# background

css2 background:[color] [image] [repeat] [attachment] [position];

css3 background:[image] [position] / [size] [repeat] [attachment] [origin] [clip] [color];

## background-image

none | url(image) | linear-gradient | radial-gradient | repeating-linear-gradient | repeating-radial-gradient

CSS3渐变属性在IE10+、Firefox19.0+、Chrome26.0+和Opera12.1+等浏览器已完全支持W3C的标准语法，但在Webkit内核下的Safari、iOS Safari、Android浏览器和Blackberry浏览器中还是需要添加浏览器的前缀 “-webkit-”

线性渐变

linear-gradient([[<angle> | to <side-or-corner> ],]?<color-stop>[,<color-stop>]+) w3c

linear-gradient(0deg,red 10%,blue);

linear-gradient(to top, red 10%,blue);从底部开始，顺时针旋转

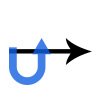


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| to top=0deg | to right=90deg | to bottom=180deg | to left=-90deg | to top right=45deg |

-webkit-linear-gradient([<point>||<angle>,]?<stop>,<stop>[,<stop>]\*) webkit

-webkit-linear-gradient(0deg,red 20px,blue);

-webkit-linear-gradient(left,red 20px,blue);从左边开始，逆时针旋转



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| left=0deg | top=-90deg | right=180deg | bottom=90deg | left top=-45deg |

ie9以下渐变

filter:progid:DXImageTransform.Microsoft.gradient(startColorstr=#ff0000,endColorstr=#00ff00,GradientType=1);

GradientType=0是垂直渐变由上到下，GradientType=1是水平渐变由左到右，颜色值使用6位字符否则颜色不准确。

径向渐变

radial-gradient([[<shape> || <size>] [at <position>]?,| at <position>,]?<color-stop>[,<color-stop>]+); w3c

radial-gradient(farthest-corner at left center, red, blue);

radial-gradient(circle farthest-side at 50% 80%, red, blue);

-webkit-radial-gradient([<position> || <angle>,]?[<shape> || <size>,]?<color-stop>,<color-stop>[,<color-stop>]\*); webkit

-webkit-radial-gradient(50% 80%, closest-corner, red, blue);

-moz-radial-gradient(left top, ellipse closest-side, red, blue);

shape：circle | ellipse

circle：指定圆形的径向渐变

ellipse：指定椭圆形的径向渐变

size：farthest-corner | farthest-side | closest-corner | closest-side

farthest-corner：（默认值）径向渐变的半径长度为从圆心到离圆心最远的角

farthest-side：径向渐变的半径长度为从圆心到离圆心最远的边

closest-corner：径向渐变的半径长度为从圆心到离圆心最近的角

closest-side：径向渐变的半径长度为从圆心到离圆心最近的边

## background-size

auto | length | percentage | cover | contain

auto：默认值，默认的大小，图片的原始大小

length：背景图片大小设置为指定长度大小

background-size:20px 40px;

percentage：背景图片大小设置为绘制背景区域大小的百分比，和background-origin相关

background-size:50%;相当于background-size:50% auto;按背景区域宽度50%等比缩 放

background-size:100% auto;宽度等于背景区域宽度，高度等比缩放

background-size:auto 100%;高度等于背景区域高度，宽度等比缩放

cover：按比例将图片缩放到最小尺寸以使其完整覆盖背景区域，其主要用在背景图片比容器小的情况下，当背景区域和图片比例不同时会裁掉一部分图片

contain：按比例将图片缩放到最大尺寸以使其高宽都在背景区域里面，其主要用在背景图片比容器大的情况下，始终显示全部图片

**兼容性：**

Pc端：ie9以上、chrome、firefox、opear

移动端：安卓4.4默认游览器、uc v9.6.1、QQ浏览器v5.0.2不支持缩写background-size方式，解决方法分开写background-size

## background-origin

padding-box | border-box | content-box

背景图片的定位区域

padding-box：默认值，背景图像相对于内边距框来定位

border-box：背景图像相对于边框盒来定位

content-box：背景图像相对于内容框来定位

当设置background-attachment:fixed;不起作用

作用就是指定background-position的参照点是在边框区域的左上角，还是在补白区域的左上角，或是在内容区域的左上角

## background-clip

border-box |padding-box | content-box

绘制背景区域

border-box：默认值，背景被裁剪到边框盒

padding-box：背景被裁剪到内边距框

content-box：背景被裁剪到内容框

border-box只对背景色起作用，背景图片还是以padding开始覆盖，content-box对背景图和背景色都有作用

真正作用是决定背景在哪些区域显示。如果它的值为border,则背景在元素的边框、补白和内容区域都会显示；如果值为padding,则背景只会在补白和内容区域显示；如果值为content，则背景只会在内容区域显示。

## background-attachment

scroll | fixed | local

规定背景图像是否固定或者随着页面滚动

scroll：默认值，背景图像会随着页面的滚动而移动

fixed：当页面滚动时背景图像不会移动 移动端浏览器不支持

local：父级元素图片可以随子元素内容一起滚动 css3

## background-repeat

repeat-x | repeat-y | [repeat | round | space]{1,2}

repeat：默认值，背景图像在纵向和横向上平铺

no-repeat：背景图像不平铺

repeat-x：背景图像在横向上平铺

repeat-y：背景图像在纵向平铺

round：背景图像自动缩放直到适应且填充满整个容器，图片也能会变形 css3

space：背景图像以相同的间距平铺且填充满整个容器或某个方向 css3

多背景

background-image:url(image), url(image), url(image);

越靠前的层级越高，对应的background-size,background-position可以对应设置