View on GitHub

Alerts

Provide contextual feedback messages for typical user actions with the handful of available and flexible alert messages.

On this page ⇕

Examples

Alerts are available for any length of text, as well as an optional close button. For proper styling, use one of the eight **required** contextual classes (e.g., `.alert-success`). For inline dismissal, use the [alerts JavaScript plugin](#).

> **Heads up!** As of v5.3.0, the `alert-variant()` Sass mixin is deprecated. Alert variants now

A simple primary alert—check it out!

A simple secondary alert—check it out!

A simple success alert—check it out!

A simple danger alert—check it out!

A simple warning alert—check it out!

A simple info alert—check it out!

A simple light alert—check it out!

A simple dark alert—check it out!

HTML

```html
<div class="alert alert-primary" role="alert">
  A simple primary alert—check it out!
</div>
<div class="alert alert-secondary" role="alert">
  A simple secondary alert—check it out!
</div>
<div class="alert alert-success" role="alert">
  A simple success alert—check it out!
</div>
<div class="alert alert-danger" role="alert">
  A simple danger alert—check it out!
</div>
<div class="alert alert-warning" role="alert">
  A simple warning alert—check it out!
</div>
<div class="alert alert-info" role="alert">
  A simple info alert—check it out!
</div>
<div class="alert alert-light" role="alert">
  A simple light alert—check it out!
</div>
<div class="alert alert-dark" role="alert">
```

```
  A simple dark alert—check it out!
</div>
```

> **Accessibility tip:** Using color to add meaning only provides a visual indication, which will not be conveyed to users of assistive technologies like screen readers. Please ensure the meaning is obvious from the content itself (e.g., the visible text) or is included through alternative means, such as additional text hidden with the `.visually-hidden` class.

Live example

Click the button below to show an alert (hidden with inline styles to start), then dismiss (and destroy) it with the built-in close button.

Show live alert

HTML

```
<div id="liveAlertPlaceholder"></div>
<button type="button" class="btn btn-primary" id="liveAlertBtn">Show live alert</butto
```

We use the following JavaScript to trigger our live alert demo:

```
const alertPlaceholder = document.getElementById('liveAlertPlaceholder')

const alert = (message, type) => {
  const wrapper = document.createElement('div')
  wrapper.innerHTML = [
    `<div class="alert alert-${type} alert-dismissible" role="alert">`,
    `   <div>${message}</div>`,
    '   <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Cl
    '</div>'
  ].join('')

  alertPlaceholder.append(wrapper)
}

const alertTrigger = document.getElementById('liveAlertBtn')
```

```
if (alertTrigger) {
  alertTrigger.addEventListener('click', () => {
    alert('Nice, you triggered this alert message!', 'success')
  })
}
```

Link color

Use the `.alert-link` utility class to quickly provide matching colored links within any alert.

A simple primary alert with **an example link**. Give it a click if you like.

A simple secondary alert with **an example link**. Give it a click if you like.

A simple success alert with **an example link**. Give it a click if you like.

A simple danger alert with **an example link**. Give it a click if you like.

A simple warning alert with **an example link**. Give it a click if you like.

A simple info alert with **an example link**. Give it a click if you like.

A simple light alert with **an example link**. Give it a click if you like.

A simple dark alert with **an example link**. Give it a click if you like.

HTML

```
<div class="alert alert-primary" role="alert">
  A simple primary alert with <a href="#" class="alert-link">an example link</a>. Give
```

```
</div>
<div class="alert alert-secondary" role="alert">
  A simple secondary alert with <a href="#" class="alert-link">an example link</a>. Gi
</div>
<div class="alert alert-success" role="alert">
  A simple success alert with <a href="#" class="alert-link">an example link</a>. Give
</div>
<div class="alert alert-danger" role="alert">
  A simple danger alert with <a href="#" class="alert-link">an example link</a>. Give
</div>
<div class="alert alert-warning" role="alert">
  A simple warning alert with <a href="#" class="alert-link">an example link</a>. Give
</div>
<div class="alert alert-info" role="alert">
  A simple info alert with <a href="#" class="alert-link">an example link</a>. Give it
</div>
<div class="alert alert-light" role="alert">
  A simple light alert with <a href="#" class="alert-link">an example link</a>. Give i
</div>
<div class="alert alert-dark" role="alert">
  A simple dark alert with <a href="#" class="alert-link">an example link</a>. Give it
</div>
```

Additional content

Alerts can also contain additional HTML elements like headings, paragraphs and dividers.

Well done!

Aww yeah, you successfully read this important alert message. This example text is going to run a bit longer so that you can see how spacing within an alert works with this kind of content.

Whenever you need to, be sure to use margin utilities to keep things nice and tidy.

HTML

```
<div class="alert alert-success" role="alert">
  <h4 class="alert-heading">Well done!</h4>
  <p>Aww yeah, you successfully read this important alert message. This example text i
  <hr>
  <p class="mb-0">Whenever you need to, be sure to use margin utilities to keep things
</div>
```

Icons

[illegible] ng on your icons and content, you may want to add more utilities or custom styles.

⚠ An example alert with an icon

HTML

```
<div class="alert alert-primary d-flex align-items-center" role="alert">
  <svg xmlns="http://www.w3.org/2000/svg" class="bi bi-exclamation-triangle-fill flex-
    <path d="M8.982 1.566a1.13 1.13 0 0 0-1.96 0L.165 13.233c-.457.778.091 1.767.98 1.
  </svg>
  <div>
    An example alert with an icon
  </div>
</div>
```

Need more than one icon for your alerts? Consider using more Bootstrap Icons and making a local SVG sprite like so to easily reference the same icons repeatedly.

ⓘ An example alert with an icon

✓ An example success alert with an icon

⚠ An example warning alert with an icon

⚠ An example danger alert with an icon

HTML

```
<svg xmlns="http://www.w3.org/2000/svg" style="display: none;">
  <symbol id="check-circle-fill" viewBox="0 0 16 16">
    <path d="M16 8A8 8 0 1 1 0 8a8 8 0 0 1 16 0zm-3.97-3.03a.75.75 0 0 0-1.08.022L7.47
  </symbol>
  <symbol id="info-fill" viewBox="0 0 16 16">
    <path d="M8 16A8 8 0 1 0 8 0a8 8 0 0 0 0 16zm.93-9.412-1 4.705c-.07.34.029.533.304
  </symbol>
  <symbol id="exclamation-triangle-fill" viewBox="0 0 16 16">
    <path d="M8.982 1.566a1.13 1.13 0 0 0-1.96 0L.165 13.233c-.457.778.091 1.767.98 1.
  </symbol>
</svg>

<div class="alert alert-primary d-flex align-items-center" role="alert">
  <svg class="bi flex-shrink-0 me-2" role="img" aria-label="Info:"><use xlink:href="#i
  <div>
    An example alert with an icon
  </div>
</div>
<div class="alert alert-success d-flex align-items-center" role="alert">
  <svg class="bi flex-shrink-0 me-2" role="img" aria-label="Success:"><use xlink:href=
  <div>
    An example success alert with an icon
  </div>
</div>
<div class="alert alert-warning d-flex align-items-center" role="alert">
  <svg class="bi flex-shrink-0 me-2" role="img" aria-label="Warning:"><use xlink:href=
  <div>
    An example warning alert with an icon
  </div>
</div>
<div class="alert alert-danger d-flex align-items-center" role="alert">
  <svg class="bi flex-shrink-0 me-2" role="img" aria-label="Danger:"><use xlink:href="
  <div>
    An example danger alert with an icon
  </div>
</div>
```

Dismissing

Using the alert JavaScript plugin, it’s possible to dismiss any alert inline. Here’s how:

- Be sure you’ve loaded the alert plugin, or the compiled Bootstrap JavaScript.
- Add a close button and the `.alert-dismissible` class, which adds extra padding to the right of the alert and positions the close button.
- On the close button, add the `data-bs-dismiss="alert"` attribute, which triggers the JavaScript functionality. Be sure to use the `<button>` element with it for proper behavior across all devices.
- To animate alerts when dismissing them, be sure to add the `.fade` and `.show` classes.

You can see this in action with a live demo:

Holy guacamole! You should check in on some of those fields below.

HTML

```html
<div class="alert alert-warning alert-dismissible fade show" role="alert">
  <strong>Holy guacamole!</strong> You should check in on some of those fields below.
  <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"><
</div>
```

> When an alert is dismissed, the element is completely removed from the page structure. If a keyboard user dismisses the alert using the close button, their focus will suddenly be lost and, depending on the browser, reset to the start of the page/document. For this reason, we recommend including additional JavaScript that listens for the `closed.bs.alert` event and programmatically sets `focus()` to the most appropriate location in the page. If you’re planning to move focus to a non-interactive element that normally does not receive focus, make sure to add `tabindex="-1"` to the element.

CSS

Variables

Added in v5.2.0

As part of Bootstrap’s evolving CSS variables approach, alerts now use local CSS variables on `.alert` for enhanced real-time customization. Values for the CSS variables are set via Sass, so Sass customization is still supported, too.

scss/_alert.scss

```
--#{$prefix}alert-bg: transparent;
--#{$prefix}alert-padding-x: #{$alert-padding-x};
--#{$prefix}alert-padding-y: #{$alert-padding-y};
--#{$prefix}alert-margin-bottom: #{$alert-margin-bottom};
--#{$prefix}alert-color: inherit;
--#{$prefix}alert-border-color: transparent;
--#{$prefix}alert-border: #{$alert-border-width} solid var(--#{$prefix}alert-border-co
--#{$prefix}alert-border-radius: #{$alert-border-radius};
--#{$prefix}alert-link-color: inherit;
```

Sass variables

scss/_variables.scss

```
$alert-padding-y:               $spacer;
$alert-padding-x:               $spacer;
$alert-margin-bottom:           1rem;
$alert-border-radius:           $border-radius;
$alert-link-font-weight:        $font-weight-bold;
$alert-border-width:            var(--#{$prefix}border-width);
$alert-bg-scale:                -80%;
$alert-border-scale:            -70%;
$alert-color-scale:             40%;
$alert-dismissible-padding-r:   $alert-padding-x * 3; // 3x covers width of x plus def
```

Sass mixin

Deprecated in v5.3.0

Used in combination with `$theme-colors` to create contextual modifier classes for our alerts.

scss/mixins/_alert.scss

```
@mixin alert-variant($background, $border, $color) {
  --#{$prefix}alert-color: #{$color};
  --#{$prefix}alert-bg: #{$background};
  --#{$prefix}alert-border-color: #{$border};
  --#{$prefix}alert-link-color: #{shade-color($color, 20%)};

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

  .alert-link {
    color: var(--#{$prefix}alert-link-color);
  }
}
```

Sass loop

Loop that generates the modifier classes with the `alert-variant()` mixin.

scss/_alert.scss

```
// Generate contextual modifier classes for colorizing the alert
@each $state in map-keys($theme-colors) {
  .alert-#{$state} {
    --#{$prefix}alert-color: var(--#{$prefix}#{$state}-text-emphasis);
    --#{$prefix}alert-bg: var(--#{$prefix}#{$state}-bg-subtle);
    --#{$prefix}alert-border-color: var(--#{$prefix}#{$state}-border-subtle);
    --#{$prefix}alert-link-color: var(--#{$prefix}#{$state}-text-emphasis);
  }
}
```

JavaScript behavior

Initialize

Initialize elements as alerts

```
const alertList = document.querySelectorAll('.alert')
const alerts = [...alertList].map(element => new bootstrap.Alert(element))
```

> For the sole purpose of dismissing an alert, it isn't necessary to initialize the component manually via the JS API. By making use of `data-bs-dismiss="alert"`, the component will be initialized automatically and properly dismissed.
>
> See the triggers section for more details.

Triggers

Dismissal can be achieved with the `data-bs-dismiss` attribute on a button **within the alert** as demonstrated below:

```
<button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></b
```

or on a button **outside the alert** using the additional `data-bs-target` as demonstrated below:

```
<button type="button" class="btn-close" data-bs-dismiss="alert" data-bs-target="#my-al
```

Note that closing an alert will remove it from the DOM.

Methods

You can create an alert instance with the alert constructor, for example:

```
const bsAlert = new bootstrap.Alert('#myAlert')
```

This makes an alert listen for click events on descendant elements which have the `data-bs-dismiss="alert"` attribute. (Not necessary when using the data-api's auto-initialization.)

Method	Description
`close`	Closes an alert by removing it from the DOM. If the `.fade` and `.show` classes are present on the element, the alert will fade out before it is removed.
`dispose`	Destroys an element's alert. (Removes stored data on the DOM element)
`getInstance`	Static method which allows you to get the alert instance associated to a DOM element. For example: `bootstrap.Alert.getInstance(alert)`.
`getOrCreateInstance`	Static method which returns an alert instance associated to a DOM element or create a new one in case it wasn't initialized. You can use it like this: `bootstrap.Alert.getOrCreateInstance(element)`.

Basic usage:

```
const alert = bootstrap.Alert.getOrCreateInstance('#myAlert')
alert.close()
```

Events

Bootstrap's alert plugin exposes a few events for hooking into alert functionality.

Event	Description
`close.bs.alert`	Fires immediately when the `close` instance method is called.
`closed.bs.alert`	Fired when the alert has been closed and CSS transitions have completed.

```
const myAlert = document.getElementById('myAlert')
myAlert.addEventListener('closed.bs.alert', event => {
  // do something, for instance, explicitly move focus to the most appropriate element
  // so it doesn't get lost/reset to the start of the page
  // document.getElementById('...').focus()
})
```

Bootstrap

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

Currently v5.3.0-alpha2.

Links

- Home
- Docs
- Examples
- Icons
- Themes
- Blog
- Swag Store

Guides

- Getting started
- Starter template
- Webpack
- Parcel
- Vite

Projects

- Bootstrap 5
- Bootstrap 4
- Icons
- RFS
- Examples repo

Community

- Issues
- Discussions
- Corporate sponsors
- Open Collective
- Stack Overflow