



BOOTSTRAP 5

Forms

([HTTPS://GETBOOTSTRAP.COM/DOCS/5.3/FORMS/](https://getbootstrap.com/docs/5.3/forms/))

CONTENT

1. Overview
2. Form control
3. Select
4. Checks & radios
5. Input group
6. Floating labels
7. Layout
8. Validation

1. OVERVIEW

- Bootstrap's form controls expand on our [Rebooted form styles](#) with classes. Use these classes to opt into their customized displays for a more consistent rendering across browsers and devices.
- Be sure to use an appropriate [type](#) attribute on all inputs (e.g., [email](#) for email address or [number](#) for numerical information) to take advantage of newer input controls like email verification, number selection, and more.

```
<form>
  <div class="mb-3">
    <label for="exampleInputEmail1" class="form-label">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
    <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>
  </div>
  <div class="mb-3">
    <label for="exampleInputPassword1" class="form-label">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1">
  </div>
  <div class="mb-3 form-check">
    <input type="checkbox" class="form-check-input" id="exampleCheck1">
    <label class="form-check-label" for="exampleCheck1">Check me out</label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

Email address

We'll never share your email with anyone else.

Password

☐ Check me out

Submit

FORM TEXT

```
<div class="row g-3 align-items-center">
  <div class="col-auto">
    <label for="inputPassword6" class="col-form-label">Password</label>
  </div>
  <div class="col-auto">
    <input type="password" id="inputPassword6" class="form-control"
      aria-describedby="passwordHelpInline">
  </div>
  <div class="col-auto">
    <span id="passwordHelpInline" class="form-text">
      Must be 8-20 characters long.
    </span>
  </div>
</div>
```

Password

Must be 8-20 characters long.

DISABLED FORMS

- Add the `disabled` boolean attribute on an input to prevent user interactions and make it appear lighter.
- Add the `disabled` attribute to a `<fieldset>` to disable all the controls within. Browsers treat all native form controls (`<input>`, `<select>`, and `<button>` elements) inside a `<fieldset disabled>` as disabled, preventing both keyboard and mouse interactions on them.
- If your form also includes custom button-like elements such as `...`. You must manually modify these controls by adding `tabindex="-1"` to prevent them from receiving focus and `aria-disabled="disabled"` to signal their state to assistive technologies.

```
<form>
  <fieldset disabled>
    <legend>Disabled fieldset example</legend>
    <div class="mb-3">
      <label for="disabledTextInput" class="form-label">Disabled input</label>
      <input type="text" id="disabledTextInput" class="form-control" placeholder="Disabled input">
    </div>
    <div class="mb-3">
      <label for="disabledSelect" class="form-label">Disabled select menu</label>
      <select id="disabledSelect" class="form-select">
        <option>Disabled select</option>
      </select>
    </div>
    <div class="mb-3">
      <div class="form-check">
        <input class="form-check-input" type="checkbox" id="disabledFieldsetCheck" disabled>
        <label class="form-check-label" for="disabledFieldsetCheck">
          Can't check this
        </label>
      </div>
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
  </fieldset>
</form>
```

Disabled fieldset example

Disabled input

Disabled input

Disabled select menu

Disabled select



☐ Can't check this

Submit

2. FORM CONTROL

Give textual form controls like `<input>`s and `<textarea>`s an upgrade with custom styles, sizing, focus states, and more.

2.1. Example

2.2. Sizing

2.3. Readonly plain text

2.4. File input

2.1. EXAMPLE

```
<div class="mb-3">
  <label for="exampleFormControlInput1" class="form-label">Email address</label>
  <input type="email" class="form-control" id="exampleFormControlInput1" placeholder="name@example.com">
</div>
<div class="mb-3">
  <label for="exampleFormControlTextarea1" class="form-label">Example textarea</label>
  <textarea class="form-control" id="exampleFormControlTextarea1" rows="3"></textarea>
</div>
```

Email address

Example textarea

2.2. SIZING

Set heights using classes like `.form-control-lg` and `.form-control-sm`.

```
<input class="form-control form-control-lg" type="text" placeholder=".form-control-lg"
      aria-label=".form-control-lg example">
<input class="form-control" type="text" placeholder="Default input"
      aria-label="default input example">
<input class="form-control form-control-sm" type="text" placeholder=".form-control-sm"
      aria-label=".form-control-sm example">
```

`.form-control-lg`

Default input

`.form-control-sm`

2.3. READONLY PLAIN TEXT

If you want to have `<input readonly>` elements in your form styled as plain text, use the `.form-control-plaintext` class to remove the default form field styling and preserve the correct margin and padding.

```
<div class="mb-3 row">
  <label for="staticEmail" class="col-sm-2 col-form-label">Email</label>
  <div class="col-sm-10">
    <input type="text" readonly class="form-control-plaintext" id="staticEmail"
      value="email@example.com">
  </div>
</div>
<div class="mb-3 row">
  <label for="inputPassword" class="col-sm-2 col-form-label">Password</label>
  <div class="col-sm-10">
    <input type="password" class="form-control" id="inputPassword">
  </div>
</div>
```

| | |
|----------|--------------------------|
| Email | email@example.com |
| Password | <input type="password"/> |

Default file input example

Browse...

No file selected.

Multiple files input example

Browse...

No files selected.

Disabled file input example

Browse...

No file selected.

Small file input example

Browse...

No file selected.

Large file input example

Browse...

No file selected.

3. SELECT

Customize the native `<select>`s with custom CSS that changes the element's initial appearance.

3.1. Default

3.2. Sizing

3.1. DEFAULT

```
<select class="form-select" aria-label="Default select example">  
  <option selected>Open this select menu</option>  
  <option value="1">One</option>  
  <option value="2">Two</option>  
  <option value="3">Three</option>  
</select>
```

Open this select menu



Open this select menu

One

Two

Three

3.2. SIZING

```
<select class="form-select form-select-lg mb-3" aria-label=".form-select-lg example">  
  <option selected>Open this select menu</option>  
  <option value="1">One</option>  
  <option value="2">Two</option>  
  <option value="3">Three</option>  
</select>
```

```
<select class="form-select form-select-sm" aria-label=".form-select-sm example">  
  <option selected>Open this select menu</option>  
  <option value="1">One</option>  
  <option value="2">Two</option>  
  <option value="3">Three</option>  
</select>
```

Open this select menu



Open this select menu



```
<select class="form-select" multiple aria-label="multiple select example">
  <option selected>Open this select menu</option>
  <option value="1">One</option>
  <option value="2">Two</option>
  <option value="3">Three</option>
</select>
```

Open this select menu

One

Two

Three

```
<select class="form-select" size="3" aria-label="size 3 select example">
  <option selected>Open this select menu</option>
  <option value="1">One</option>
  <option value="2">Two</option>
  <option value="3">Three</option>
</select>
```

Open this select menu

One

Two

4. CHECKS & RADIOS

Create consistent cross-browser and cross-device checkboxes and radios with our completely rewritten checks component.

4.1. Checks

4.2. Radios

4.3. Switches

4.4. Inline

4.1. CHECKS

```
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="flexCheckDefault">
  <label class="form-check-label" for="flexCheckDefault">
    Default checkbox
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="flexCheckChecked" checked>
  <label class="form-check-label" for="flexCheckChecked">
    Checked checkbox
  </label>
</div>
```



Default checkbox



Checked checkbox

```
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="flexCheckDisabled" disabled>
  <label class="form-check-label" for="flexCheckDisabled">
    Disabled checkbox
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="flexCheckCheckedDisabled" checked disabled>
  <label class="form-check-label" for="flexCheckCheckedDisabled">
    Disabled checked checkbox
  </label>
</div>
```



Disabled checkbox



Disabled checked checkbox

4.2. RADIOS

```
<div class="form-check">
  <input class="form-check-input" type="radio" name="flexRadioDefault" id="flexRadioDefault1">
  <label class="form-check-label" for="flexRadioDefault1">
    Default radio
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="flexRadioDefault" id="flexRadioDefault2" checked>
  <label class="form-check-label" for="flexRadioDefault2">
    Default checked radio
  </label>
</div>
```

- ☐ Default radio
- ☒ Default checked radio

4.3. SWITCHES

A switch has the markup of a custom checkbox but uses the `.form-switch` class to render a toggle switch.

```
<div class="form-check form-switch">
  <input class="form-check-input" type="checkbox" id="flexSwitchCheckDefault">
  <label class="form-check-label" for="flexSwitchCheckDefault">Default switch checkbox input</label>
</div>
<div class="form-check form-switch">
  <input class="form-check-input" type="checkbox" id="flexSwitchCheckChecked" checked>
  <label class="form-check-label" for="flexSwitchCheckChecked">Checked switch checkbox input</label>
</div>
<div class="form-check form-switch">
  <input class="form-check-input" type="checkbox" id="flexSwitchCheckDisabled" disabled>
  <label class="form-check-label" for="flexSwitchCheckDisabled">Disabled switch checkbox input</label>
</div>
<div class="form-check form-switch">
  <input class="form-check-input" type="checkbox" id="flexSwitchCheckCheckedDisabled" checked disabled>
  <label class="form-check-label" for="flexSwitchCheckCheckedDisabled">Disabled checked switch checkbox input</label>
</div>
```



Default switch checkbox input



Checked switch checkbox input



Disabled switch checkbox input



Disabled checked switch checkbox input

4.4. INLINE

Group checkboxes or radios on the same horizontal row by adding `.form-check-inline` to any `.form-check`.

```
<div class="form-check form-check-inline">
  <input class="form-check-input" type="checkbox" id="inlineCheckbox1" value="option1">
  <label class="form-check-label" for="inlineCheckbox1">1</label>
</div>
<div class="form-check form-check-inline">
  <input class="form-check-input" type="checkbox" id="inlineCheckbox2" value="option2">
  <label class="form-check-label" for="inlineCheckbox2">2</label>
</div>
<div class="form-check form-check-inline">
  <input class="form-check-input" type="checkbox" id="inlineCheckbox3" value="option3" disabled>
  <label class="form-check-label" for="inlineCheckbox3">3 (disabled)</label>
</div>
```

☐ 1 ☐ 2 ☐ 3 (disabled)

5. INPUT GROUP

Easily extend form controls by adding text, buttons, or button groups on either side of textual inputs, custom selects, and custom file inputs.

5.1. Basic example

5.2. Sizing

5.3. Checkboxes and radios

5.4. Button addons

5.1. BASIC EXAMPLE

| | |
|---|---------------------------------------|
| @ | <input type="text" value="Username"/> |
|---|---------------------------------------|

| | |
|---|--------------|
| <input type="text" value="Recipient's username"/> | @example.com |
|---|--------------|

Your vanity URL

| | |
|----------------------------|----------------------|
| https://example.com/users/ | <input type="text"/> |
|----------------------------|----------------------|

| | | |
|----|----------------------|-----|
| \$ | <input type="text"/> | .00 |
|----|----------------------|-----|

| | | |
|---------------------------------------|---|-------------------------------------|
| <input type="text" value="Username"/> | @ | <input type="text" value="Server"/> |
|---------------------------------------|---|-------------------------------------|

| | |
|---------------|------------------------|
| With textarea | <div><div></div></div> |
|---------------|------------------------|

```
<div class="input-group mb-3">
  <span class="input-group-text" id="basic-addon1">@</span>
  <input type="text" class="form-control" placeholder="Username" aria-label="Username" aria-describedby="basic-addon1">
</div>
```

```
<div class="input-group mb-3">
  <input type="text" class="form-control" placeholder="Recipient's username" aria-label="Recipient's username"
    aria-describedby="basic-addon2">
  <span class="input-group-text" id="basic-addon2">@example.com</span>
</div>
```

```
<label for="basic-url" class="form-label">Your vanity URL</label>
<div class="input-group mb-3">
  <span class="input-group-text" id="basic-addon3">https://example.com/users/</span>
  <input type="text" class="form-control" id="basic-url" aria-describedby="basic-addon3">
</div>
```

```
<div class="input-group mb-3">
  <span class="input-group-text">$</span>
  <input type="text" class="form-control" aria-label="Amount (to the nearest dollar)">
  <span class="input-group-text">.00</span>
</div>
```

```
<div class="input-group mb-3">
  <input type="text" class="form-control" placeholder="Username" aria-label="Username">
  <span class="input-group-text">@</span>
  <input type="text" class="form-control" placeholder="Server" aria-label="Server">
</div>
```

```
<div class="input-group">
  <span class="input-group-text">With textarea</span>
  <textarea class="form-control" aria-label="With textarea"></textarea>
</div>
```

5.2. SIZING

```
<div class="input-group input-group-sm mb-3">
  <span class="input-group-text" id="inputGroup-sizing-sm">Small</span>
  <input type="text" class="form-control" aria-label="Sizing example input"
    aria-describedby="inputGroup-sizing-sm">
</div>

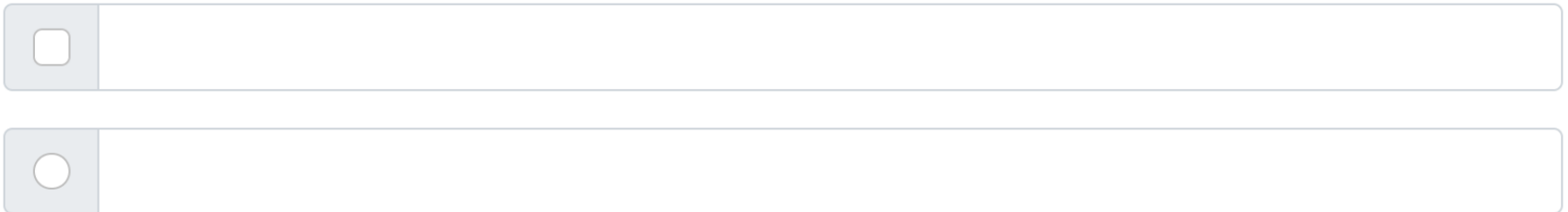
<div class="input-group mb-3">
  <span class="input-group-text" id="inputGroup-sizing-default">Default</span>
  <input type="text" class="form-control" aria-label="Sizing example input"
    aria-describedby="inputGroup-sizing-default">
</div>

<div class="input-group input-group-lg">
  <span class="input-group-text" id="inputGroup-sizing-lg">Large</span>
  <input type="text" class="form-control" aria-label="Sizing example input"
    aria-describedby="inputGroup-sizing-lg">
</div>
```

Small

Default

Large



The image shows the rendered HTML code above. It consists of two rows. The first row features a checkbox on the left, followed by a text input field. The second row features a radio button on the left, followed by a text input field. Both input fields are empty and have a light gray border.

5.4. BUTTON ADDONS

```
<div class="input-group mb-3">
  <button class="btn btn-outline-secondary" type="button" id="button-addon1">Button</button>
  <input type="text" class="form-control" placeholder=""
    aria-label="Example text with button addon" aria-describedby="button-addon1">
</div>
<div class="input-group mb-3">
  <input type="text" class="form-control" placeholder="Recipient's username"
    aria-label="Recipient's username" aria-describedby="button-addon2">
  <button class="btn btn-outline-secondary" type="button" id="button-addon2">Button</button>
</div>
<div class="input-group mb-3">
  <button class="btn btn-outline-secondary" type="button">Button</button>
  <button class="btn btn-outline-secondary" type="button">Button</button>
  <input type="text" class="form-control" placeholder=""
    aria-label="Example text with two button addons">
</div>
<div class="input-group">
  <input type="text" class="form-control" placeholder="Recipient's username"
    aria-label="Recipient's username with two button addons">
  <button class="btn btn-outline-secondary" type="button">Button</button>
  <button class="btn btn-outline-secondary" type="button">Button</button>
</div>
```

Button

Recipient's username

Button

Button

Button

Recipient's username

Button

Button

6. FLOATING LABELS

Create beautifully simple form labels that float over your input fields.

6.1. Example

6.2. Textareas

6.3. Selects

6.1. EXAMPLE

Wrap a pair of `<input class="form-control">` and `<label>` elements in `.form-floating` to enable floating labels with Bootstrap's textual form fields.

```
<div class="form-floating mb-3">
  <input type="email" class="form-control" id="floatingInput" placeholder="name@example.com">
  <label for="floatingInput">Email address</label>
</div>
<div class="form-floating">
  <input type="password" class="form-control" id="floatingPassword" placeholder="Password">
  <label for="floatingPassword">Password</label>
</div>
```


Email address

Password

6.2. TEXTAREAS

By default, `<textarea>`s with `.form-control` will be the same height as `<input>`s.

```
<div class="form-floating">  
  <textarea class="form-control" placeholder="Leave a comment here"  
    id="floatingTextarea"></textarea>  
  <label for="floatingTextarea">Comments</label>  
</div>
```



To set a custom height on your `<textarea>`, do not use the `rows` attribute. Instead, set an explicit `height` (either inline or via custom CSS).

```
<div class="form-floating">  
  <textarea class="form-control" placeholder="Leave a comment here"  
    id="floatingTextarea2" style="height: 100px"></textarea>  
  <label for="floatingTextarea2">Comments</label>  
</div>
```

Comments

6.3. SELECTS

Other than `.form-control`, floating labels are only available on `.form-selects`. They work in the same way, but unlike `<input>`s, they'll always show the `<label>` in its floated state. Selects with `size` and `multiple` are **not supported**.

```
<div class="form-floating">
  <select class="form-select" id="floatingSelect" aria-label="Floating label select example">
    <option selected>Open this select menu</option>
    <option value="1">One</option>
    <option value="2">Two</option>
    <option value="3">Three</option>
  </select>
  <label for="floatingSelect">Works with selects</label>
</div>
```

Works with selects
Open this select menu



7. LAYOUT

Give your forms some structure—from inline to horizontal to custom grid implementations—with our form layout options.

7.1. Forms

7.2. Horizontal form

7.3. Column sizing

7.4. Inline forms

7.1. FORMS

- Every group of form fields should reside in a `<form>` element. Bootstrap provides no default styling for the `<form>` element, but there are some powerful browser features that are provided by default.
 - `<button>`s within a `<form>` default to `type="submit"`, so strive to be specific and always include a type.
 - You can disable every form element within a form with the `disabled` attribute on the `<form>`.
- Since Bootstrap applies `display: block` and `width: 100%` to almost all our form controls, forms will by default stack vertically. Additional classes can be used to vary this layout on a per-form basis.

7.5. HORIZONTAL FORM

Create horizontal forms with the grid by adding the `.row` class to form groups and using the `.col-*-*` classes to specify the width of your labels and controls. Be sure to add `.col-form-label` to your `<label>`s as well so they're vertically centered with their associated form controls.

Email

Password

Radios

- ☒ First radio
- ☐ Second radio
- ☐ Example checkbox

Sign in

```
<form>
  <div class="row mb-3">
    <label for="inputEmail3" class="col-sm-2 col-form-label">Email</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="inputEmail3">
    </div>
  </div>
  <div class="row mb-3">
    <label for="inputPassword3" class="col-sm-2 col-form-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword3">
    </div>
  </div>
  <fieldset class="row mb-3">
    <legend class="col-form-label col-sm-2 pt-0">Radios</legend>
    <div class="col-sm-10">
      <div class="form-check">
        <input class="form-check-input" type="radio" name="gridRadios" id="gridRadios1" value="option1" checked>
        <label class="form-check-label" for="gridRadios1">First radio</label>
      </div>
      <div class="form-check">
        <input class="form-check-input" type="radio" name="gridRadios" id="gridRadios2" value="option2">
        <label class="form-check-label" for="gridRadios2">Second radio</label>
      </div>
    </div>
  </fieldset>
  <div class="row mb-3">
    <div class="col-sm-10 offset-sm-2">
      <div class="form-check">
        <input class="form-check-input" type="checkbox" id="gridCheck1">
        <label class="form-check-label" for="gridCheck1">Example checkbox</label>
      </div>
    </div>
  </div>
  <button type="submit" class="btn btn-primary">Sign in</button>
</form>
```

HORIZONTAL FORM LABEL SIZING

Be sure to use `.col-form-label-sm` or `.col-form-label-lg` to your `<label>`s or `<legend>`s to correctly follow the size of `.form-control-lg` and `.form-control-sm`.

```
<div class="row mb-3">
  <label for="colFormLabelSm" class="col-sm-2 col-form-label col-form-label-sm">Email</label>
  <div class="col-sm-10">
    <input type="email" class="form-control form-control-sm" id="colFormLabelSm" placeholder="col-form-label-sm">
  </div>
</div>
<div class="row mb-3">
  <label for="colFormLabel" class="col-sm-2 col-form-label">Email</label>
  <div class="col-sm-10">
    <input type="email" class="form-control" id="colFormLabel" placeholder="col-form-label">
  </div>
</div>
<div class="row">
  <label for="colFormLabelLg" class="col-sm-2 col-form-label col-form-label-lg">Email</label>
  <div class="col-sm-10">
    <input type="email" class="form-control form-control-lg" id="colFormLabelLg" placeholder="col-form-label-lg">
  </div>
</div>
```

Email

col-form-label-sm

Email

col-form-label

Email

col-form-label-lg

7.6. COLUMN SIZING

```
<div class="row g-3">
  <div class="col-sm-7">
    <input type="text" class="form-control" placeholder="City" aria-label="City">
  </div>
  <div class="col-sm">
    <input type="text" class="form-control" placeholder="State" aria-label="State">
  </div>
  <div class="col-sm">
    <input type="text" class="form-control" placeholder="Zip" aria-label="Zip">
  </div>
</div>
```


@

Username

Choose... ▾

☐ Remember me

Submit

8. VALIDATION

Provide valuable, actionable feedback to your users with HTML5 form validation, via browser default behaviors or custom styles and JavaScript.

8.1. How it works

8.2. Server side

8.1. HOW IT WORKS

- HTML form validation is applied via CSS's two pseudo-classes, `:invalid` and `:valid`. It applies to `<input>`, `<select>`, and `<textarea>` elements.
- Bootstrap scopes the `:invalid` and `:valid` styles to parent `.was-validated` class, usually applied to the `<form>`. Otherwise, any required field without a value shows up as invalid on page load.
- To reset the appearance of the form (for instance, in the case of dynamic form submissions using AJAX), remove the `.was-validated` class from the `<form>` again after submission.
- As a fallback, `.is-invalid` and `.is-valid` classes may be used instead of the pseudo-classes for server-side validation. They do not require a `.was-validated` parent class.
- All modern browsers support the constraint validation API, a series of JavaScript methods for validating form controls.
- Feedback messages may utilize the browser defaults (different for each browser) or our custom feedback styles with additional HTML and CSS.

8.2. SERVER SIDE

We recommend using client-side validation, but in case you require server-side validation, you can indicate invalid and valid form fields with `.is-invalid` and `.is-valid`. Note that `.invalid-feedback` is also supported with these classes.

For invalid fields, ensure that the invalid feedback/error message is associated with the relevant form field using `aria-describedby` (noting that this attribute allows more than one id to be referenced, in case the field already points to additional form text).

Test at url:

<https://getbootstrap.com/docs/5.3/forms/validation/#server-side>



THE END