设默认为最远端角度）]，**color stop, color stop[, color stop]***（2个以上的渐变颜色及颜色边界值））；

其中[]中的参数为可选值，radial-gradient参数取值说明如表6-2。



表6-2 radial-gradient参数说明表

参数类型	取值说明	
圆心坐标	用于设置放射的圆形坐标，可设置为形如10px 20px 的x-offset y-offset，或使用预设值center（默认值）	
渐变形状	circle	圆形
	ellipse	椭圆形，默认值；
渐变大小	closest-sid或 contain	以距离圆心最近的边的距离作为渐变半径
	closest-corner	以距离圆心最近的角的距离作为渐变半径
	farthest-side	以距离圆心最远的边的距离作为渐变半径
	farthest-corner 或 cove	以距离圆心最远的角的距离作为渐变半径



重复渐变是对以上两种渐变方式使用，只需在两个属性前添加repeating-，具体语法如下：

/*线性重复渐变*/

repeating-linear-gradient([起始角度], color stop, color stop[, color stop]*)

/*径向重复渐变*/

repeating-radial-gradient([圆心坐标],[渐变形状], [渐变大小], color stop, color stop[, color stop]*)



案例6-2:

在ch06的网站项目下新建网页文件，命名为index6-2背景与渐变效果.html，分别在网页中编写不同的渐变样式，再在<body>标签内引用相关样式。

关键代码:

```
.rainbow-linear-gradient{  
    width: 300px;  
    height: 60px;  
    background-image: linear-gradient(right,#E50743 0%,  
    #F9870F 15%, #E8ED30 30%, #3FA62E 45%,#3BB4D7 60%,  
    #2F4D9E 75%,#71378A 80%); }
```

```
.rainbow-radial-gradient{  
    width: 300px;  
    height: 100px;  
    background-image: radial-gradient(100px, #ffe07b  
10%,  
    #ffb151 2%, #16104b 70%);}
```



元素的边框（**border**）是围绕元素内容和那边句的一条或多条线，每个边框有3个属性：宽度、样式以及颜色。

在CSS3中除了上述方法之外，还增加了以下功能：

border-radius：用于创建圆角。

border-image：将图片设置为边框。

box-shadow：用于给边框添加阴影。



6.3.1 圆角边框

网页上的图片可以通过CSS3样式成形圆角边框，要实现这样的效果，在以前的版本中只能由UI设计师给我们制作圆角的背景图，在CSS3中可以使用border-radius属性赋不同的值来设计各种类型的圆角边框。

border-radius属性可以赋4个值，这4个值按照top-left、top-right、bottom-right、bottom-left的顺序来设置。

如果只设置1个值，则表示4个圆角相同。

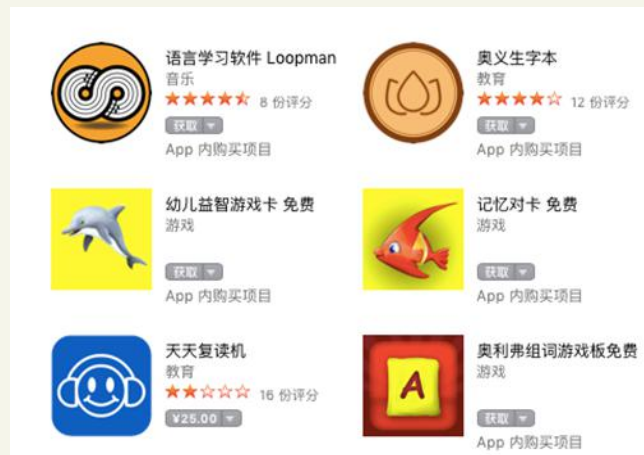
如果设二个值，则第1个值为左上和右下，第2个值为右上和左下。

如果设三个值，则前面三个值分别为左上、右上、右下的圆角，

左下圆角没有设置，默认跟对角线上的右上角相同。

如果设四个参数值，则从左上角开始，按顺时针方向设置角度。

注意：border-radius除了可以用在图片外，还可以用于常用的块标签中，如div、p、ul、li、form等。



6.3.2 CSS3圆角边框应用实例

案例6-3：在ch06的网站项目下新建网页文件，命名为index6-3圆角边框.html，分别在网页中编写不同的圆角参数样式，再在<body>标签内引用相关样式，关键代码如下：



/*一个值，代表四方向50px像素*/

```
#img1{  
    -moz-border-radius: 50px;  
    -webkit-border-radius: 50px;  
    border-radius: 50px;  
}
```

/*两个值，正反对角线方向分别为10px，50px像素*/

```
#img2{  
    -moz-border-radius: 10px 50px;  
    -webkit-border-radius: 10px 50px;  
    border-radius: 10px 50px;}  
}
```

/*三个值，代表左上、右上+左下、右下*/

```
#img3{  
    -moz-border-radius: 10px 50px 80px;  
    -webkit-border-radius: 10px 50px 80px;  
    border-radius: 10px 50px 80px;  
}
```

/*四个值，代表左上、右上、右下、左下*/

```
#img4{  
    -moz-border-radius: 10px 30px 50px 70px;  
    -webkit-border-radius: 10px 30px 50px 70px;  
    border-radius: 10px 30px 50px 70px;  
}
```

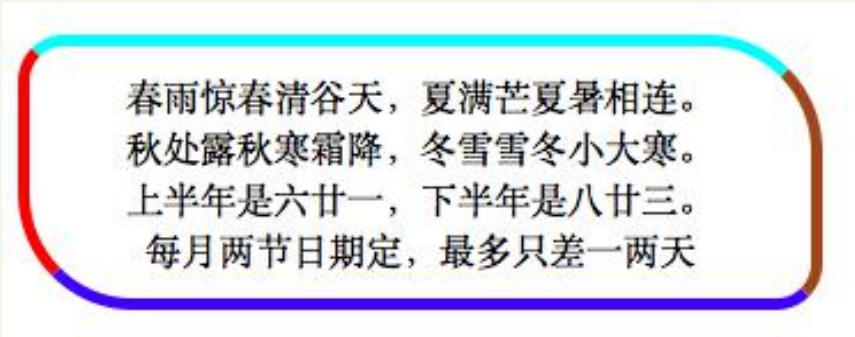


案例6-4:

在ch06的网站项目下新建网页文件，命名为index6-4半径不同的圆角边框.html，分别在网页中编写不同的圆角参数样式，再在<body>标签内引用相关样式，关键代码如下：

```
/*半径不同圆角效果*/  
.jiegige{  
    width: 300px;  
    padding: 10px 20px;  
    border:5px solid;  
    border-radius: 20px 50px;  
    border-top-color:aqua;  
    border-right-color: #994D26;  
    border-left-color: red;  
    border-bottom-color:blue;  
    text-align: center;  
}
```

```
<p class="jiegige">  
    春雨惊春清谷天，夏满芒夏暑相连。<br>  
    秋处露秋寒霜降，冬雪雪冬小大寒。<br>  
    上半年是六廿一，下半年是八廿三。<br>  
    每月两节日期定，最多只差一两天<br>  
</p>
```

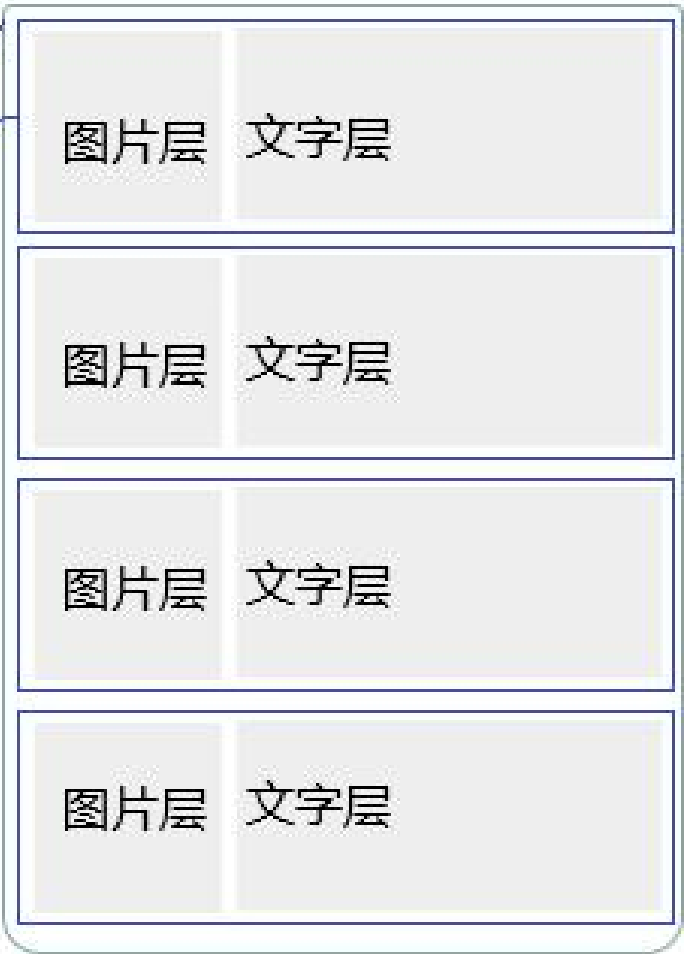


案例6-5:

在ch06的网站项目下新建网页文件，命名为index6-5.html，效果图如下。



容器层



为了加强网页的元素效果，常会使用阴影晕染的方式对于图片、段落的边框等盒子结构进行增强效果，其中CSS3中box-shadow可以进行晕染阴影的设置，语法如下：

box-shadow:h-shadow v-shadow blur spread color inset;

其中h-shadow v-shadow、 box-shadow为必选，其他为可选属性，参数说明如表6-3。

表6-3 box-shadow参数说明

值	说明
h-shadow	必需，水平阴影的位置，正值为水平向右方向，允许负值，负值代表相反方向
v-shadow	必需，垂直阴影的位置，正值为垂直向下方向，允许负值，负值代表相反方向
blur	可选，模糊距离
spread	可选，阴影的尺寸
color	可选，阴影的颜色
inset	可选，将外部阴影（outset）改为内部阴影

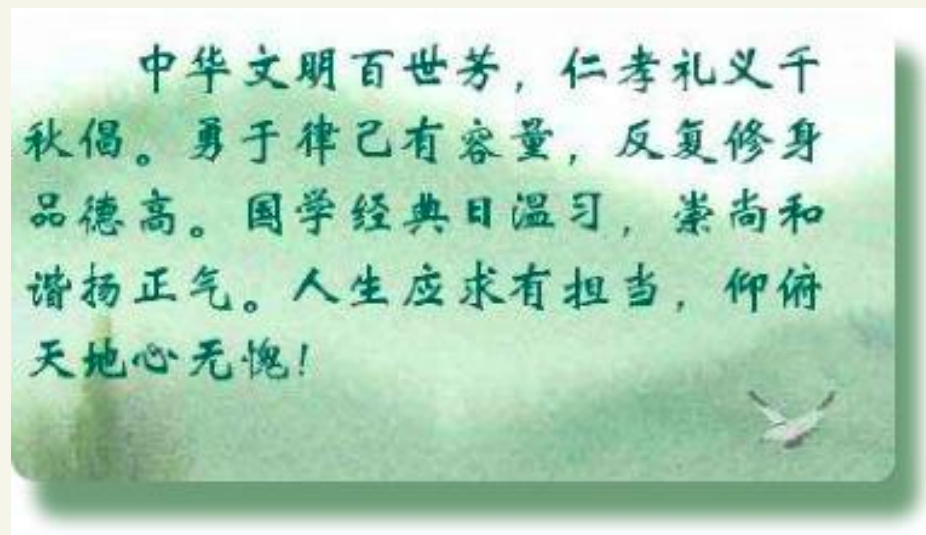


案例6-6:

在ch06的网站项目下新建网页文件，命名为index6-6边框阴影.html，分别在网页中编写不同的图像阴影样式，再在<body>标签内引用相关样式，关键代码如下。

```
img{  
    -moz-border-radius: 10px;  
    -webkit-border-radius: 10px;  
    border-radius: 10px;  
    -webkit-box-shadow: #779977 10px 15px  
8px;  
    box-shadow: 10px 15px 8px #779977;  
}
```

```
<body>  
      
</body>
```



无论是国内网站还是国外网站，动画都在网页上有着举足轻重的作用。动画在网页上随处可见，已经成为网上活动的标志，广泛用于广告、导航、步骤展示等。在CSS3之前，网页做出动画主要由两种方式，一是使用flash，二是使用JavaScript，这两种都需要网页设计人员单独学习Flash和JavaScript以及jQuery技术，且Flash和JavaScript制作的动画在页面加载的时候都需要占用资源。而CSS3的动画能避免上面提到的问题。通过设置CSS3中的属性可以制作两种类型的动画：一种是过渡动画，另一种是动画方式。



CSS3的过渡就是平滑的改变一个元素的CSS值，使元素从一个样式逐渐过渡到另一个样式。

CSS3 过渡使用transition属性来定义，transition属性的基本语法如下所示：

transition:产生过渡效果的属性名property （可以是all或指定宽度、高度颜色等） 持续时间duration 过度速度 timing-function （默认为ease慢速快速再慢速结束） 效果开始需要等待的时间delay （默认为0）

属性	描述
transition-property	规定应用过渡的CSS属性的名称，对应用的属性值有： none all property
transition-duration	定义过渡效果花费的时间（秒或毫秒），默认0没有动画
transition-timing-function	规定过渡效果的时间曲线，默认ease
transition-delay	规定效果开始之前需要等待的时间，过渡效果何时开始，以秒或毫秒计，默认0没有动画

其中属性transition-property三个属性值分别为：

- 1) none: 没有属性会获得过渡效果；
- 2) all: 所有属性都会获得过渡效果；
- 3) property: 定义应用过渡效果的CSS属性名称列表，列表以逗号分隔，例如width、height.



表6-5 transition-property常用过渡属性

属性	改变的对象
background-color	颜色
background-image	只是渐变
background-position	百分比，长度
color	色彩
font-size	百分比，长度
font-weight	数字
Height、left、bottom、top	百分比，长度
text-shadow	阴影
opacity	数字
padding	长度
line-height	百分比，长度，数字

表6-6 transition-timing –function的值

值	说明
linear	规定以相同速度开始至结束的过渡效果
ease	规定慢速开始，然后变快，然后慢速结束的过渡效果
ease-in	规定以慢速开始的过渡效果
ease-out	规定以慢速结束的过渡效果
ease-in-out	规定以慢速开始和结束的过度效果



案例6-7:

在ch06的网站项目下新建网页文件，命名为index6-7过渡动画.html，分别在网页中编写不同的动画样式，再在<body>标签内引用相关样式，关键代码如下：

```
div{
    width: 200px;
    height: 280px;
    background: url(img/qian.png) no-repeat;
    -webkit-transition: all 3s 1s ease-in;
}
div:hover{
    background: url(img/hou.png) no-repeat;
}
```

```
p:hover{
    width: 193px;
    /*过渡属性width*/
    height: 25px;
    background-color:rgba(0,0,230,0.9);
}
```

```
p{
    width: 100px;
    height: 25px;
    text-align: left;
    background-color:rgba(0,0,230,0.5);
    -webkit-transition: all 3s 1s ease-in;
    /*宽度和颜色过渡*/
    transition: all 3s 1s ease-in;
    /*transition: width 3s 1s ease-in;*/
}
```



在CSS3中除了使用transition功能（过渡）实现动画效果以外，还可以使用animation功能实现更为复杂的动画效果。animation是一个复合属性，通过定义多个关键帧以及定义每个关键帧中元素的属性值来实现更为复杂的动画效果。创建动画分为两步，第一步是创建动画，第二步是在对应元素上使用动画。

在CSS3中使用@keyframes规则来创建动画，keyframes可以设置多个关键帧，每个关键帧表示动画过程中的一个状态，多个关键帧就可以使动画十分绚丽。@keyframes规则的语法格式如下所示：

```
@keyframes animationname {  
    keyframes-selector{css-styles;}  
}
```

或说：@keyframes 动画名{ 动画指定位置 { 该位置的状态} }



表6-7 @keyframes属性

类型	描述
animationname	表示当前动画的名称，它将作为引用时的唯一标识，因此不能为空。
keyframes-selector	keyframes-selector是关键帧选择器，即指定当前关键帧要应用到整个动画过程中的位置值可以是一个百分比、from或者to。其中，from和0%效果相同表示动画的开始，to和100%效果相同表示动画的结束。
css-styles	css-styles定义执行到当前关键帧时对应的动画状态。以上三个属性都是必需品，缺一不可。

@keyframes创建动画部分代码，如下：

```
@-webkit-keyframes name{
    0%{
        width: 120px;
    }
    25%{
        width: 200px;
    }
    50%{
        width: 300px;
    }
    100%{
        width: 500px;
    }
}
```



animation属性的基本语法如下所示：

animation: 帧动画名 animation-name 一周动画持续时间 animation- duration [动画速度
曲线 animation- duration] （不设则默认为慢快慢 ease） [动画开始前的延迟时间 animation-
delay] （不设则默认为0） [动画次数 animation- iteration-count] （不设默认1次，infinite
为无限次） [是否反向运动 animation- direction] （默认否，alternate则为可以反向动作）
[是否暂停状态 animation- play-state] （不设则默认为运行）；

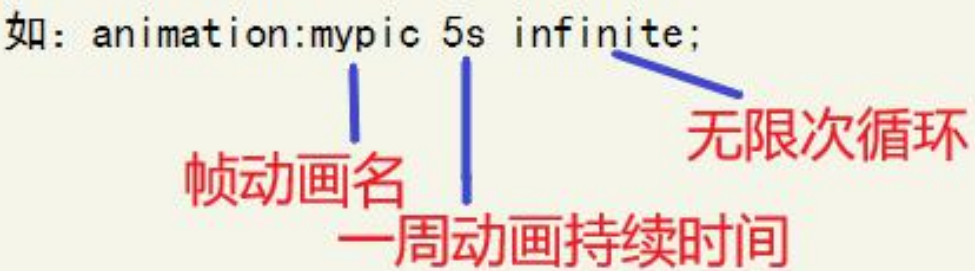


表6-8 animation属性

属性	描述
animation-name	规定@keyframes 动画的名称。
animation- duration	规定动画完成一个周期所花费时间， time值以秒或毫秒计， 默认是0。
animation- timing-function	规定动画的速度曲线。
animation-delay	规定动画开始前的延迟， time值以秒或毫秒计， 默认是0， 可选。
animation- iteration-count	规定动画被播放的次数n， 默认是1。 infinite规定动画应该无限次播放。
animation- direction	规定动画是否在下一周期逆向播放。 normal默认值， 动画应该正常播放。 alternate动画应该轮流反向播放。
animation- play-state	规定动画是否正在运行或暂停。 paused规定动画已暂停。 running默认值， 规定动画正在播放。



案例6-8:

雪碧图是一种CSS图像合并技术生成的图片，该方法是将若干个小图标和背景图像合并到一张图片上，软后利用CSS的背景定位技术来显示需要显示的图片部分，案例使用雪碧图完成生长的花朵效果图。



雪碧图

关键代码:

```
#hua{  
    margin: 0 20px;  
    padding-top: 15px;  
    border-radius: 3px;  
    width: 73px;  
    height: 117px;  
    background-image:url(img/animation.png);  
    animation: hua 5s step-start 0s infinite normal;  
}
```



花生长效果图



案例6-9： 帧动画可以指定运行线路运行，如下图所示，图片按指定轨迹循环往复运行。

定义动画图片框的大小、一周动画持续时间、是否持续播放和是否逆向运行等，代码如下。

```
.animationpic
{
    width:120px;
    height:126px;
    background:red;
    position:relative;
    animation:mypic 5s infinite;
    animation-direction:alternate;
    /* 浏览器兼容Safari 和 Chrome */
    -webkit-animation:mypic 5s infinite;
    -webkit-animation-direction:alternate;
}
```

@keyframes mypic

{

0% {background:red; left:0px; top:0px;}

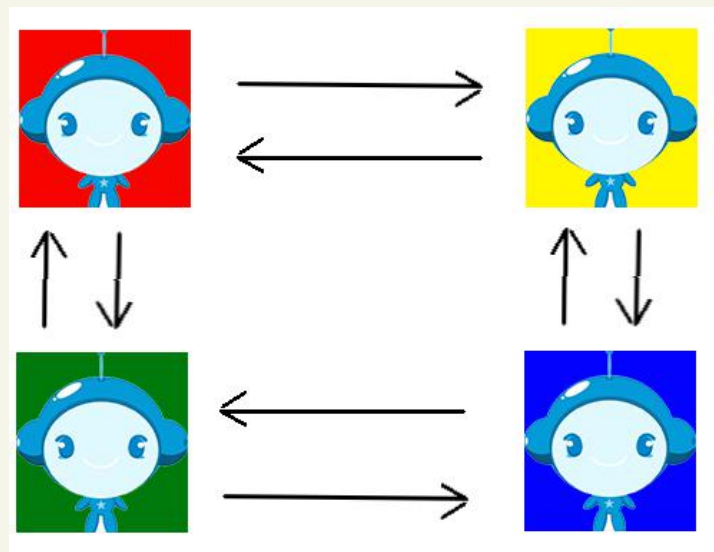
25% {background:yellow; left:200px; top:0px;}

50% {background:blue; left:200px; top:200px;}

75% {background:green; left:0px; top:200px;}

100% {background:red; left:0px; top:0px;}

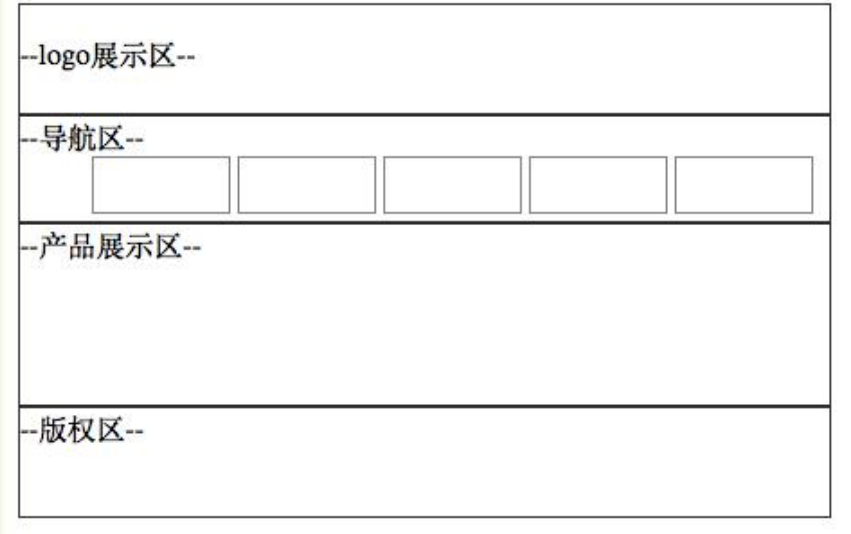
}



案例6-10：综合本章所学知识设计网上花店首页，步骤见课本。



网页结构图



Thank you

