

CSS Backgrounds

CSS backgrounds are used to add **color or image** behind an element.

Example:

```
div {  
    background-color: lightblue;  
    background-image: url("bg.jpg");  
}
```

This adds a background color and image to the div.

Multiple Backgrounds

CSS allows **more than one background image** on a single element.

Example:

```
div {  
    background-image: url("img1.png"), url("img2.png");  
    background-position: left top, right bottom;  
}
```

First image appears on top of the second one.

Background Size

`background-size` controls the **size of background image**.

Example:

```
div {  
    background-image: url("bg.jpg");  
    background-size: cover;  
}
```

Image covers the entire element.

Background Origin

background-origin defines **where the background starts from**.

Example:

```
div {  
    background-origin: content-box;  
}
```

Background starts from the content area.

Background Clip

background-clip controls **how far the background spreads**.

Example:

```
div {  
    background-clip: padding-box;  
}
```

Background is visible only till padding.

CSS Colors

CSS supports 140+ color names, HEX values, RGB values, RGBA values, HSL values, HSLA values, and opacity.

RGBA Colors

RGBA color values are an extension of [RGB colors](#) with an alpha channel - which specifies the opacity for a color.

An RGBA color value is specified with:

`rgba(red, green, blue, alpha)`

The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example

```
#p1 {background-color: rgba(255, 0, 0, 0.3); /* red with opacity */  
#p2 {background-color: rgba(0, 255, 0, 0.3); /* green with opacity */  
#p3 {background-color: rgba(0, 0, 255, 0.3); /* blue with opacity */
```

HSLA Colors

HSLA color values are an extension of [HSL colors](#) with an alpha channel - which specifies the opacity for a color.

An HSLA color value is specified with:

`hsla(hue, saturation, lightness, alpha)`

The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque):

Example

```
#p1 {background-color: hsla(120, 100%, 50%, 0.3); /* green with opacity */  
#p2 {background-color: hsla(120, 100%, 75%, 0.3); /* light green with opacity */  
#p3 {background-color: hsla(120, 100%, 25%, 0.3); /* dark green with opacity */  
#p4 {background-color: hsla(120, 60%, 70%, 0.3); /* pastel green with opacity */
```

CSS opacity Property

The opacity property sets the opacity for the whole element (both background color and text will be opaque/transparent).

The opacity property value must be a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example

```
#p1 {background-color:rgb(255,0,0);opacity:0.6;} /* red with opacity */  
#p2 {background-color:rgb(0,255,0);opacity:0.6;} /* green with opacity */  
#p3 {background-color:rgb(0,0,255);opacity:0.6;} /* blue with opacity */
```

CSS Color Keywords

CSS has some special predefined color keywords that can be used with properties like color, background-color and border-color.

The special keywords are:

- transparent
 - currentcolor
 - inherit
-

The transparent Keyword

The transparent keyword is used to make a color transparent. This is often used to make a transparent background color for an element.

Example

Here, the background color of the <div> element will be fully transparent, and the background image will show through:

```
body {  
background-image: url("paper.gif");
```

```
}
```

```
div {  
    background-color: transparent;  
}
```

The currentcolor Keyword

The `currentcolor` keyword is like a variable that holds the current value of the `color` property of an element.

This keyword can be useful if you want a specific color to be consistent in an element or a page.

Example

In this example the border color of the `<div>` element will be blue, because the text color of the `<div>` element is blue:

```
div {  
    color: blue;  
    border: 10px solid currentcolor;  
}
```

Example

In this example the `<div>`'s background color is set to the current color value of the `body` element:

```
body {  
    color: purple;  
}  
  
div {  
    background-color: currentcolor;  
}
```

Example

In this example the <div>'s border color and shadow color is set to the current color value of the body element:

```
body {  
    color: green;  
}  
  
div {  
    box-shadow: 0px 0px 15px currentcolor;  
    border: 5px solid currentcolor;  
}
```

The inherit Keyword

The inherit keyword specifies that a property should inherit its value from its parent element.

The inherit keyword can be used for any CSS property, and on any HTML element.

Example

In this example the 's border settings will be inherited from the parent element:

```
div {  
    border: 2px solid red;  
}  
  
span {  
    border: inherit;  
}
```

CSS Gradients

CSS **gradients** are used to create **smooth color transitions** instead of using image files.

They are mostly used as **backgrounds** and help make designs more attractive.

1. Linear Gradients

A **Linear Gradient** changes colors in a **straight line** (top to bottom, left to right, or at an angle).

Example:

```
div {  
    background: linear-gradient(to right, red, yellow);  
}
```

Color smoothly changes from **red to yellow horizontally**.

Use: Buttons, headers, banners.

2. Radial Gradients

A **Radial Gradient** spreads colors **from the center outward in a circular or oval shape**.

Example:

```
div {  
    background: radial-gradient(circle, red, yellow);  
}
```

Red starts from the center and fades into yellow.

Use: Highlights, background effects.

3. Conic Gradients

A **Conic Gradient** rotates colors **around a center point**, like a pie chart.

Example:

```
div {  
background: conic-gradient(red, yellow, green);  
}
```

Colors rotate in a circular pattern.

Use: Pie charts, loaders, decorative designs.
