There's a newer version of Bootstrap!

Search docs... Ctrl + /

v5.1

View on GitHub

Background

Convey meaning through `background-color` and add decoration with gradients.

Design and Development tips in your inbox. Every weekday.

ads via Carbon

On this page

Background color

Similar to the contextual text color classes, set the background of an element to any contextual class. Background utilities **do not set** `color`, so in some cases you'll want to use `.text-*` color utilities.

.bg-primary

.bg-secondary

.bg-success

.bg-danger

.bg-warning

.bg-info

.bg-light

.bg-dark

.bg-body

.bg-white

.bg-transparent

```
<div class="p-3 mb-2 bg-primary text-white">.bg-primary</div>
<div class="p-3 mb-2 bg-secondary text-white">.bg-secondary</div>
<div class="p-3 mb-2 bg-success text-white">.bg-success</div>
<div class="p-3 mb-2 bg-danger text-white">.bg-danger</div>
<div class="p-3 mb-2 bg-warning text-dark">.bg-warning</div>
<div class="p-3 mb-2 bg-info text-dark">.bg-info</div>
<div class="p-3 mb-2 bg-light text-dark">.bg-light</div>
<div class="p-3 mb-2 bg-dark text-white">.bg-dark</div>
<div class="p-3 mb-2 bg-body text-dark">.bg-body</div>
<div class="p-3 mb-2 bg-white text-dark">.bg-white</div>
<div class="p-3 mb-2 bg-transparent text-dark">.bg-transparent</div>
```

Background gradient

By adding a `.bg-gradient` class, a linear gradient is added as background image to the backgrounds. This gradient starts with a semi-transparent white which fades out to the bottom.

Do you need a gradient in your custom CSS? Just add `background-image: var(--bs-gradient);`.

.bg-primary.bg-gradient

.bg-secondary.bg-gradient

.bg-success.bg-gradient

.bg-danger.bg-gradient

.bg-warning.bg-gradient

.bg-info.bg-gradient

.bg-light.bg-gradient

.bg-dark.bg-gradient

Opacity

`Added in v5.1.0`

As of v5.1.0, `background-color` utilities are generated with Sass using CSS variables. This allows for real-time color changes without compilation and dynamic alpha transparency changes.

How it works

Consider our default `.bg-success` utility.

```
.bg-success {
  --bs-bg-opacity: 1;
  background-color: rgba(var(--bs-success-rgb), var(--bs-bg-opacity)) !important;
```

```
}
```

We use an RGB version of our `--bs-success` (with the value of `25, 135, 84`) CSS variable and attached a second CSS variable, `--bs-bg-opacity`, for the alpha transparency (with a default value `1` thanks to a local CSS variable). That means anytime you use `.bg-success` now, your computed `color` value is `rgba(25, 135, 84, 1)`. The local CSS variable inside each `.bg-*` class avoids inheritance issues so nested instances of the utilities don't automatically have a modified alpha transparency.

Example

To change that opacity, override `--bs-bg-opacity` via custom styles or inline styles.

This is default success background

This is 50% opacity success background

```
<div class="bg-success p-2 text-white">This is default success background</div>
<div class="bg-success p-2" style="--bs-bg-opacity: .5;">This is 50% opacity success
```

Or, choose from any of the `.bg-opacity` utilities:

This is default success background

This is 75% opacity success background

This is 50% opacity success background

This is 25% opacity success background

This is 10% opacity success background

```
<div class="bg-success p-2 text-white">This is default success background</div>
<div class="bg-success p-2 text-white bg-opacity-75">This is 75% opacity success bac
<div class="bg-success p-2 text-dark bg-opacity-50">This is 50% opacity success back
<div class="bg-success p-2 text-dark bg-opacity-25">This is 25% opacity success back
<div class="bg-success p-2 text-dark bg-opacity-10">This is 10% opacity success back
```

Sass

In addition to the following Sass functionality, consider reading about our included CSS custom properties (aka CSS variables) for colors and more.

Variables

Most `background-color` utilities are generated by our theme colors, reassigned from our generic color palette variables.

```scss
$blue:    #0d6efd;
$indigo:  #6610f2;
$purple:  #6f42c1;
$pink:    #d63384;
$red:     #dc3545;
$orange:  #fd7e14;
$yellow:  #ffc107;
$green:   #198754;
$teal:    #20c997;
$cyan:    #0dcaf0;
```

```scss
$primary:       $blue;
$secondary:     $gray-600;
$success:       $green;
$info:          $cyan;
$warning:       $yellow;
$danger:        $red;
$light:         $gray-100;
$dark:          $gray-900;
```

```scss
$gradient: linear-gradient(180deg, rgba($white, .15), rgba($white, 0));
```

Grayscale colors are also available, but only a subset are used to generate any utilities.

```scss
$white:    #fff;
$gray-100: #f8f9fa;
$gray-200: #e9ecef;
$gray-300: #dee2e6;
$gray-400: #ced4da;
$gray-500: #adb5bd;
$gray-600: #6c757d;
```

```
$gray-700: #495057;
$gray-800: #343a40;
$gray-900: #212529;
$black:    #000;
```

Map

Theme colors are then put into a Sass map so we can loop over them to generate our utilities, component modifiers, and more.

```
$theme-colors: (
  "primary":    $primary,
  "secondary":  $secondary,
  "success":    $success,
  "info":       $info,
  "warning":    $warning,
  "danger":     $danger,
  "light":      $light,
  "dark":       $dark
);
```

Grayscale colors are also available as a Sass map. **This map is not used to generate any utilities.**

```
$grays: (
  "100": $gray-100,
  "200": $gray-200,
  "300": $gray-300,
  "400": $gray-400,
  "500": $gray-500,
  "600": $gray-600,
  "700": $gray-700,
  "800": $gray-800,
  "900": $gray-900
);
```

RGB colors are generated from a separate Sass map:

```
$theme-colors-rgb: map-loop($theme-colors, to-rgb, "$value");
```

And background color opacities build on that with their own map that's consumed by the utilities API:

```
$utilities-bg: map-merge(
  $utilities-colors,
  (
    "black": to-rgb($black),
    "white": to-rgb($white),
    "body": to-rgb($body-bg)
  )
);
$utilities-bg-colors: map-loop($utilities-bg, rgba-css-var, "$key", "bg");
```

Mixins

No mixins are used to generate our background utilities, but we do have some additional mixins for other situations where you'd like to create your own gradients.

```
@mixin gradient-bg($color: null) {
  background-color: $color;

  @if $enable-gradients {
    background-image: var(--#{$variable-prefix}gradient);
  }
}
```

```
// Horizontal gradient, from left to right
//
// Creates two color stops, start and end, by specifying a color and position for ea
@mixin gradient-x($start-color: $gray-700, $end-color: $gray-800, $start-percent: 0%
  background-image: linear-gradient(to right, $start-color $start-percent, $end-colo
}

// Vertical gradient, from top to bottom
//
// Creates two color stops, start and end, by specifying a color and position for ea
@mixin gradient-y($start-color: $gray-700, $end-color: $gray-800, $start-percent: nu
  background-image: linear-gradient(to bottom, $start-color $start-percent, $end-col
}

@mixin gradient-directional($start-color: $gray-700, $end-color: $gray-800, $deg: 45
  background-image: linear-gradient($deg, $start-color, $end-color);
}

@mixin gradient-x-three-colors($start-color: $blue, $mid-color: $purple, $color-stop
  background-image: linear-gradient(to right, $start-color, $mid-color $color-stop,
```

```
}

@mixin gradient-y-three-colors($start-color: $blue, $mid-color: $purple, $color-stop
  background-image: linear-gradient($start-color, $mid-color $color-stop, $end-color
}

@mixin gradient-radial($inner-color: $gray-700, $outer-color: $gray-800) {
  background-image: radial-gradient(circle, $inner-color, $outer-color);
}

@mixin gradient-striped($color: rgba($white, .15), $angle: 45deg) {
  background-image: linear-gradient($angle, $color 25%, transparent 25%, transparent
```

Utilities API

Background utilities are declared in our utilities API in `scss/_utilities.scss`. Learn how to use the utilities API.

```
  "background-color": (
    property: background-color,
    class: bg,
    local-vars: (
      "bg-opacity": 1
    ),
    values: map-merge(
      $utilities-bg-colors,
      (
        "transparent": transparent
      )
    )
  ),
  "bg-opacity": (
    css-var: true,
    class: bg-opacity,
    values: (
      10: .1,
      25: .25,
      50: .5,
      75: .75,
      100: 1
    )
  ),
```

Bootstrap

Designed and built with all the love in the world by the Bootstrap team with the help of our contributors.

Code licensed MIT, docs CC BY 3.0.

Currently v5.1.3.

Analytics by Fathom.

Links

Home

Docs

Examples

Themes

Blog

Swag Store

Guides

Getting started

Starter template

Webpack

Parcel

Projects

Bootstrap 5

Bootstrap 4

Icons

RFS

npm starter

Community

Issues

Discussions

Corporate sponsors

Open Collective

Stack Overflow