

## 选择器

## Transition, Transform和Animation

这三个特性是CSS3新增的和动画相关的特性。

### Transition

Transition可以在当元素从一种样式变换为另一种样式时为元素添加效果，而不用使用Flash动画或JavaScript。

Transition有如下属性：

1. transition-property: 规定应用过渡的CSS属性的名称。
2. transition-duration: 规定完成过渡效果需要多长时间。
3. transition-delay: 规定过渡效果何时开始，默认是0。
4. transition-timing-function: 规定过渡效果的时间曲线，默认是“ease”，还有linear、ease-in、ease-out、ease-in-out和cubic-bezier等过渡类型。
5. transition: 简写属性，用于在一个属性中设置四个过渡属性。

在一个例子中使用所有过渡属性如下：

使用transition属性简写如下：

### Transform

Transform用来向元素应用各种2D和3D转换，该属性允许我们对元素进行旋转、缩放、移动或倾斜等操作。使用方式如下：

#### 变换类型

transform可以有各种变换类型，即属性值：

### Animation

Animation让CSS拥有了可以制作动画的功能。使用CSS3的Animation制作动画我们可以省去复杂的js代码。使用方法大概如下：

具体用法可以参考教程：[CSS3 Animation](#)。

## 边框

CSS3新增了三个边框属性，分别是border-radius、box-shadow和border-image。border-radius可以创建圆角边框，box-shadow可以为元素添加阴影，border-image可以使用图片来绘制边框。IE9+支持border-radius和box-shadow属性。Firefox、Chrome以及Safari支持所有新的边框属性。

# 背景

CSS3新增了几个关于背景的属性，分别是background-clip、background-origin、background-size和background-break。

## 文字效果

### word-wrap

CSS3中，word-wrap属性允许您允许文本强制文本进行换行，即这意味着会对单词进行拆分。所有主流浏览器都支持 word-wrap 属性。

123	<pre>p { word-wrap: break-word; }</pre>
-----	---

### text-overflow

它与word-wrap是协同工作的，word-wrap设置或检索当当前行超过指定容器的边界时是否断开转行，而 text-overflow则设置或检索当当前行超过指定容器的边界时如何显示。对于“text-overflow”属性，有“clip”和“ellipsis”两种可供选择。

### text-shadow

CSS3中，text-shadow可向文本应用阴影。能够规定水平阴影、垂直阴影、模糊距离，以及阴影的颜色。

123	<pre>h1 { text-shadow: 5px 5px 5px #FF0000; }</pre>
-----	---

### text-decoration

CSS3里面开始支持对文字的更深层次的渲染，具体有三个属性可供设置：

1. text-fill-color: 设置文字内部填充颜色
2. text-stroke-color: 设置文字边界填充颜色
3. text-stroke-width: 设置文字边界宽度

## 渐变

CSS3新增了渐变效果，包括linear-gradient(线性渐变)和radial-gradient(径向渐变)。具体用法参考教程：[CSS3 Gradient](#)

## @font-face特性

在CSS3之前，web设计师必须使用已在用户计算机上安装好的字体。通过CSS3，web设计师可以使用他们喜欢的任意字体。当您找到或购买到希望使用的字体时，可将该字体文件存放

到web服务器上，它会在需要时被自动下载到用户的计算机上。字体是在 CSS3 @font-face 规则中定义的。Firefox、Chrome、Safari以及Opera支持 .ttf(True Type Fonts)和 .otf(OpenType Fonts)类型的字体。IE9+ 支持新的@font-face规则，但是仅支持 .eot类型的字体(Embedded OpenType)。

在新的@font-face规则中，必须首先定义字体的名称（比如myFont），然后指向该字体文件。

如需为HTML元素使用字体，请通过font-family属性来引用字体的名称（myFont）

## 多列布局

通过CSS3，能够创建多个列来对文本进行布局，IE10和Opera支持多列属性。Firefox 需要前缀-moz-，Chrome和Safari需要前缀-webkit-。主要有如下三个属性：

1. column-count: 规定元素应该被分隔的列数。
2. column-gap: 规定列之间的间隔。
3. column-rule: 设置列之间的宽度、样式和颜色规则

## 用户界面

CSS3中，新的用户界面特性包括重设元素尺寸、盒尺寸以及轮廓等。Firefox、Chrome以及Safari 支持resize属性。IE、Chrome、Safari以及Opera支持box-sizing属性。Firefox需要前缀-moz-。

所有主流浏览器都支持outline-offset属性，除了IE。

### resize

resize 属性规定是否可由用户调整元素尺寸。如果希望此属性生效，需要设置元素的 overflow 属性，值可以是 auto、hidden 或 scroll。

### box-sizing

box-sizing属性可设置的值有content-box、border-box和inherit。

1. content-box: padding和border不被包含在定义的width和height之内。对象的实际宽度等于设置的width值和border、padding之和，即 (Element width = width + border + padding)，此属性表现为标准模式下的盒模型。
2. border-box: padding和border被包含在定义的width和height之内。对象的实际宽度就等于设置的width值，即使定义有border和padding也不会改变对象的实际宽度，即 (Element width = width)，此属性表现为怪异模式下的盒模型。

### outline-offset

outline-offset属性对轮廓进行偏移，并在超出边框边缘的位置绘制轮廓。

---

## 弹性盒子

```
display: flex;
align-items: flex-start / center / flex-end;
justify-content: flex-start | flex-end | center | space-between | space-around;
flex-direction: row | row-reverse | column | column-reverse;
flex-wrap: nowrap | wrap | wrap-reverse;
flex-flow: <flex-direction> || <flex-wrap>;
align-content: flex-start | flex-end | center | space-between | space-around | stretch;
```

## 几种可以让元素水平垂直居中的方法

**1绝对定位50%--一半宽度**

**2弹性盒子**

**3margin auto 绝对定位四边为0**

## transform和translate区别

translate: 移动, transform的一个方法

通过 translate() 方法, 元素从其当前位置移动, 根据给定的 left (x 坐标) 和 top (y 坐标) 位置参数:

用法: transform: translate(50px, 100px);

transform: 变形。改变

CSS3中主要包括

**旋转:** rotate() 顺时针旋转给定的角度, 允许负值 rotate(30deg)

**扭曲:** skew() 元素翻转给定的角度, 根据给定的水平线 (X 轴) 和垂直线 (Y 轴) 参数: skew(30deg, 20deg)

**缩放:** scale() 放大或缩小, 根据给定的宽度 (X 轴) 和高度 (Y 轴) 参数: scale(2, 4)

**移动:** translate() 平移, 传进 x, y 值, 代表沿x轴和y轴平移的距离

所有的2D转换方法组合在一起: matrix() 旋转、缩放、移动以及倾斜元素

matrix(scale.x , , , scale.y , translate.x, translate.y)

改变起点位置 transform-origin: bottom left;

综合起来使用: transform: 30deg 1.5 30deg 20deg 100px 200px;

transition: 允许CSS属性值在一定的时间区间内平滑的过渡，  
需要事件的触发，例如单击、获取焦点、失去焦点等

transition:property duration timing-function delay;

property:CSS的属性，例如: width height 为none时停止所有的运动，可以以为transform

duration:持续时间

timing-function:ease等

delay:延迟

注意: 当property为all的时候所有动画

例如: transition:width 2s ease 0s;

## animation:用于设置动画属性

### animation (动画)

语法: `animation: name duration timing-function delay iteration-count direction;`

1. name (需要绑定到选择器的 keyframe 名称)
2. duration (完成动画所花费的时间，以秒或毫秒计)
3. function (动画的速度曲线)
4. delay (动画开始之前的延迟)
5. count (动画应该播放的次数)
6. direction (是否应该轮流反向播放动画)

```
.wrap {
  width: 100px;
  height: 100px;
  background-color: orange;
  margin: 200px auto;
}

.wrap:hover {
  transform: rotate(60deg);
  background-color: red;
  transition: all 1000ms linear 500ms;
}

div {
  width: 100px;
  height: 100px;
  background: #1199ff;
```

```
position: relative;
animation: mymove 5s infinite;
-webkit-animation: mymove 5s infinite;
}

@keyframes mymove {
  1% {
    left: 0px;
  }

  to {
    left: 200px;
  }
}
```

## offsetHeight&clientHeight

offsetHeight是包含contentHeight+padding+borderHeight+ScrollBarHeight的

clientHeight仅包含contentHeight+padding