

Components

Over a dozen reusable components built to provide iconography, dropdowns, input groups, navigation, alerts, and much more.

Glyphicons

Available glyphs

Includes over 250 glyphs in font format from the Glyphicon Halflings set. Glyphicons (<http://glyphicons.com/>) Halflings are normally not available for free, but their creator has made them available for Bootstrap free of cost. As a thank you, we only ask that you include a link back to Glyphicons (<http://glyphicons.com/>) whenever possible.



glyphicon
glyphicon-
asterisk



glyphicon
glyphicon-pl
us



glyphicon
glyphicon-eu
ro



glyphicon
glyphicon-eu
r



glyphicon
glyphicon-mi
nus



glyphicon
glyphicon-cl
oud



glyphicon
glyphicon-
envelope



glyphicon
glyphicon-
pencil



glyphicon
glyphicon-gl
ass



glyphicon
glyphicon-m
usic



glyphicon
glyphicon-
search



glyphicon
glyphicon-he
art



glyphicon
glyphicon-st
ar



glyphicon
glyphicon-
star-empty



glyphicon
glyphicon-us
er



glyphicon
glyphicon-fil
m



glyphicon
glyphicon-
th-large



glyphicon
glyphicon-th



glyphicon
glyphicon-
th-list



glyphicon
glyphicon-ok



glyphicon
glyphicon-
remove



glyphicon
glyphicon-
zoom-in



glyphicon
glyphicon-
zoom-out



glyphicon
glyphicon-off



glyphicon
glyphicon-
signal



glyphicon
glyphicon-co
g



glyphicon
glyphicon-tra
sh



glyphicon
glyphicon-ho
me



glyphicon
glyphicon-fil
e



glyphicon
glyphicon-ti
me



glyphicon
glyphicon-ro
ad



glyphicon
glyphicon-
download-alt



glyphicon
glyphicon-
download



glyphicon
glyphicon-
upload



glyphicon
glyphicon-in
box



glyphicon
glyphicon-
play-circle



glyphicon
glyphicon-
repeat



glyphicon
glyphicon-
refresh



glyphicon
glyphicon-
list-alt



glyphicon
glyphicon-lo
ck



glyphicon



glyphicon



glyphicon



glyphicon



glyphicon



glyphicon



glyphicon



glyphicon

How to use

For performance reasons, all icons require a base class and individual icon class. To use, place the following code just about anywhere – be sure to leave a space between the icon and text for proper padding.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-ta	glyphicon-bo	glyphicon-bookmark	glyphicon-print	glyphicon-camera	glyphicon-font	glyphicon-bolt	glyphicon-italic

Don't mix with other components

Icon classes cannot be directly combined with other components. They should not be used along with other classes on the same element. Instead, add a nested `` and apply the icon classes to the ``.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-text-height	glyphicon-text-width	glyphicon-align-left	glyphicon-align-center	glyphicon-align-right	glyphicon-align-justify	glyphicon-list	glyphicon-indent-left

Only for use on empty elements

Icon classes should only be used on elements that contain no text content and have no child elements.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-indent-left	glyphicon-indent-right	glyphicon-picture	glyphicon-map-marker	glyphicon-adjust	glyphicon-tint	glyphicon-edit	glyphicon-share

Changing the icon font location

Bootstrap assumes icon font files will be located in the `../fonts/` directory, relative to the compiled CSS files. Moving or renaming those font files means updating the CSS in one of three ways:

- Change the `@icon-font-path` and/or `@icon-font-name` variables in the source Less files.
- Utilize the relative URL's option (<http://lesscss.org/usage/#command-line-usage-relative-urls>) provided by the Less compiler.
- Change the `url()` paths in the compiled CSS.

Use whatever option best suits your specific development setup.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-forward	glyphicon-fast-forward	glyphicon-step-forward	glyphicon-ect	glyphicon-chevron-left	glyphicon-chevron-right	glyphicon-plus-sign	glyphicon-minus-sign

Accessible icons

Modern versions of assistive technologies will announce CSS generated content, as well as specific Unicode characters. To avoid unintended and confusing output in screen readers (particularly when icons are used purely for decoration), we hide them with the `aria-hidden="true"` attribute.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-remove-sign	glyphicon-ok-sign	glyphicon-question-sign	glyphicon-info-sign	glyphicon-screenshot	glyphicon-remove-circle	glyphicon-ok-circle	glyphicon-ban-circle

If you're using an icon to convey meaning (rather than only as a decorative element), ensure that this meaning is also conveyed to assistive technologies – for instance, include additional content, visually hidden with the `.sr-only` class.

If you're creating controls with no other text (such as a `<button>` that only contains an icon), you should always provide alternative content to identify the purpose of the control, so that it will make sense to users of assistive technologies. In this case, you could add an `aria-label` attribute to the control itself.

glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-arrow-left	glyphicon-arrow-right	glyphicon-arrow-up	glyphicon-arrow-down	glyphicon-share-alt	glyphicon-resize-full	glyphicon-resize-small	glyphicon-exclamation-sign

<code></code>	<code></code>	<code></code>	<code></code>	<code></code>	<code></code>	<code></code>	<code></code>
--	---	--	---	--	---	--	---

Examples

Use them in buttons, button groups for a toolbar, navigation, or prepended form inputs.

EXAMPLE	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon
glyphicon-random	glyphicon-comment	glyphicon-magnet	glyphicon-chevron-up	glyphicon-chevron-down	glyphicon-retweet	glyphicon-shopping-cart	glyphicon-folder-close
Star	Star	Star	Star	Star	Star	Star	Star
glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon	glyphicon

Copy

<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>	<code>glyphicon- button-type="button" class="btn btn-default" aria-label="Left Align"> </button></code>
<code>glyphicon- thumbs-down</code>	<code>glyphicon- hand-right</code>	<code>glyphicon- hand-left</code>	<code>glyphicon- hand-up</code>	<code>glyphicon- hand-down</code>	<code>glyphicon- circle- arrow-right</code>	<code>glyphicon- circle- arrow-left</code>	<code>glyphicon- circle- arrow-up</code>

An icon used in an alert to convey that it's an error message, with additional text to convey this information to users of assistive technologies.

<code>glyphicon- circle- arrow-down</code>	<code>glyphicon- heart-empty</code>	<code>glyphicon- wrench</code>	<code>glyphicon- tag</code>	<code>glyphicon- filter</code>	<code>glyphicon- briefcase</code>	<code>glyphicon- fullscreen</code>	<code>glyphicon- dashboard</code>
<code>glyphicon- paperclip</code>	<code>glyphicon- heart-empty</code>	<code>glyphicon- link</code>	<code>glyphicon- phone</code>	<code>glyphicon- pushpin</code>	<code>glyphicon- us-dollar</code>	<code>glyphicon- gb-pound</code>	<code>glyphicon- copy</code>
<code>glyphicon- sort-by- alphabet</code>	<code>glyphicon- sort-by- alphabet-alt</code>	<code>glyphicon- sort-by-order</code>	<code>glyphicon- sort-by- order-alt</code>	<code>glyphicon- sort-by- attributes</code>	<code>glyphicon- sort-by- attributes-alt</code>	<code>glyphicon- unchecked</code>	<code>glyphicon- expand</code>

Dropdowns

Toggleable, contextual menu for displaying lists of links. Made interactive with the `dropdown.js` JavaScript plugin (`$(document).ready(function() {`).

Example

Wrap the dropdown's trigger and the dropdown menu within `.dropdown`, or another element that declares `position: relative;`. Then add the menu's HTML. Dropdown menus can be changed to expand upwards (instead of downwards) by adding `.dropup` to the parent.

<code>glyphicon- floppy-save</code>	<code>glyphicon- floppy-open</code>	<code>glyphicon- credit-card</code>	<code>glyphicon- transfer</code>	<code>glyphicon- cutlery</code>	<code>glyphicon- header</code>	<code>glyphicon- compressed</code>	<code>glyphicon- earphone</code>
<code>glyphicon- phone</code>	<code>glyphicon- phone</code>	<code>glyphicon- stats</code>	<code>glyphicon- sd-video</code>	<code>glyphicon- hd-video</code>	<code>glyphicon- subtitles</code>	<code>glyphicon- sound- stereo</code>	<code>glyphicon- sound-dolby</code>
<code>glyphicon- sound-5-1</code>	<code>glyphicon- sound-6-1</code>	<code>glyphicon- sound-7-1</code>	<code>glyphicon- copyright- mark</code>	<code>glyphicon- registration- mark</code>	<code>glyphicon- cloud- download</code>	<code>glyphicon- cloud- upload</code>	<code>glyphicon- tree-conifer</code>
<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>	<code>glyphicon- cloud</code>

Alignment

By default, a dropdown menu is automatically positioned 100% from the top and along the left side of its parent. Add

object-align- dropdown-menu-right	object-align- horizontal	object-align- vertical	object-align- triangle-right	object-align- triangle-left	object-align- triangle-bottom	object-align- triangle-top
align-right to a .dropdown-menu to right align the dropdown menu.	align-left		align-right			

$\geq T^2$ T_2
May require additional positioning

Dropdowns are automatically positioned via CSS within the normal flow of the document. This means dropdowns may be cropped by parents with certain overflow properties or appear out-of-bounds of the viewport. Address these issues on your own as they arise.

Deprecated .pull-right alignment

As of v3.1.0, we've deprecated `.pull-right` on dropdown menus. To right-align a menu, use `.dropdown-menu-right`. Right-aligned nav components in the navbar use a mixin version of this class to automatically align the menu. To override it, use `.dropdown-menu-left`.

```
<ul class="dropdown-menu dropdown-menu-right" role="menu" aria-labelledby="dLabel">
  ...
</ul>
```

Copy

Headers

Add a header to label sections of actions in any dropdown menu.

EXAMPLE

Dropdown ▾

Dropdown header
Action
Another action
Something else here
Dropdown header
Separated link

```
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu3">
  ...
  <li role="presentation" class="dropdown-header">Dropdown header</li>
  ...
</ul>
```

Copy

Divider

Add a divider to separate series of links in a dropdown menu.

EXAMPLE

Dropdown ▾

Action
Another action
Something else here

Separated link

```
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenuDivider">
  ...
  <li role="presentation" class="divider"></li>
  ...
</ul>
```

Copy

Disabled menu items

Add `.disabled` to a `` in the dropdown to disable the link.

EXAMPLE

Dropdown ▾

Regular link
Disabled link
Another link

Copy

```
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu4">
  <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Regular link</a></li>
  <li role="presentation" class="disabled"><a role="menuitem" tabindex="-1"
href="#">Disabled link</a></li>
  <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Another link</a></li>
</ul>
```

Button groups

Group a series of buttons together on a single line with the button group. Add on optional JavaScript radio and checkbox style behavior with our buttons plugin ([../javascript/#buttons](#)).

Tooltips & popovers in button groups require special setting

When using tooltips or popovers on elements within a `.btn-group`, you'll have to specify the option `container: 'body'` to avoid unwanted side effects (such as the element growing wider and/or losing its rounded corners when the tooltip or popover is triggered).

Ensure correct `role` and provide a label

In order for assistive technologies – such as screen readers – to convey that a series of buttons is grouped, an appropriate `role` attribute needs to be provided. For button groups, this would be `role="group"`, while toolbars should have a `role="toolbar"`.

One exception are groups which only contain a single control (for instance the justified button groups with `<button>` elements) or a dropdown.

In addition, groups and toolbars should be given an explicit label, as most assistive technologies will otherwise not announce them, despite the presence of the correct `role` attribute. In the examples provided here, we use `aria-label`, but alternatives such as `aria-labelledby` can also be used.

Basic example

Wrap a series of buttons with `.btn` in `.btn-group`.

EXAMPLE



```
<div class="btn-group" role="group" aria-label="...">
  <button type="button" class="btn btn-default">Left</button>
  <button type="button" class="btn btn-default">Middle</button>
  <button type="button" class="btn btn-default">Right</button>
</div>
```

[Copy](#)

Button toolbar

Combine sets of `<div class="btn-group">` into a `<div class="btn-toolbar">` for more complex components.

EXAMPLE



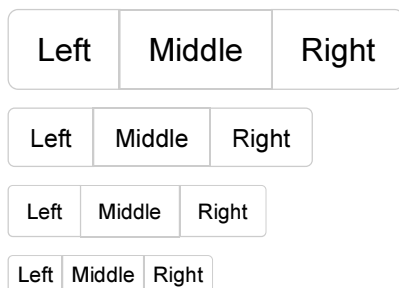
```
<div class="btn-toolbar" role="toolbar" aria-label="...">
  <div class="btn-group" role="group" aria-label="...">...</div>
  <div class="btn-group" role="group" aria-label="...">...</div>
  <div class="btn-group" role="group" aria-label="...">...</div>
</div>
```

[Copy](#)

Sizing

Instead of applying button sizing classes to every button in a group, just add `.btn-group-*` to each `.btn-group`, including when nesting multiple groups.

EXAMPLE



```
<div class="btn-group btn-group-lg" role="group" aria-label="...">...</div>
<div class="btn-group" role="group" aria-label="...">...</div>
<div class="btn-group btn-group-sm" role="group" aria-label="...">...</div>
<div class="btn-group btn-group-xs" role="group" aria-label="...">...</div>
```

[Copy](#)

Nesting

Place a `.btn-group` within another `.btn-group` when you want dropdown menus mixed with a series of buttons.

EXAMPLE


[Copy](#)

```

<div class="btn-group" role="group" aria-label="...">
  <button type="button" class="btn btn-default">1</button>
  <button type="button" class="btn btn-default">2</button>

  <div class="btn-group" role="group">
    <button type="button" class="btn btn-default dropdown-toggle" data-toggle="dropdown"
    aria-expanded="false">
      Dropdown
      <span class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
      <li><a href="#">Dropdown link</a></li>
      <li><a href="#">Dropdown link</a></li>
    </ul>
  </div>
</div>

```

Vertical variation

Make a set of buttons appear vertically stacked rather than horizontally. **Split button dropdowns are not supported here.**

EXAMPLE



```

<div class="btn-group-vertical" role="group" aria-label="...">
  ...
</div>

```

[Copy](#)

Justified button groups

Make a group of buttons stretch at equal sizes to span the entire width of its parent. Also works with button dropdowns within the button group.

Handling borders

Due to the specific HTML and CSS used to justify buttons (namely `display: table-cell`), the borders between them are doubled. In regular button groups, `margin-left: -1px` is used to stack the borders instead of removing them. However, `margin` doesn't work with `display: table-cell`. As a result, depending on your customizations to Bootstrap, you may wish to remove or re-color the borders.

IE8 and borders

Internet Explorer 8 doesn't render borders on buttons in a justified button group, whether it's on `<a>` or `<button>` elements. To get around that, wrap each button in another `.btn-group`.

See #12476 (<https://github.com/twbs/bootstrap/issues/12476>) for more information.

With `<a>` elements

Just wrap a series of `.btn` s in `.btn-group.btn-group-justified`.

EXAMPLE

Left	Middle	Right
Left	Middle	Dropdown ▼

```
<div class="btn-group btn-group-justified" role="group" aria-label="...">
  ...
</div>
```

[Copy](#)

Links acting as buttons

If the `<a>` elements are used to act as buttons – triggering in-page functionality, rather than navigating to another document or section within the current page – they should also be given an appropriate `role="button"`.

With `<button>` elements

To use justified button groups with `<button>` elements, **you must wrap each button in a button group**. Most browsers don't properly apply our CSS for justification to `<button>` elements, but since we support button dropdowns, we can work around that.

EXAMPLE

Left	Middle	Right
------	--------	-------

```
<div class="btn-group btn-group-justified" role="group" aria-label="...">
  <div class="btn-group" role="group">
    <button type="button" class="btn btn-default">Left</button>
  </div>
  <div class="btn-group" role="group">
    <button type="button" class="btn btn-default">Middle</button>
  </div>
  <div class="btn-group" role="group">
    <button type="button" class="btn btn-default">Right</button>
  </div>
</div>
```

[Copy](#)

Button dropdowns

Use any button to trigger a dropdown menu by placing it within a `.btn-group` and providing the proper menu markup.

Plugin dependency

Button dropdowns require the dropdown plugin (`../javascript/#dropdowns`) to be included in your version of Bootstrap.

Single button dropdowns

Turn a button into a dropdown toggle with some basic markup changes.

EXAMPLE



```
<!-- Single button -->
<div class="btn-group">
  <button type="button" class="btn btn-default dropdown-toggle" data-toggle="dropdown"
  aria-expanded="false">
    Action <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu">
    <li><a href="#">Action</a></li>
    <li><a href="#">Another action</a></li>
    <li><a href="#">Something else here</a></li>
    <li class="divider"></li>
    <li><a href="#">Separated link</a></li>
  </ul>
</div>
```

[Copy](#)

Split button dropdowns

Similarly, create split button dropdowns with the same markup changes, only with a separate button.

EXAMPLE

[Copy](#)

```

<!-- Split button -->
<div class="btn-group">
  <button type="button" class="btn btn-danger">Action</button>
  <button type="button" class="btn btn-danger dropdown-toggle" data-toggle="dropdown"
aria-expanded="false">
    <span class="caret"></span>
    <span class="sr-only">Toggle Dropdown</span>
  </button>
  <ul class="dropdown-menu" role="menu">
    <li><a href="#">Action</a></li>
    <li><a href="#">Another action</a></li>
    <li><a href="#">Something else here</a></li>
    <li class="divider"></li>
    <li><a href="#">Separated link</a></li>
  </ul>
</div>

```

Sizing

Button dropdowns work with buttons of all sizes.

EXAMPLE

Large button ▼

Small button ▼

Extra small button ▼

Copy

```

<!-- Large button group -->
<div class="btn-group">
  <button class="btn btn-default btn-lg dropdown-toggle" type="button"
data-toggle="dropdown" aria-expanded="false">
    Large button <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu">
    ...
  </ul>
</div>

<!-- Small button group -->
<div class="btn-group">
  <button class="btn btn-default btn-sm dropdown-toggle" type="button"
data-toggle="dropdown" aria-expanded="false">
    Small button <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu">
    ...
  </ul>
</div>

<!-- Extra small button group -->
<div class="btn-group">
  <button class="btn btn-default btn-xs dropdown-toggle" type="button"
data-toggle="dropdown" aria-expanded="false">
    Extra small button <span class="caret"></span>
  </button>
  <ul class="dropdown-menu" role="menu">
    ...
  </ul>
</div>

```

Dropup variation

Trigger dropdown menus above elements by adding `.dropup` to the parent.

EXAMPLE



```

<div class="btn-group dropup">
  <button type="button" class="btn btn-default">Dropup</button>
  <button type="button" class="btn btn-default dropdown-toggle" data-toggle="dropdown"
aria-expanded="false">
    <span class="caret"></span>
    <span class="sr-only">Toggle Dropdown</span>
  </button>
  <ul class="dropdown-menu" role="menu">
    <!-- Dropdown menu links -->
  </ul>
</div>

```

[Copy](#)

Input groups

Extend form controls by adding text or buttons before, after, or on both sides of any text-based `<input>`. Use `.input-group` with an `.input-group-addon` to prepend or append elements to a single `.form-control`.

Textual `<input>`s only

Avoid using `<select>` elements here as they cannot be fully styled in WebKit browsers.

Avoid using `<textarea>` elements here as their `rows` attribute will not be respected in some cases.

Tooltips & popovers in input groups require special setting

When using tooltips or popovers on elements within an `.input-group`, you'll have to specify the option `container: 'body'` to avoid unwanted side effects (such as the element growing wider and/or losing its rounded corners when the tooltip or popover is triggered).

Don't mix with other components

Do not mix form groups or grid column classes directly with input groups. Instead, nest the input group inside of the form group or grid-related element.

Always add labels

Screen readers will have trouble with your forms if you don't include a label for every input. For these input groups, ensure that any additional label or functionality is conveyed to assistive technologies.

The exact technique to be used (`<label>` elements hidden using the `.sr-only` class, or use of the `aria-label`, `aria-labelledby`, `aria-describedby`, `title` or `placeholder` attribute) and what additional information will need to be conveyed will vary depending on the exact type of interface widget you're implementing. The examples in this section provide a few suggested, case-specific approaches.

Basic example

Place one add-on or button on either side of an input. You may also place one on both sides of an input.

We do not support multiple add-ons on a single side.

We do not support multiple form-controls in a single input group.

EXAMPLE

@	Username	
Recipient's username		@example.com
\$.00

```

<div class="input-group">
  <span class="input-group-addon" id="basic-addon1">@</span>
  <input type="text" class="form-control" placeholder="Username" aria-describedby="basic-addon1">
</div>

<div class="input-group">
  <input type="text" class="form-control" placeholder="Recipient's username"
  aria-describedby="basic-addon2">
  <span class="input-group-addon" id="basic-addon2">@example.com</span>
</div>

<div class="input-group">
  <span class="input-group-addon">$</span>
  <input type="text" class="form-control" aria-label="Amount (to the nearest dollar)">
  <span class="input-group-addon">.00</span>
</div>

```

Copy

Sizing

Add the relative form sizing classes to the `.input-group` itself and contents within will automatically resize—no need for repeating the form control size classes on each element.

EXAMPLE

@	Username	
@	Username	
@	Username	

Copy

```

<div class="input-group input-group-lg">
  <span class="input-group-addon" id="sizing-addon1">@</span>
  <input type="text" class="form-control" placeholder="Username" aria-describedby="sizing-addon1">
</div>

<div class="input-group">
  <span class="input-group-addon" id="sizing-addon2">@</span>
  <input type="text" class="form-control" placeholder="Username" aria-describedby="sizing-addon2">
</div>

<div class="input-group input-group-sm">
  <span class="input-group-addon" id="sizing-addon3">@</span>
  <input type="text" class="form-control" placeholder="Username" aria-describedby="sizing-addon3">
</div>

```

Checkboxes and radio addons

Place any checkbox or radio option within an input group's addon instead of text.

EXAMPLE

<input type="checkbox"/>	
<input type="radio"/>	

```

<div class="row">
  <div class="col-lg-6">
    <div class="input-group">
      <span class="input-group-addon">
        <input type="checkbox" aria-label="...">
      </span>
      <input type="text" class="form-control" aria-label="...">
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
  <div class="col-lg-6">
    <div class="input-group">
      <span class="input-group-addon">
        <input type="radio" aria-label="...">
      </span>
      <input type="text" class="form-control" aria-label="...">
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
</div><!-- /.row -->

```

[Copy](#)

Button addons

Buttons in input groups are a bit different and require one extra level of nesting. Instead of `.input-group-addon`, you'll need to use `.input-group-btn` to wrap the buttons. This is required due to default browser styles that cannot be overridden.

EXAMPLE

Go!	Search for...
-----	---------------

Search for...

Go!

```

<div class="row">
  <div class="col-lg-6">
    <div class="input-group">
      <span class="input-group-btn">
        <button class="btn btn-default" type="button">Go!</button>
      </span>
      <input type="text" class="form-control" placeholder="Search for...">
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
  <div class="col-lg-6">
    <div class="input-group">
      <input type="text" class="form-control" placeholder="Search for...">
      <span class="input-group-btn">
        <button class="btn btn-default" type="button">Go!</button>
      </span>
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
</div><!-- /.row -->

```

Copy

Buttons with dropdowns

EXAMPLE

Action ▼

Action ▼

Copy


```

<div class="row">
  <div class="col-lg-6">
    <div class="input-group">
      <div class="input-group-btn">
        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown" aria-expanded="false">Action <span class="caret"></span></button>
        <ul class="dropdown-menu" role="menu">
          <li><a href="#">Action</a></li>
          <li><a href="#">Another action</a></li>
          <li><a href="#">Something else here</a></li>
          <li class="divider"></li>
          <li><a href="#">Separated link</a></li>
        </ul>
      </div><!-- /btn-group -->
      <input type="text" class="form-control" aria-label="...">
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
  <div class="col-lg-6">
    <div class="input-group">
      <input type="text" class="form-control" aria-label="...">
      <div class="input-group-btn">
        <button type="button" class="btn btn-default dropdown-toggle"
data-toggle="dropdown" aria-expanded="false">Action <span class="caret"></span></button>
        <ul class="dropdown-menu dropdown-menu-right" role="menu">
          <li><a href="#">Action</a></li>
          <li><a href="#">Another action</a></li>
          <li><a href="#">Something else here</a></li>
          <li class="divider"></li>
          <li><a href="#">Separated link</a></li>
        </ul>
      </div><!-- /btn-group -->
    </div><!-- /input-group -->
  </div><!-- /.col-lg-6 -->
</div><!-- /.row -->

```

Segmented buttons

EXAMPLE

Action	▼	
		Action ▼

```

<div class="input-group">
  <div class="input-group-btn">
    <!-- Button and dropdown menu -->
  </div>
  <input type="text" class="form-control" aria-label="...">
</div>

<div class="input-group">
  <input type="text" class="form-control" aria-label="...">
  <div class="input-group-btn">
    <!-- Button and dropdown menu -->
  </div>
</div>

```

Copy

Navs

Navs available in Bootstrap have shared markup, starting with the base `.nav` class, as well as shared states. Swap modifier classes to switch between each style.

Using navs for tab panels requires JavaScript tabs plugin

For tabs with tabbable areas, you must use the tabs JavaScript plugin (`../javascript/#tabs`). The markup will also require additional `role` and ARIA attributes – see the plugin's example markup (`../javascript/#tabs-usage`) for further details.

Make navs used as navigation accessible

If you are using navs to provide a navigation bar, be sure to add a `role="navigation"` to the most logical parent container of the ``, or wrap a `<nav>` element around the whole navigation. Do not add the role to the `` itself, as this would prevent it from being announced as an actual list by assistive technologies.

Tabs

Note the `.nav-tabs` class requires the `.nav` base class.

EXAMPLE

Home Profile Messages

```
<ul class="nav nav-tabs">
  <li role="presentation" class="active"><a href="#">Home</a></li>
  <li role="presentation"><a href="#">Profile</a></li>
  <li role="presentation"><a href="#">Messages</a></li>
</ul>
```

Copy

Pills

Take that same HTML, but use `.nav-pills` instead:

EXAMPLE

Home Profile Messages

```
<ul class="nav nav-pills">
  <li role="presentation" class="active"><a href="#">Home</a></li>
  <li role="presentation"><a href="#">Profile</a></li>
  <li role="presentation"><a href="#">Messages</a></li>
</ul>
```

Copy

Pills are also vertically stackable. Just add `.nav-stacked`.

EXAMPLE

Home

Profile

Messages

```
<ul class="nav nav-pills nav-stacked">
  ...
</ul>
```

Copy

Justified

Easily make tabs or pills equal widths of their parent at screens wider than 768px with `.nav-justified`. On smaller screens, the nav links are stacked.

Justified navbar nav links are currently not supported.

Safari and responsive justified navs

As of v8.0, Safari exhibits a bug in which resizing your browser horizontally causes rendering errors in the justified nav that are cleared upon refreshing. This bug is also shown in the justified nav example ([../examples/justified-nav/](#)).

EXAMPLE

Home

Profile

Messages

Home

Profile

Messages

```
<ul class="nav nav-tabs nav-justified">
  ...
</ul>
<ul class="nav nav-pills nav-justified">
  ...
</ul>
```

Copy

Disabled links

For any nav component (tabs or pills), add `.disabled` for **gray links and no hover effects**.

Link functionality not impacted

This class will only change the `<a>`'s appearance, not its functionality. Use custom JavaScript to disable links here.

EXAMPLE

Clickable link Clickable link Disabled link

```
<ul class="nav nav-pills">
  ...
  <li role="presentation" class="disabled"><a href="#">Disabled link</a></li>
  ...
</ul>
```

Copy

Using dropdowns

Add dropdown menus with a little extra HTML and the dropdowns JavaScript plugin ([../javascript/#dropdowns](#)).

Tabs with dropdowns

EXAMPLE

Home Help Dropdown ▼

```
<ul class="nav nav-tabs">
  ...
  <li role="presentation" class="dropdown">
    <a class="dropdown-toggle" data-toggle="dropdown" href="#" role="button"
    aria-expanded="false">
      Dropdown <span class="caret"></span>
    </a>
    <ul class="dropdown-menu" role="menu">
      ...
    </ul>
  </li>
  ...
</ul>
```

Copy

Pills with dropdowns

EXAMPLE

Home Help Dropdown ▼

```
<ul class="nav nav-pills">
  ...
  <li role="presentation" class="dropdown">
    <a class="dropdown-toggle" data-toggle="dropdown" href="#" role="button"
    aria-expanded="false">
      Dropdown <span class="caret"></span>
    </a>
    <ul class="dropdown-menu" role="menu">
      ...
    </ul>
  </li>
  ...
</ul>
```

Copy

Navbar

Default navbar

Navbars are responsive meta components that serve as navigation headers for your application or site. They begin collapsed (and are toggleable) in mobile views and become horizontal as the available viewport width increases.

Justified navbar nav links are currently not supported.

Overflowing content

Since Bootstrap doesn't know how much space the content in your navbar needs, you might run into issues with content wrapping into a second row. To resolve this, you can:

- Reduce the amount or width of navbar items.
- Hide certain navbar items at certain screen sizes using responsive utility classes (`../css/#responsive-utilities`).
- Change the point at which your navbar switches between collapsed and horizontal mode. Customize the `@grid-float-breakpoint` variable or add your own media query.

Requires JavaScript plugin

If JavaScript is disabled and the viewport is narrow enough that the navbar collapses, it will be impossible to expand the navbar and view the content within the `.navbar-collapse`.

The responsive navbar requires the collapse plugin (`../javascript/#collapse`) to be included in your version of Bootstrap.

Changing the collapsed mobile navbar breakpoint

The navbar collapses into its vertical mobile view when the viewport is narrower than `@grid-float-breakpoint`, and expands into its horizontal non-mobile view when the viewport is at least `@grid-float-breakpoint` in width. Adjust this variable in the Less source to control when the navbar collapses/expands. The default value is `768px` (the smallest "small" or "tablet" screen).

Make navbars accessible

Be sure to use a `<nav>` element or, if using a more generic element such as a `<div>`, add a `role="navigation"` to every navbar to explicitly identify it as a landmark region for users of assistive technologies.

EXAMPLE

[Copy](#)

```

<nav class="navbar navbar-default">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">Brand</a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Link <span class="sr-only">(current)</span>
</a></li>
        <li><a href="#">Link</a></li>
        <li class="dropdown">
          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button"
aria-expanded="false">Dropdown <span class="caret"></span></a>
          <ul class="dropdown-menu" role="menu">
            <li><a href="#">Action</a></li>
            <li><a href="#">Another action</a></li>
            <li><a href="#">Something else here</a></li>
            <li class="divider"></li>
            <li><a href="#">Separated link</a></li>
            <li class="divider"></li>
            <li><a href="#">One more separated link</a></li>
          </ul>
        </li>
      </ul>
      <form class="navbar-form navbar-left" role="search">
        <div class="form-group">
          <input type="text" class="form-control" placeholder="Search">
        </div>
        <button type="submit" class="btn btn-default">Submit</button>
      </form>
      <ul class="nav navbar-nav navbar-right">
        <li><a href="#">Link</a></li>
        <li class="dropdown">
          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button"
aria-expanded="false">Dropdown <span class="caret"></span></a>
          <ul class="dropdown-menu" role="menu">
            <li><a href="#">Action</a></li>
            <li><a href="#">Another action</a></li>
            <li><a href="#">Something else here</a></li>
            <li class="divider"></li>
            <li><a href="#">Separated link</a></li>
          </ul>
        </li>
      </ul>
    </div><!-- /.navbar-collapse -->
  </div><!-- /.container-fluid -->
</nav>

```

Brand image

Replace the navbar brand with your own image by swapping the text for an ``. Since the `.navbar-brand` has its own padding and height, you may need to override some CSS depending on your image.

EXAMPLE

```
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">
        
      </a>
    </div>
  </div>
</nav>
```

Copy

Forms

Place form content within `.navbar-form` for proper vertical alignment and collapsed behavior in narrow viewports. Use the alignment options to decide where it resides within the navbar content.

As a heads up, `.navbar-form` shares much of its code with `.form-inline` via mixin. **Some form controls, like input groups, may require fixed widths to be show up properly within a navbar.**

EXAMPLE

```
<form class="navbar-form navbar-left" role="search">
  <div class="form-group">
    <input type="text" class="form-control" placeholder="Search">
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

Copy

Mobile device caveats

There are some caveats regarding using form controls within fixed elements on mobile devices. See our browser support docs ([../getting-started/#support-fixed-position-keyboards](#)) for details.

Always add labels

Screen readers will have trouble with your forms if you don't include a label for every input. For these inline forms, you can hide the labels using the `.sr-only` class. There are further alternative methods of providing a label for assistive technologies, such as the `aria-label`, `aria-labelledby` or `title` attribute. If none of these is present, screen readers may resort to using the `placeholder` attribute, if present, but note that use of `placeholder` as a replacement for other labelling methods is not advised.

Buttons

Add the `.navbar-btn` class to `<button>` elements not residing in a `<form>` to vertically center them in the navbar.

EXAMPLE

```
<button type="button" class="btn btn-default navbar-btn">Sign in</button>
```

Copy

Context-specific usage

Like the standard button classes (`../css/#buttons`), `.navbar-btn` can be used on `<a>` and `<input>` elements. However, neither `.navbar-btn` nor the standard button classes should be used on `<a>` elements within `.navbar-nav`.

Text

Wrap strings of text in an element with `.navbar-text`, usually on a `<p>` tag for proper leading and color.

EXAMPLE

```
<p class="navbar-text">Signed in as Mark Otto</p>
```

Copy

Non-nav links

For folks using standard links that are not within the regular navbar navigation component, use the `.navbar-link` class to add the proper colors for the default and inverse navbar options.

EXAMPLE

```
<p class="navbar-text navbar-right">Signed in as <a href="#" class="navbar-link">Mark  
Otto</a></p>
```

Copy

Component alignment

Align nav links, forms, buttons, or text, using the `.navbar-left` or `.navbar-right` utility classes. Both classes will add a CSS float in the specified direction. For example, to align nav links, put them in a separate `` with the respective utility class applied.

These classes are mixin-ed versions of `.pull-left` and `.pull-right`, but they're scoped to media queries for easier handling of navbar components across device sizes.

Right aligning multiple components

Navbars currently have a limitation with multiple `.navbar-right` classes. To properly space content, we use negative margin on the last `.navbar-right` element. When there are multiple elements using that class, these margins don't work as intended.

We'll revisit this when we can rewrite that component in v4.

Fixed to top

Add `.navbar-fixed-top` and include a `.container` or `.container-fluid` to center and pad navbar content.

EXAMPLE

```
<nav class="navbar navbar-default navbar-fixed-top">  
  <div class="container">  
    ...  
  </div>  
</nav>
```

Copy

Body padding required

The fixed navbar will overlay your other content, unless you add `padding` to the top of the `<body>`. Try out your own values or use our snippet below. Tip: By default, the navbar is 50px high.

```
body { padding-top: 70px; }
```

Copy

Make sure to include this **after** the core Bootstrap CSS.

Fixed to bottom

Add `.navbar-fixed-bottom` and include a `.container` or `.container-fluid` to center and pad navbar content.

EXAMPLE

```
<nav class="navbar navbar-default navbar-fixed-bottom">
  <div class="container">
    ...
  </div>
</nav>
```

Copy

Body padding required

The fixed navbar will overlay your other content, unless you add `padding` to the bottom of the `<body>`. Try out your own values or use our snippet below. Tip: By default, the navbar is 50px high.

```
body { padding-bottom: 70px; }
```

Copy

Make sure to include this **after** the core Bootstrap CSS.

Static top

Create a full-width navbar that scrolls away with the page by adding `.navbar-static-top` and include a `.container` or `.container-fluid` to center and pad navbar content.

Unlike the `.navbar-fixed-*` classes, you do not need to change any padding on the `body`.

EXAMPLE

```
<nav class="navbar navbar-default navbar-static-top">
  <div class="container">
    ...
  </div>
</nav>
```

Copy

Inverted navbar

Modify the look of the navbar by adding `.navbar-inverse`.

EXAMPLE

Copy

```
<nav class="navbar navbar-inverse">
  ...
</nav>
```

Breadcrumbs

Indicate the current page's location within a navigational hierarchy.

Separators are automatically added in CSS through `:before` and `content` .

EXAMPLE

Home

Home / Library

Home / Library / Data

```
<ol class="breadcrumb">
  <li><a href="#">Home</a></li>
  <li><a href="#">Library</a></li>
  <li class="active">Data</li>
</ol>
```

[Copy](#)

Pagination

Provide pagination links for your site or app with the multi-page pagination component, or the simpler pager alternative.

Default pagination

Simple pagination inspired by Rdio, great for apps and search results. The large block is hard to miss, easily scalable, and provides large click areas.

EXAMPLE

« 1 2 3 4 5 »

[Copy](#)

```
<nav>
  <ul class="pagination">
    <li>
      <a href="#" aria-label="Previous">
        <span aria-hidden="true">&laquo;</span>
      </a>
    </li>
    <li><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
    <li>
      <a href="#" aria-label="Next">
        <span aria-hidden="true">&raquo;</span>
      </a>
    </li>
  </ul>
</nav>
```

Disabled and active states

Links are customizable for different circumstances. Use `.disabled` for unclickable links and `.active` to indicate the current page.

EXAMPLE



```
<nav>
  <ul class="pagination">
    <li class="disabled"><a href="#" aria-label="Previous"><span aria-hidden="true">&laquo;</span></a></li>
    <li class="active"><a href="#">1 <span class="sr-only">(current)</span></a></li>
    ...
  </ul>
</nav>
```

Copy

You can optionally swap out active or disabled anchors for ``, or omit the anchor in the case of the previous/next arrows, to remove click functionality while retaining intended styles.

```
<nav>
  <ul class="pagination">
    <li class="disabled">
      <span>
        <span aria-hidden="true">&laquo;</span>
      </span>
    </li>
    <li class="active">
      <span>1 <span class="sr-only">(current)</span></span>
    </li>
    ...
  </ul>
</nav>
```

Copy

Sizing

Fancy larger or smaller pagination? Add `.pagination-lg` or `.pagination-sm` for additional sizes.

EXAMPLE



```
<nav><ul class="pagination pagination-lg">...</ul></nav>
<nav><ul class="pagination">...</ul></nav>
<nav><ul class="pagination pagination-sm">...</ul></nav>
```

Copy

Pager

Quick previous and next links for simple pagination implementations with light markup and styles. It's great for simple sites like blogs or magazines.

Default example

By default, the pager centers links.

EXAMPLE

Previous Next

```
<nav>
  <ul class="pager">
    <li><a href="#">Previous</a></li>
    <li><a href="#">Next</a></li>
  </ul>
</nav>
```

Copy

Aligned links

Alternatively, you can align each link to the sides:

EXAMPLE

← Older

Newer →

```
<nav>
  <ul class="pager">
    <li class="previous"><a href="#"><span aria-hidden="true">&larr;</span> Older</a></li>
    <li class="next"><a href="#">Newer <span aria-hidden="true">&rarr;</span></a></li>
  </ul>
</nav>
```

Copy

Optional disabled state

Pager links also use the general `.disabled` utility class from the pagination.

EXAMPLE

← Older

Newer →

```
<nav>
  <ul class="pager">
    <li class="previous disabled"><a href="#"><span aria-hidden="true">&larr;</span>
Older</a></li>
    <li class="next"><a href="#">Newer <span aria-hidden="true">&rarr;</span></a></li>
  </ul>
</nav>
```

Copy

Labels

Example

EXAMPLE

Example heading New

Example heading New

Example heading New

Example heading New

Example heading New

Example heading New

```
<h3>Example heading <span class="label label-default">New</span></h3>
```

Copy

Available variations

Add any of the below mentioned modifier classes to change the appearance of a label.

EXAMPLE

Default

Primary

Success

Info

Warning

Danger

Copy

```
<span class="label label-default">Default</span>
<span class="label label-primary">Primary</span>
<span class="label label-success">Success</span>
<span class="label label-info">Info</span>
<span class="label label-warning">Warning</span>
<span class="label label-danger">Danger</span>
```

Have tons of labels?

Rendering problems can arise when you have dozens of inline labels within a narrow container, each containing its own `inline-block` element (like an icon). The way around this is setting `display: inline-block`; . For context and an example, see #13219 (<https://github.com/twbs/bootstrap/issues/13219>).

Badges

Easily highlight new or unread items by adding a `` to links, Bootstrap navs, and more.

EXAMPLE

Inbox 42

Messages 4

```
<a href="#">Inbox <span class="badge">42</span></a>
```

Copy

```
<button class="btn btn-primary" type="button">
  Messages <span class="badge">4</span>
</button>
```

Self collapsing

When there are no new or unread items, badges will simply collapse (via CSS's `:empty` selector) provided no content exists within.

Cross-browser compatibility

Badges won't self collapse in Internet Explorer 8 because it lacks support for the `:empty` selector.

Adapts to active nav states

Built-in styles are included for placing badges in active states in pill navigations.

EXAMPLE

Home 42 Profile Messages 3

Copy

```
<ul class="nav nav-pills" role="tablist">
  <li role="presentation" class="active"><a href="#">Home <span class="badge">42</span>
</a></li>
  <li role="presentation"><a href="#">Profile</a></li>
  <li role="presentation"><a href="#">Messages <span class="badge">3</span></a></li>
</ul>
```

Jumbotron

A lightweight, flexible component that can optionally extend the entire viewport to showcase key content on your site.

EXAMPLE

Hello, world!

This is a simple hero unit, a simple jumbotron-style component for calling extra attention to featured content or information.

Learn more

```
<div class="jumbotron">
  <h1>Hello, world!</h1>
  <p>...</p>
  <p><a class="btn btn-primary btn-lg" href="#" role="button">Learn more</a></p>
</div>
```

Copy

To make the jumbotron full width, and without rounded corners, place it outside all `.container` s and instead add a `.container` within.

```
<div class="jumbotron">
  <div class="container">
    ...
  </div>
</div>
```

Copy

Page header

A simple shell for an `h1` to appropriately space out and segment sections of content on a page. It can utilize the `h1`'s default `small` element, as well as most other components (with additional styles).

EXAMPLE

Example page header

Subtext for header

```
<div class="page-header">
  <h1>Example page header <small>Subtext for header</small></h1>
</div>
```

Copy

Thumbnails

Extend Bootstrap's grid system ([../css/#grid](#)) with the thumbnail component to easily display grids of images, videos, text, and more.

If you're looking for Pinterest-like presentation of thumbnails of varying heights and/or widths, you'll need to use a third-party plugin such as [Masonry](http://masonry.desandro.com) (<http://masonry.desandro.com>), [Isotope](http://isotope.metafizzy.co) (<http://isotope.metafizzy.co>), or [Salvattore](http://salvattore.com) (<http://salvattore.com>).

Default example

By default, Bootstrap's thumbnails are designed to showcase linked images with minimal required markup.

EXAMPLE



Copy


```
<div class="row">
  <div class="col-xs-6 col-md-3">
    <a href="#" class="thumbnail">
      
    </a>
  </div>
  ...
</div>
```

Custom content

With a bit of extra markup, it's possible to add any kind of HTML content like headings, paragraphs, or buttons into thumbnails.

EXAMPLE



Thumbnail label

Cras justo odio, dapibus ac facilisis in, egestas eget quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id nibh ultricies vehicula ut id elit.

[Button](#)[Button](#)

Thumbnail label

Cras justo odio, dapibus ac facilisis in, egestas eget quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id nibh ultricies vehicula ut id elit.

[Button](#)[Button](#)

Thumbnail label

Cras justo odio, dapibus ac facilisis in, egestas eget quam. Donec id elit non mi porta gravida at eget metus. Nullam id dolor id nibh ultricies vehicula ut id elit.

Button

Button

```
<div class="row">
  <div class="col-sm-6 col-md-4">
    <div class="thumbnail">
      
      <div class="caption">
        <h3>Thumbnail label</h3>
        <p>...</p>
        <p><a href="#" class="btn btn-primary" role="button">Button</a> <a href="#"
class="btn btn-default" role="button">Button</a></p>
      </div>
    </div>
  </div>
</div>
</div>
```

Copy

Alerts

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

Examples

Wrap any text and an optional dismiss button in `.alert` and one of the four contextual classes (e.g., `.alert-success`) for basic alert messages.

No default class

Alerts don't have default classes, only base and modifier classes. A default gray alert doesn't make too much sense, so you're required to specify a type via contextual class. Choose from success, info, warning, or danger.

EXAMPLE

Well done! You successfully read this important alert message.

Heads up! This alert needs your attention, but it's not super important.

Warning! Better check yourself, you're not looking too good.

Oh snap! Change a few things up and try submitting again.

```
<div class="alert alert-success" role="alert">...</div>
<div class="alert alert-info" role="alert">...</div>
<div class="alert alert-warning" role="alert">...</div>
<div class="alert alert-danger" role="alert">...</div>
```

Copy

Dismissible alerts

Build on any alert by adding an optional `.alert-dismissible` and close button.

Requires JavaScript alert plugin

For fully functioning, dismissible alerts, you must use the alerts JavaScript plugin (`../javascript/#alerts`).

EXAMPLE

Warning! Better check yourself, you're not looking too good.



```
<div class="alert alert-warning alert-dismissible" role="alert">
  <button type="button" class="close" data-dismiss="alert" aria-label="Close"><span
    aria-hidden="true">&times;</span></button>
  <strong>Warning!</strong> Better check yourself, you're not looking too good.
</div>
```

Copy

Ensure proper behavior across all devices

Be sure to use the `<button>` element with the `data-dismiss="alert"` data attribute.

Links in alerts

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

EXAMPLE

Well done! You successfully read **this important alert message**.

Heads up! This **alert needs your attention**, but it's not super important.

Warning! Better check yourself, you're **not looking too good**.

Oh snap! Change a **few things up** and try submitting again.

Copy

```
<div class="alert alert-success" role="alert">
  <a href="#" class="alert-link">...</a>
</div>
<div class="alert alert-info" role="alert">
  <a href="#" class="alert-link">...</a>
</div>
<div class="alert alert-warning" role="alert">
  <a href="#" class="alert-link">...</a>
</div>
<div class="alert alert-danger" role="alert">
  <a href="#" class="alert-link">...</a>
</div>
```

Progress bars

Provide up-to-date feedback on the progress of a workflow or action with simple yet flexible progress bars.

Cross-browser compatibility

Progress bars use CSS3 transitions and animations to achieve some of their effects. These features are not supported in Internet Explorer 9 and below or older versions of Firefox. Opera 12 does not support animations.

Basic example

Default progress bar.

EXAMPLE

```
<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="60" aria-valuemin="0"
  aria-valuemax="100" style="width: 60%;">
    <span class="sr-only">60% Complete</span>
  </div>
</div>
```

[Copy](#)

With label

Remove the `` with `.sr-only` class from within the progress bar to show a visible percentage.

EXAMPLE

60%

[Copy](#)

```
<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="60" aria-valuemin="0"
aria-valuemax="100" style="width: 60%;">
    60%
  </div>
</div>
```

To ensure that the label text remains legible even for low percentages, consider adding a `min-width` to the progress bar.

EXAMPLE

0%

2%

```
<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="0" aria-valuemin="0"
aria-valuemax="100" style="min-width: 2em;">
    0%
  </div>
</div>
<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="2" aria-valuemin="0"
aria-valuemax="100" style="min-width: 2em; width: 2%;">
    2%
  </div>
</div>
```

Copy

Contextual alternatives

Progress bars use some of the same button and alert classes for consistent styles.

EXAMPLE

Copy

```

<div class="progress">
  <div class="progress-bar progress-bar-success" role="progressbar" aria-valuenow="40"
aria-valuemin="0" aria-valuemax="100" style="width: 40%">
    <span class="sr-only">40% Complete (success)</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-info" role="progressbar" aria-valuenow="20"
aria-valuemin="0" aria-valuemax="100" style="width: 20%">
    <span class="sr-only">20% Complete</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-warning" role="progressbar" aria-valuenow="60"
aria-valuemin="0" aria-valuemax="100" style="width: 60%">
    <span class="sr-only">60% Complete (warning)</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-danger" role="progressbar" aria-valuenow="80"
aria-valuemin="0" aria-valuemax="100" style="width: 80%">
    <span class="sr-only">80% Complete (danger)</span>
  </div>
</div>

```

Striped

Uses a gradient to create a striped effect. Not available in IE9 and below.

EXAMPLE

[Copy](#)

```

<div class="progress">
  <div class="progress-bar progress-bar-success progress-bar-striped" role="progressbar"
aria-valuenow="40" aria-valuemin="0" aria-valuemax="100" style="width: 40%">
    <span class="sr-only">40% Complete (success)</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-info progress-bar-striped" role="progressbar"
aria-valuenow="20" aria-valuemin="0" aria-valuemax="100" style="width: 20%">
    <span class="sr-only">20% Complete</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-warning progress-bar-striped" role="progressbar"
aria-valuenow="60" aria-valuemin="0" aria-valuemax="100" style="width: 60%">
    <span class="sr-only">60% Complete (warning)</span>
  </div>
</div>
<div class="progress">
  <div class="progress-bar progress-bar-danger progress-bar-striped" role="progressbar"
aria-valuenow="80" aria-valuemin="0" aria-valuemax="100" style="width: 80%">
    <span class="sr-only">80% Complete (danger)</span>
  </div>
</div>

```

Animated

Add `.active` to `.progress-bar-striped` to animate the stripes right to left. Not available in IE9 and below.

EXAMPLE

Toggle animation

```

<div class="progress">
  <div class="progress-bar progress-bar-striped active" role="progressbar"
aria-valuenow="45" aria-valuemin="0" aria-valuemax="100" style="width: 45%">
    <span class="sr-only">45% Complete</span>
  </div>
</div>

```

Copy

Stacked

Place multiple bars into the same `.progress` to stack them.

EXAMPLE

Copy

```
<div class="progress">
  <div class="progress-bar progress-bar-success" style="width: 35%">
    <span class="sr-only">35% Complete (success)</span>
  </div>
  <div class="progress-bar progress-bar-warning progress-bar-striped" style="width: 20%">
    <span class="sr-only">20% Complete (warning)</span>
  </div>
  <div class="progress-bar progress-bar-danger" style="width: 10%">
    <span class="sr-only">10% Complete (danger)</span>
  </div>
</div>
```

Media object

Abstract object styles for building various types of components (like blog comments, Tweets, etc) that feature a left- or right-aligned image alongside textual content.

Default media

The default media displays a media object (images, video, audio) to the left or right of a content block.

EXAMPLE

64x64

Media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

64x64

Media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

64x64

Nested media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

Media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

64x64

64x64

Media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

64x64

Copy


```
<div class="media">
  <div class="media-left">
    <a href="#">
      
    </a>
  </div>
  <div class="media-body">
    <h4 class="media-heading">Media heading</h4>
    ...
  </div>
</div>
```

The classes `.pull-left` and `.pull-right` also exist and were previously used as part of the media component, but are deprecated for that use as of v3.3.0. They are approximately equivalent to `.media-left` and `.media-right`, except that `.media-right` should be placed after the `.media-body` in the html.

Media alignment

The images or other media can be aligned top, middle, or bottom. The default is top aligned.

EXAMPLE

64x64

Top aligned media

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

Donec sed odio dui. Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

64x64

Middle aligned media

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

Donec sed odio dui. Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

64x64

Bottom aligned media

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis. Fusce condimentum nunc ac nisi vulputate fringilla. Donec lacinia congue felis in faucibus.

Donec sed odio dui. Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

```
<div class="media">
  <div class="media-left media-middle">
    <a href="#">
      
    </a>
  </div>
  <div class="media-body">
    <h4 class="media-heading">Middle aligned media</h4>
    ...
  </div>
</div>
```

Copy

Media list

With a bit of extra markup, you can use media inside list (useful for comment threads or articles lists).

EXAMPLE

64x64

Media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

64x64

Nested media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

64x64

Nested media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

64x64

Nested media heading

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus scelerisque ante sollicitudin commodo. Cras purus odio, vestibulum in vulputate at, tempus viverra turpis.

```
<ul class="media-list">
  <li class="media">
    <div class="media-left">
      <a href="#">
        
      </a>
    </div>
    <div class="media-body">
      <h4 class="media-heading">Media heading</h4>
      ...
    </div>
  </li>
</ul>
```

Copy

List group

List groups are a flexible and powerful component for displaying not only simple lists of elements, but complex ones with custom content.

Basic example

The most basic list group is simply an unordered list with list items, and the proper classes. Build upon it with the options that follow, or your own CSS as needed.

EXAMPLE

Cras justo odio

Dapibus ac facilisis in
Morbi leo risus
Porta ac consectetur ac
Vestibulum at eros

```
<ul class="list-group">
  <li class="list-group-item">Cras justo odio</li>
  <li class="list-group-item">Dapibus ac facilisis in</li>
  <li class="list-group-item">Morbi leo risus</li>
  <li class="list-group-item">Porta ac consectetur ac</li>
  <li class="list-group-item">Vestibulum at eros</li>
</ul>
```

Copy

Badges

Add the badges component to any list group item and it will automatically be positioned on the right.

EXAMPLE

Cras justo odio	14
Dapibus ac facilisis in	2
Morbi leo risus	1

```
<ul class="list-group">
  <li class="list-group-item">
    <span class="badge">14</span>
    Cras justo odio
  </li>
</ul>
```

Copy

Linked items

Linkify list group items by using anchor tags instead of list items (that also means a parent `<div>` instead of an ``). No need for individual parents around each element.

EXAMPLE

Cras justo odio
Dapibus ac facilisis in
Morbi leo risus
Porta ac consectetur ac
Vestibulum at eros

Copy

```
<div class="list-group">
  <a href="#" class="list-group-item active">
    Cras justo odio
  </a>
  <a href="#" class="list-group-item">Dapibus ac facilisis in</a>
  <a href="#" class="list-group-item">Morbi leo risus</a>
  <a href="#" class="list-group-item">Porta ac consectetur ac</a>
  <a href="#" class="list-group-item">Vestibulum at eros</a>
</div>
```

Disabled items

Add `.disabled` to a `.list-group-item` to gray it out to appear disabled.

EXAMPLE

Cras justo odio

Dapibus ac facilisis in

Morbi leo risus

Porta ac consectetur ac

Vestibulum at eros

```
<div class="list-group">
  <a href="#" class="list-group-item disabled">
    Cras justo odio
  </a>
  <a href="#" class="list-group-item">Dapibus ac facilisis in</a>
  <a href="#" class="list-group-item">Morbi leo risus</a>
  <a href="#" class="list-group-item">Porta ac consectetur ac</a>
  <a href="#" class="list-group-item">Vestibulum at eros</a>
</div>
```

[Copy](#)

Contextual classes

Use contextual classes to style list items, default or linked. Also includes `.active` state.

EXAMPLE

Dapibus ac facilisis in

Cras sit amet nibh libero

Porta ac consectetur ac

Vestibulum at eros

Dapibus ac facilisis in

Cras sit amet nibh libero

Porta ac consectetur ac

Vestibulum at eros

[Copy](#)

```

<ul class="list-group">
  <li class="list-group-item list-group-item-success">Dapibus ac facilisis in</li>
  <li class="list-group-item list-group-item-info">Cras sit amet nibh libero</li>
  <li class="list-group-item list-group-item-warning">Porta ac consectetur ac</li>
  <li class="list-group-item list-group-item-danger">Vestibulum at eros</li>
</ul>
<div class="list-group">
  <a href="#" class="list-group-item list-group-item-success">Dapibus ac facilisis in</a>
  <a href="#" class="list-group-item list-group-item-info">Cras sit amet nibh libero</a>
  <a href="#" class="list-group-item list-group-item-warning">Porta ac consectetur ac</a>
  <a href="#" class="list-group-item list-group-item-danger">Vestibulum at eros</a>
</div>

```

Custom content

Add nearly any HTML within, even for linked list groups like the one below.

EXAMPLE

List group item heading

Donec id elit non mi porta gravida at eget metus.
Maecenas sed diam eget risus varius blandit.

List group item heading

Donec id elit non mi porta gravida at eget metus.
Maecenas sed diam eget risus varius blandit.

List group item heading

Donec id elit non mi porta gravida at eget metus.
Maecenas sed diam eget risus varius blandit.

```

<div class="list-group">
  <a href="#" class="list-group-item active">
    <h4 class="list-group-item-heading">List group item heading</h4>
    <p class="list-group-item-text">...</p>
  </a>
</div>

```

[Copy](#)

Panels

While not always necessary, sometimes you need to put your DOM in a box. For those situations, try the panel component.

Basic example

By default, all the `.panel` does is apply some basic border and padding to contain some content.

EXAMPLE

Basic panel example

```
<div class="panel panel-default">
  <div class="panel-body">
    Basic panel example
  </div>
</div>
```

Copy

Panel with heading

Easily add a heading container to your panel with `.panel-heading`. You may also include any `<h1>` - `<h6>` with a `.panel-title` class to add a pre-styled heading.

For proper link coloring, be sure to place links in headings within `.panel-title`.

EXAMPLE

Panel heading without title

Panel content

Panel title

Panel content

```
<div class="panel panel-default">
  <div class="panel-heading">Panel heading without title</div>
  <div class="panel-body">
    Panel content
  </div>
</div>

<div class="panel panel-default">
  <div class="panel-heading">
    <h3 class="panel-title">Panel title</h3>
  </div>
  <div class="panel-body">
    Panel content
  </div>
</div>
```

Copy

Panel with footer

Wrap buttons or secondary text in `.panel-footer`. Note that panel footers **do not** inherit colors and borders when using contextual variations as they are not meant to be in the foreground.

EXAMPLE

Panel content

Panel footer

Copy

```
<div class="panel panel-default">
  <div class="panel-body">
    Panel content
  </div>
  <div class="panel-footer">Panel footer</div>
</div>
```

Contextual alternatives

Like other components, easily make a panel more meaningful to a particular context by adding any of the contextual state classes.

EXAMPLE

Panel title

Panel content

Panel title

Panel content

Panel title

Panel content

Panel title

Panel content

Panel title

Panel content

```
<div class="panel panel-primary">...</div>
<div class="panel panel-success">...</div>
<div class="panel panel-info">...</div>
<div class="panel panel-warning">...</div>
<div class="panel panel-danger">...</div>
```

Copy

With tables

Add any non-bordered `.table` within a panel for a seamless design. If there is a `.panel-body`, we add an extra border to the top of the table for separation.

EXAMPLE

Panel heading

Some default panel content here. Nulla vitae elit libero, a pharetra augue. Aenean lacinia bibendum nulla sed consectetur. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Nullam id dolor id nibh ultricies vehicula ut id elit.

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

```
<div class="panel panel-default">
  <!-- Default panel contents -->
  <div class="panel-heading">Panel heading</div>
  <div class="panel-body">
    <p>...</p>
  </div>

  <!-- Table -->
  <table class="table">
    ...
  </table>
</div>
```

Copy

If there is no panel body, the component moves from panel header to table without interruption.

EXAMPLE

Panel heading			
#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

```
<div class="panel panel-default">
  <!-- Default panel contents -->
  <div class="panel-heading">Panel heading</div>

  <!-- Table -->
  <table class="table">
    ...
  </table>
</div>
```

Copy

With list groups

Easily include full-width list groups within any panel.

EXAMPLE

Panel heading

Some default panel content here. Nulla vitae elit libero, a pharetra augue. Aenean lacinia bibendum nulla sed consectetur. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Nullam id dolor id nibh ultricies vehicula ut id elit.

Cras justo odio

Dapibus ac facilisis in

Morbi leo risus

Porta ac consectetur ac

Vestibulum at eros

```
<div class="panel panel-default">
  <!-- Default panel contents -->
  <div class="panel-heading">Panel heading</div>
  <div class="panel-body">
    <p>...</p>
  </div>

  <!-- List group -->
  <ul class="list-group">
    <li class="list-group-item">Cras justo odio</li>
    <li class="list-group-item">Dapibus ac facilisis in</li>
    <li class="list-group-item">Morbi leo risus</li>
    <li class="list-group-item">Porta ac consectetur ac</li>
    <li class="list-group-item">Vestibulum at eros</li>
  </ul>
</div>
```

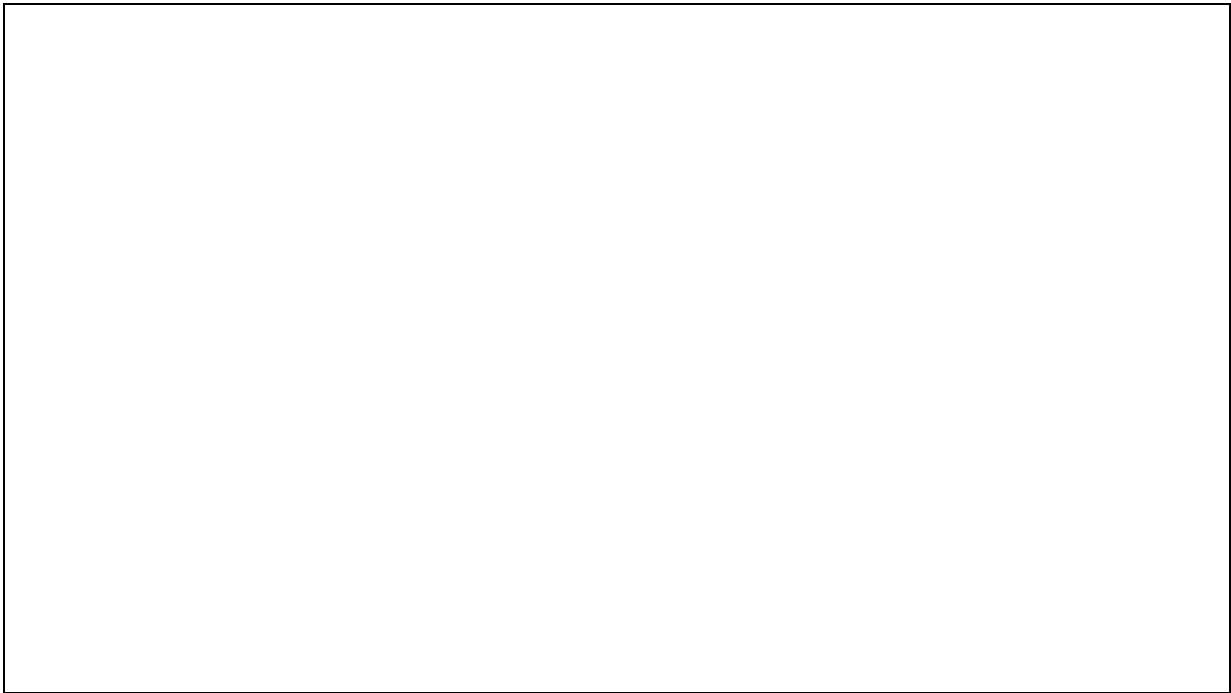
Copy

Responsive embed

Allow browsers to determine video or slideshow dimensions based on the width of their containing block by creating an intrinsic ratio that will properly scale on any device.

Rules are directly applied to `<iframe>`, `<embed>`, `<video>`, and `<object>` elements; optionally use an explicit descendant class `.embed-responsive-item` when you want to match the styling for other attributes.

Pro-Tip! You don't need to include `frameborder="0"` in your `<iframe>`s as we override that for you.

EXAMPLE

```
<!-- 16:9 aspect ratio -->
<div class="embed-responsive embed-responsive-16by9">
  <iframe class="embed-responsive-item" src="..."></iframe>
</div>

<!-- 4:3 aspect ratio -->
<div class="embed-responsive embed-responsive-4by3">
  <iframe class="embed-responsive-item" src="..."></iframe>
</div>
```

Copy

Wells

Default well

Use the well as a simple effect on an element to give it an inset effect.

EXAMPLE

Look, I'm in a well!

```
<div class="well">...</div>
```

Copy

Optional classes

Control padding and rounded corners with two optional modifier classes.

EXAMPLE

Look, I'm in a large well!

```
<div class="well well-lg">...</div>
```

Copy

EXAMPLE

Look, I'm in a small well!

```
<div class="well well-sm">...</div>
```

Copy



(<https://twitter.com/getbootstrap>)



(<https://twitter.com/share>)

Designed and built with all the love in the world by @mdo (<https://twitter.com/mdo>) and @fat (<https://twitter.com/fat>).

Maintained by the core team (<https://github.com/orgs/twbs/people>) with the help of our contributors (<https://github.com/twbs/bootstrap/graphs/contributors>).

Code licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>), documentation under CC BY 3.0 (<https://creativecommons.org/licenses/by/3.0/>).

Currently v3.3.4 · GitHub (<https://github.com/twbs/bootstrap>) · Examples ([../getting-started/#examples](#)) · v2.3.2 docs ([../2.3.2/](#)) · About ([../about/](#)) · Expo (<http://expo.getbootstrap.com>) · Blog (<http://blog.getbootstrap.com>) · Issues (<https://github.com/twbs/bootstrap/issues>) · Releases (<https://github.com/twbs/bootstrap/releases>)