Text

Contents

Text alignment	1
Text wrapping and overflow	2
Word break	2
Text transform	2
Font size	3
Font weight and italics	3
Line height	4
Monospace	4
Reset color	4
Text decoration	4
Sass	5
Variables	5
Maps	5
Utilities API	ŏ

Text alignment

Easily realign text to components with text alignment classes. For start, end, and center alignment, responsive classes are available that use the same viewport width breakpoints as the grid system.

```
\{\{<\mathrm{example}>\}\}
```

Start aligned text on all viewport sizes.

Center aligned text on all viewport sizes.

End aligned text on all viewport sizes.

Start aligned text on viewports sized SM (small) or wider.

Start aligned text on viewports sized MD (medium) or wider.

Start aligned text on viewports sized LG (large) or wider.

Start aligned text on viewports sized XL (extra-large) or wider.

 $\{\{</\text{example}>\}\}$

 $\{\{< \text{callout info} >\}\}$ Note that we don't provide utility classes for justified text. While, aesthetically, justified text might look more appealing, it does make word-spacing more random and therefore harder to read. $\{\{</\text{callout} >\}\}$

Text wrapping and overflow

```
Wrap text with a .text-wrap class.

{{< example >}}

This text should wrap.

{{< /example >}}

Prevent text from wrapping with a .text-nowrap class.

{{< example >}}

This text should overflow the parent.

{{< /example >}}
```

Word break

Prevent long strings of text from breaking your components' layout by using .text-break to set word-wrap: break-word and word-break: break-word. We use word-wrap instead of the more common overflow-wrap for wider browser support, and add the deprecated word-break: break-word to avoid issues with flex containers.

```
\{\{\langle example \rangle\}\}
```

```
{ < / \text{example} > }
{ < \text{callout warning} > } Note that bre
```

 $\{\{< {\rm callout\; warning}>\}\}\$ Note that breaking words isn't possible in Arabic, which is the most used RTL language. Therefore .text-break is removed from our RTL compiled CSS. $\{\{</{\rm callout}>\}\}$

Text transform

Transform text in components with text capitalization classes.

```
\{\{< \text{ example } >\}\}
Lowercased text.
```

Uppercased text.

CapiTaliZed text.

Note how .text-capitalize only changes the first letter of each word, leaving the case of any other letters unaffected.

Font size

Quickly change the font-size of text. While our heading classes (e.g., .h1-.h6) apply font-size, font-weight, and line-height, these utilities *only* apply font-size. Sizing for these utilities matches HTML's heading elements, so as the number increases, their size decreases.

```
{{< example >}}
.fs-1 text
.fs-2 text
.fs-3 text
.fs-4 text
.fs-5 text
.fs-6 text
{{< /example >}}
```

Customize your available font-sizes by modifying the \$font-sizes Sass map.

Font weight and italics

Quickly change the font-weight or font-style of text with these utilities. font-style utilities are abbreviated as .fst-* and font-weight utilities are abbreviated as .fw-*.

```
{{< example >}}
Bold text.
Bolder weight text (relative to the parent element).
Normal weight text.
Light weight text.
Lighter weight text (relative to the parent element).
Italic text.
Text with normal font style
{{< /example >}}
```

Line height

Change the line height with .1h-* utilities.

```
\{\{< \text{example} > \}\}
```

This is a long paragraph written to show how the line-height of an element is affected by our utilities. Classes are applied to the element itself or sometimes the parent element. These classes can be customized as needed with our utility API.

This is a long paragraph written to show how the line-height of an element is affected by our utilities. Classes are applied to the element itself or sometimes the parent element. These classes can be customized as needed with our utility API.

This is a long paragraph written to show how the line-height of an element is affected by our utilities. Classes are applied to the element itself or sometimes the parent element. These classes can be customized as needed with our utility API.

This is a long paragraph written to show how the line-height of an element is affected by our utilities. Classes are applied to the element itself or sometimes the parent element. These classes can be customized as needed with our utility API.

```
\{\{</\text{example}>\}\}
```

Monospace

Change a selection to our monospace font stack with .font-monospace.

```
{{< example >}}
This is in monospace
{{< /example >}}
```

Reset color

Reset a text or link's color with .text-reset, so that it inherits the color from its parent.

```
{ < example > }
Muted text with a reset link.
{ < /example > }
```

Text decoration

Decorate text in components with text decoration classes.

```
\{\{\langle example \rangle\}\}
```

This text has a line underneath it.

This text has a line going through it.

This link has its text decoration removed $\{\{</\text{example}>\}\}$

Sass

Variables

```
\{\{< scss-docs\ name="font-variables"\ file="scss/\_variables.scss">\}\}
```

Maps

Font-size utilities are generated from this map, in combination with our utilities API.

```
\{ < scss-docs name="font-sizes" file="scss/_variables.scss" > \} \}
```

Utilities API

Font and text utilities are declared in our utilities API in scss/_utilities.scss. [Learn how to use the utilities API.]({{< docsref "/utilities/api#using-the-api">>}})

```
\{\{<\tt scss-docs\ name="utils-text"\ file="scss/\_utilities.scss">\}\}
```