

Crazy advanced CSS

Shadows

To create a text shadow, use

text-shadow: *color offsetX offsetY blur*;

To apply a box shadow to an element, use

box-shadow: *color offsetX offsetY blur*;

where ***color*** is the color of the shadow, ***offsetX*** and ***offsetY*** are the displacements of the shadow from the text in the horizontal and vertical directions (can be negative values), and ***blur*** is the strength of the blurring effect (the bigger, the more blurry).

Shadow examples

```
h1{  
color:rgb(100,150,0);  
text-shadow:black 1px 1px 0px,  
           rgb(150, 200, 0) 5px 5px 10px;  
}
```

This code first makes the h1 a greenish color, then applies two different text-shadows (separated by a comma).

Shadow examples

box-shadow:15px 15px;

Makes a plain black shadow at the bottom-right.

box-shadow: rgb(150, 100, 0) -10px -10px 15px;

Makes a gold-colored shadow at the top-left.

Internet Explorer filters

To apply an Internet Explorer filter, use

```
-ms-filter:"progid:DXImageTransform.Microsoft.filter(param)";
```

where ***filter*** is the name of a visual effect, and ***param*** is the parameter that applies to that effect. These only work with Internet Explorer (and you often have to give them permission to work).

Filter examples

Alpha(style=2)

BasicImage(invert=1)

Emboss()

Engrave()

Pixelate(maxSquare=20)

Wave(freq=2, strength=5)

For example,

```
img{-ms-filter:"progid:DXImageTransform.Microsoft.Emboss()";}
```

Prefixes

Different browsers need different prefixes to display some styles:

-ms- for Internet Explorer

-moz- for Firefox

-webkit- for Safari and Chrome

-o- for Opera

Transformations

You can apply various transformations with the following syntax:

-prefix-transform: effect(params);

where ***prefix*** is the browser prefix, ***effect*** is the transformation function that will be applied to the object, and ***params*** are any parameters required by the transformation. It is usually best to include a separate line of code for each different browser (and Internet Explorer doesn't always work).

Transformation examples

transform: translate(10px, -10px)

Object is moved 10 pixels in horizontally, -10 pixels vertically.

transform: scale(0.8, 0.5)

Dimensions are multiplied by 0.8 horizontally, and 0.5 vertically.

transform: skew(30deg, -10deg)

Object is skewed 30 degrees horizontally, -10 degrees vertically.

transform: rotate(45deg)

Object is rotated 45 degrees clockwise.

Opacity

To create a semi-transparent object, use
opacity: value;

where **value** ranges from 0 (completely transparent) up to 1 (completely opaque).