

Advanced Responsive HTML & Query Resolution



TatvaSoft

sculpting
thoughts...

Topics

- Modals
- Dropdown
- Scrollspy
- Tooltip
- Popovers
- Alerts
- Collapse
- Accordion
- Helper Classes

Bootstrap JavaScript Component Overview

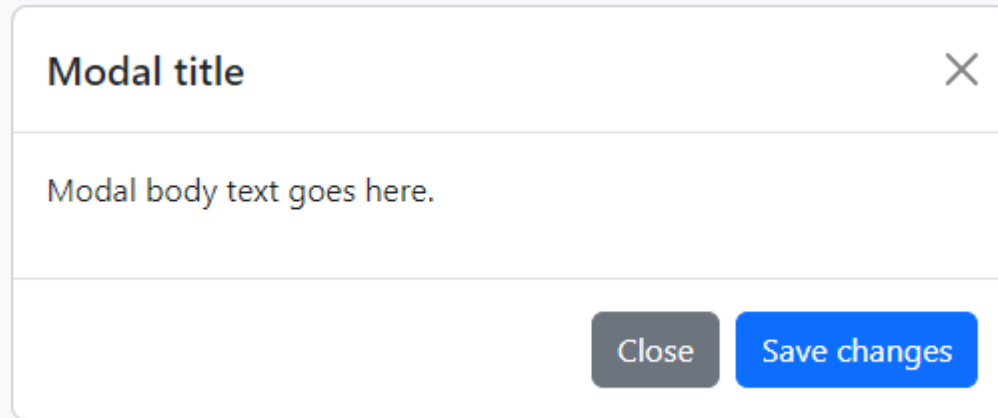
Plugins can be included individually (using Bootstrap's individual *.js files), or all at once (using bootstrap.js or the minified bootstrap.min.js).

Using the compiled JavaScript

Both bootstrap.js and bootstrap.min.js contain all plugins in a single file. Include any one.

Modals

- Modals are streamlined, but flexible, dialog prompts with the minimum required functionality and smart defaults.



<https://getbootstrap.com/docs/5.2/components/modal/>

Dropdown

- Dropdowns are toggleable, contextual overlays for displaying lists of links and more.
- They're toggled by clicking, not by hovering; this is an intentional design decision.

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" data-bs-toggle="dropdown" aria-expanded="false">
    Dropdown button
  </button>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#">Action</a></li>
    <li><a class="dropdown-item" href="#">Another action</a></li>
    <li><a class="dropdown-item" href="#">Something else here</a></li>
  </ul>
</div>
```

Dropdown button ▾

Action

Another action

Something else here

- <https://getbootstrap.com/docs/5.2/components/dropdowns/>

Scrollspy

What is Scrollspy ?

The ScrollSpy plugin is for automatically updating nav targets based on scroll position. Scroll the area below the navbar and watch the active class change. The dropdown sub items will be highlighted as well.

How it works ?

Navbar links must have resolvable id targets.

For example, a `home` must correspond to something in the DOM like `<div id="home"></div>`.

Tooltip

You can activate tooltip by simply defining it in JavaScript file.

```
const tooltipTriggerList = document.querySelectorAll('[data-bs-toggle="tooltip"]')
const tooltipList = [...tooltipTriggerList].map(tooltipTriggerEl => new bootstrap.Tooltip(tooltipTriggerEl))
```

Static Tooltip

Four options are available: top, right, bottom, and left aligned.

EXAMPLE

Tooltip on the left

Tooltip on the top

Tooltip on the bottom

Tooltip on the right

```
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="top" data-bs-title="Tooltip on top">
  Tooltip on top
</button>
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="right" data-bs-title="Tooltip on right">
  Tooltip on right
</button>
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="bottom" data-bs-title="Tooltip on bottom">
  Tooltip on bottom
</button>
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="left" data-bs-title="Tooltip on left">
  Tooltip on left
</button>
```

Popovers

- Add small overlays of content, like those on the iPad, to any element for housing secondary information.
- Four options are available: top, right, bottom, and left aligned.

```
const popoverTriggerList = document.querySelectorAll('[data-bs-toggle="popover"]')
const popoverList = [...popoverTriggerList].map(popoverTriggerEl => new bootstrap.Popover(popoverTriggerEl))
```

Click to toggle popover

Popover title

And here's some amazing content. It's very engaging. Right?

Alerts

- Provide contextual feedback messages for typical user actions with the handful of available and flexible alert messages.
- Add dismiss functionality to all alert messages with this plugin.
- Just add `data-bs-dismiss="alert"` to your close button to automatically give an alert close functionality. Closing an alert removes it from the DOM.

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!

Collapse

- Utilizes a handful of classes for easy toggle behavior.
- Click the buttons to show and hide another element via class changes:
- collapse hides content
- collapsing is applied during transitions

Link with href

Button with data-bs-target

HTML

```
<p>
  <a class="btn btn-primary" data-bs-toggle="collapse" href="#collapseExample" role="button" an
    Link with href
  </a>
  <button class="btn btn-primary" type="button" data-bs-toggle="collapse" data-bs-target="#coll
    Button with data-bs-target
  </button>
</p>
<div class="collapse" id="collapseExample">
  <div class="card card-body">
    Some placeholder content for the collapse component. This panel is hidden by default but re
  </div>
</div>
```

Accordion

- Extend the default collapse behavior to create an accordion with the panel component.
- Just add `data-bs-toggle="collapse"` and a `data-bs-target` to the element to automatically assign control of a collapsible element. The `data-bs-target` attribute accepts a CSS selector to apply the collapse to. Be sure to add the class `collapse` to the collapsible element. If you'd like it to default open, add the additional class in.

Collapsible Group Item #1

Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo. Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt you probably haven't heard of them accusamus labore sustainable VHS.

Collapsible Group Item #2

Collapsible Group Item #3

Helper Classes

Quick floats

Float an element to the left or right with a class. !important is included to avoid specificity issues.

```
<div class="float-start">...</div>
```

```
<div class="float-end">...</div>
```

```
<div class="text-center">...</div>
```

```
.float-start { float: left !important; }
```

```
.float-end { float: right !important; }
```

```
.text-center { text-align: center !important; }
```

Thank You

THANK YOU

ANY QUESTIONS ?