Bootstrap 4 Custom Forms

❮ Previous

Next ❯

Bootstrap 4 Custom Forms

Bootstrap 4 comes with customized form elements, that are meant to replace browser defaults:

☑ Custom checkbox   ☑ Default checkbox

⦿ Custom radio   ○ Default radio

Custom Range:

Default Range:

Custom select menu ⇕   Default select menu ⌄

Custom file upload   Browse

🔘 Toggle this custom switch

Custom Checkbox

To create a custom checkbox, wrap a container element, like <div>, with a class of `.custom-control` and `.custom-checkbox` around the checkbox. Then add the `.custom-control-input` to the input with type="checkbox".

Tip: If you use labels for accompanying text, add the `.custom-control-label` class to it. Note that the value of the for attribute should match the id of the checkbox:

☐ Custom checkbox   ☐ Default checkbox

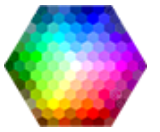## Example

```
<form class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control custom-checkbox"="tagcolor"
style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="checkbox" class="attributevaluecolor"
style="color:mediumblue">="custom-control-input" id="attributevaluecolor"
style="color:mediumblue">="customCheck" name="attributevaluecolor"
style="color:mediumblue">="example1"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control-label" for="attributevaluecolor"
style="color:mediumblue">="customCheck"="tagcolor" style="color:mediumblue">>Check
this custom checkbox</label class="tagcolor" style="color:mediumblue">>
  </div class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Custom Switch

To create a custom "toggle switch", wrap a container element, like <div>, with a class of `.custom-control` and `.custom-switch` around a checkbox. Then add the `.custom-control-input` class to the checkbox:

🔘 Toggle me

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control custom-switch"="tagcolor"
style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="checkbox" class="attributevaluecolor"
style="color:mediumblue">="custom-control-input" id="attributevaluecolor"
style="color:mediumblue">="switch1"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control-label" for="attributevaluecolor"
style="color:mediumblue">="switch1"="tagcolor" style="color:mediumblue">>Toggle
me</label class="tagcolor" style="color:mediumblue">>
  </div class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Custom Radio buttons

To create a custom radio button, wrap a container element, like <div>, with a class of `.custom-control` and `.custom-radio` around the radio button. Then add the `.custom-control-input`

to the input with type="radio".

Tip: If you use labels for accompanying text, add the `.custom-control-label` class to it. Note that the value of the for attribute should match the id of the radio:

○ Custom radio   ○ Default radio

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control custom-radio"="tagcolor"
style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="radio" class="attributevaluecolor"
style="color:mediumblue">="custom-control-input" id="attributevaluecolor"
style="color:mediumblue">="customRadio" name="attributevaluecolor"
style="color:mediumblue">="example1" value="attributevaluecolor"
style="color:mediumblue">="customEx"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control-label" for="attributevaluecolor"
style="color:mediumblue">="customRadio"="tagcolor"
style="color:mediumblue">>Custom radio</label class="tagcolor"
style="color:mediumblue">>
  </div class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Inline Custom Form Controls

If you want the custom form controls to sit side by side (inline), add the `.custom-control-inline` to the wrapper/container:

○ Custom radio   ○ Custom radio

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control custom-radio custom-control-
inline"="tagcolor" style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="radio" class="attributevaluecolor"
style="color:mediumblue">="custom-control-input" id="attributevaluecolor"
style="color:mediumblue">="customRadio" name="attributevaluecolor"
style="color:mediumblue">="example" value="attributevaluecolor"
style="color:mediumblue">="customEx"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control-label" for="attributevaluecolor"
style="color:mediumblue">="customRadio"="tagcolor"
style="color:mediumblue">>Custom radio 1</label class="tagcolor"
style="color:mediumblue">>
  </div class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control custom-radio custom-control-
inline"="tagcolor" style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="radio" class="attributevaluecolor"
style="color:mediumblue">="custom-control-input" id="attributevaluecolor"
style="color:mediumblue">="customRadio2" name="attributevaluecolor"
style="color:mediumblue">="example" value="attributevaluecolor"
style="color:mediumblue">="customEx"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-control-label" for="attributevaluecolor"
style="color:mediumblue">="customRadio2"="tagcolor"
style="color:mediumblue">>Custom radio 2</label class="tagcolor"
```

```
  style="color:mediumblue">>
    </div class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Custom Select Menu

To create a custom select menu, add the `.custom-select` class to the `<select>` element:

Custom Select Menu ⇕

Default select menu ⌄

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <select class="attributecolor" style="color:red"> name="attributevaluecolor"
style="color:mediumblue">="cars" class="attributevaluecolor"
style="color:mediumblue">="custom-select"="tagcolor" style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> selected="tagcolor"
style="color:mediumblue">>Custom Select Menu</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="volvo"="tagcolor"
style="color:mediumblue">>Volvo</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="fiat"="tagcolor" style="color:mediumblue">>Fiat</option
class="tagcolor" style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="audi"="tagcolor" style="color:mediumblue">>Audi</option
class="tagcolor" style="color:mediumblue">>
  </select class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Custom Select Menu Size

Use the `.custom-select-sm` class to create a small select menu and the `.custom-select-lg`
class for a large one:

Small Custom Select Menu ⇅

Default Custom Select Menu ⇅

Large Custom Select Menu ⇅

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <!-- Small -->
  <select class="attributecolor" style="color:red"> name="attributevaluecolor"
style="color:mediumblue">="cars" class="attributevaluecolor"
style="color:mediumblue">="custom-select custom-select-sm"="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> selected="tagcolor"
style="color:mediumblue">>Small Custom Select Menu</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="volvo"="tagcolor"
style="color:mediumblue">>Volvo</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="fiat"="tagcolor" style="color:mediumblue">>Fiat</option
class="tagcolor" style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="audi"="tagcolor" style="color:mediumblue">>Audi</option
class="tagcolor" style="color:mediumblue">>
  </select class="tagcolor" style="color:mediumblue">>

  <!-- Large -->
  <select class="attributecolor" style="color:red"> name="attributevaluecolor"
style="color:mediumblue">="cars" class="attributevaluecolor"
style="color:mediumblue">="custom-select custom-select-lg"="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> selected="tagcolor"
style="color:mediumblue">>Large Custom Select Menu</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="volvo"="tagcolor"
style="color:mediumblue">>Volvo</option class="tagcolor"
style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="fiat"="tagcolor" style="color:mediumblue">>Fiat</option
```

```
class="tagcolor" style="color:mediumblue">>
    <option class="attributecolor" style="color:red"> value="attributevaluecolor"
style="color:mediumblue">="audi"="tagcolor" style="color:mediumblue">>Audi</option
class="tagcolor" style="color:mediumblue">>
  </select class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Custom Range

To create a custom range menu, add the `.custom-range` class to an input with type="<range>":

Custom range

Default range

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <label class="attributecolor" style="color:red"> for="attributevaluecolor"
style="color:mediumblue">="customRange"="tagcolor"
style="color:mediumblue">>Custom range</label class="tagcolor"
style="color:mediumblue">>
  <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="range" class="attributevaluecolor"
style="color:mediumblue">="custom-range" id="attributevaluecolor"
style="color:mediumblue">="customRange" name="attributevaluecolor"
style="color:mediumblue">="points1"="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

Custom File Upload

To create a custom file upload, wrap a container element with a class of `.custom-file` around the input with type="file". Then add the `.custom-file-input` to it.

Tip: If you use labels for accompanying text, add the `.custom-file-label` class to it. Note that the value of the for attribute should match the id of the checkbox:

Choose file	Browse

Default file: Choose File No file chosen

Note that you also have to include some jQuery code if you want the name of the file to appear when you select a specific file:

Example

```
<form class="tagcolor" style="color:mediumblue">>
  <div class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-file"="tagcolor" style="color:mediumblue">>
    <input class="attributecolor" style="color:red"> type="attributevaluecolor"
style="color:mediumblue">="file" class="attributevaluecolor"
style="color:mediumblue">="custom-file-input" id="attributevaluecolor"
style="color:mediumblue">="customFile"="tagcolor" style="color:mediumblue">>
    <label class="attributecolor" style="color:red"> class="attributevaluecolor"
style="color:mediumblue">="custom-file-label" for="attributevaluecolor"
style="color:mediumblue">="customFile"="tagcolor" style="color:mediumblue">>Choose
file</label class="tagcolor" style="color:mediumblue">>
  </div class="tagcolor" style="color:mediumblue">>
</form class="tagcolor" style="color:mediumblue">>
```

```
<script class="tagcolor" style="color:mediumblue">>
// Add the following code if you want the name of the file appear on select
$(".custom-file-input").on("change", function() {
  var fileName = $(this).val().split("\\").pop();
  $(this).siblings(".custom-file-label").addClass("selected").html(fileName);
});
</script class="tagcolor" style="color:mediumblue">>
```

Try it Yourself »

Top Tutorials

HTML Tutorial
CSS Tutorial
JavaScript Tutorial
How To Tutorial
SQL Tutorial
Python Tutorial
W3.CSS Tutorial
Bootstrap Tutorial
PHP Tutorial
Java Tutorial
C++ Tutorial
jQuery Tutorial

Top References

HTML Reference
CSS Reference
JavaScript Reference
SQL Reference
Python Reference
W3.CSS Reference
Bootstrap Reference
PHP Reference
HTML Colors
Java Reference
Angular Reference
jQuery Reference

Top Examples

HTML Examples
CSS Examples
JavaScript Examples
How To Examples
SQL Examples
Python Examples
W3.CSS Examples
Bootstrap Examples
PHP Examples
Java Examples
XML Examples
jQuery Examples

Get Certified

HTML Certificate
CSS Certificate
JavaScript Certificate
Front End Certificate
SQL Certificate
Python Certificate
PHP Certificate
jQuery Certificate
Java Certificate
C++ Certificate
C# Certificate
XML Certificate

FORUM   ABOUT