



Bootstrap



Introduction



- Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.
- Bootstrap is a free and open-source tool collection for creating responsive websites and web applications.
- It solves many problems which we had once, one of which is the cross-browser compatibility issue.





Version



- Bootstrap was created at Twitter in mid-2010 by Mark Otto and Jacob Thornton.
- August 19, 2011, released as an open source project.
- January 31, 2012, Bootstrap 2 was released.
- October 29, 2014, Bootstrap 4.
- December 2020, Bootstrap 5 Beta.





- Advantages of Bootstrap:
 - **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
 - **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
 - **Mobile-first approach:** In Bootstrap 3, mobile-first styles are part of the core framework
 - **Browser compatibility:** Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Safari, and Opera)





Quick start



- Use CDN:

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.
min.css">
```

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" ></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.m
in.js" ></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.
js"></script>
```





Quick start



– Download:

Link: <https://github.com/twbs/bootstrap/releases>



6



Content



- **Reboot:** a collection of element-specific CSS changes in a single file, kickstart. Providing many HTML elements with somewhat opinionated styles using only element selectors
- **Page defaults:** The `<html>` and `<body>` elements are updated to provide better page-wide defaults.
- **Native font stack:** The default web fonts (Helvetica Neue, Helvetica, and Arial) have been dropped in Bootstrap 4 and replaced with a “native font stack” for optimum text rendering on every device and OS.





- **Headings and paragraphs:** All heading elements—e.g., `<h1>`—and `<p>` are reset to have their margin-top removed. Headings have margin-bottom: .5rem added and paragraphs margin-bottom: 1rem for easy spacing.
- **Lists:** All lists—``, ``, and `<dl>`—have their margin-top removed and a margin-bottom: 1rem. Nested lists have no margin-bottom.





Content



- **Preformatted text:** The `<pre>` element is reset to remove its margin-top and use rem units for its margin-bottom.
- **Tables:** Tables are slightly adjusted to style `<caption>`, collapse borders, and ensure consistent text-align throughout. Additional changes for borders, padding, and more come with the `.table` class.





Content



- **Forms:** Various form elements have been rebooted for simpler base styles. Here are some of the most notable changes:

`<fieldset>`s have no borders, padding, or margin so they can be easily used as wrappers for individual inputs or groups of inputs.

`<legend>`s, like fieldsets, have also been restyled to be displayed as a heading of sorts.

`<label>`s are set to display: inline-block to allow margin to be applied.

`<input>`s, `<select>`s, `<textarea>`s, and `<button>`s are mostly addressed by Normalize, but Reboot removes their margin and sets line-height: inherit, too.

`<textarea>`s are modified to only be resizable vertically as horizontal resizing often “breaks” page layout.



Three hexagonal icons arranged in a triangular pattern. The top-left hexagon shows two hands shaking in a firm grip, symbolizing a business deal or partnership. The top-right hexagon depicts a server room with rows of blue-lit server racks and binary code (0s and 1s) floating in the air, representing technology and data. The bottom hexagon shows a globe with yellow lines connecting various points, symbolizing global communication and networks.

Component



Alerts



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

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

A simple primary alert—check it out!

A simple secondary alert—check it out!

A simple success alert—check it out!

A simple danger alert—check it out!

A simple warning alert—check it out!

A simple info alert—check it out!

A simple light alert—check it out!

A simple dark alert—check it out!



12



Buttons



- Bootstrap includes several predefined button styles, each serving its own semantic purpose, with a few extras thrown in for more control.

Primary Secondary Success Danger Warning Info Light Dark Link

```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>

<button type="button" class="btn btn-link">Link</button>
```

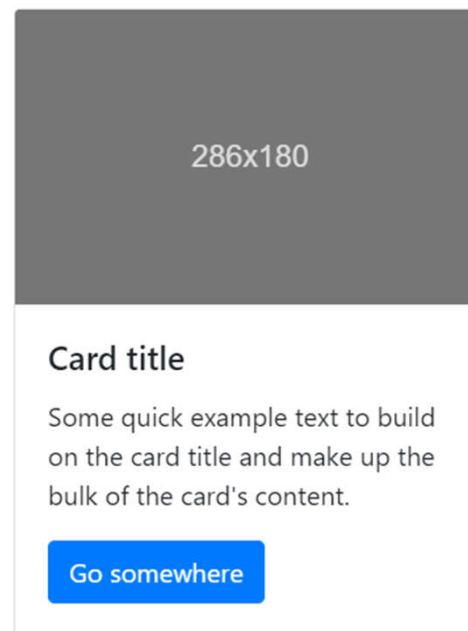




Cards



- A **card** is a flexible and extensible content container. It includes options for headers and footers, a wide variety of content, contextual background colors, and powerful display options.



14



Dropdowns



- Toggle contextual overlays for displaying lists of links and more with the Bootstrap dropdown plugin





Forms



- Form controls - Textual form controls—like `<input>`s, `<select>`s, and `<textarea>`s—are styled with the `.form-control` class. Included are styles for general appearance, focus state, sizing, and more.

Email address

Example select

1

Example multiple select

1234

Example textarea





- Example file input

Example file input

Choose File No file chosen

```
<form>
  <div class="form-group">
    <label for="exampleFormControlFile1">Example file input</label>
    <input type="file" class="form-control-file" id="exampleFormControlFile1">
  </div>
</form>
```

Copy





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

`.form-control-lg`

Default input

`.form-control-sm`

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





Form



– Range Inputs

Readonly input here...

```
<input class="form-control" type="text" placeholder="Readonly input here..." readonly>
```

– Readonly field plain text: adding class `.form-control-plaintext`





Form



– Readonly field

Example Range input



```
<form>
  <div class="form-group">
    <label for="formControlRange">Example Range input</label>
    <input type="range" class="form-control-range" id="formControlRange">
  </div>
</form>
```





Form



– Checkboxes

- ☐ Default checkbox
- ☐ Disabled checkbox

```
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="defaultCheck1">
  <label class="form-check-label" for="defaultCheck1">
    Default checkbox
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="checkbox" value="" id="defaultCheck2" disabled>
  <label class="form-check-label" for="defaultCheck2">
    Disabled checkbox
  </label>
</div>
```





Form



– Radios

- ☒ Default radio
- ☐ Second default radio
- ☐ Disabled radio

```
<div class="form-check">
  <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios1" value="option1" />
  <label class="form-check-label" for="exampleRadios1">
    Default radio
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios2" value="option2" />
  <label class="form-check-label" for="exampleRadios2">
    Second default radio
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="exampleRadios" id="exampleRadios3" value="option3" />
  <label class="form-check-label" for="exampleRadios3">
    Disabled radio
  </label>
</div>
```





Form Layout



- 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.
 - Form groups.
 - Form grid.
 - Form row.
 - Horizontal form.
 - Inline forms.





Form Layout



– Form groups.

Example label

Example input

Another label

Another input

```
<form>
  <div class="form-group">
    <label for="formGroupExampleInput">Example label</label>
    <input type="text" class="form-control" id="formGroupExampleInput" placeholder="Example input">
  </div>
  <div class="form-group">
    <label for="formGroupExampleInput2">Another label</label>
    <input type="text" class="form-control" id="formGroupExampleInput2" placeholder="Another input">
  </div>
</form>
```



24



Form Layout



– Form grid.

```
<form>
  <div class="row">
    <div class="col">
      <input type="text" class="form-control" placeholder="First name">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
    </div>
  </div>
</form>
```





Form Layout



– Form row.

```
<form>
  <div class="form-row">
    <div class="col">
      <input type="text" class="form-control" placeholder="First name">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
    </div>
  </div>
</form>
```





Form Layout



– Horizontal form.

Email

Email

Password

Password

Radios

☒ First radio
☐ Second radio
 Third disabled radio

Checkbox

☐ Example checkbox

Sign in



27



Form Layout



– Inline forms.

```
<form class="form-inline">
  <label class="sr-only" for="inlineFormInputName2">Name</label>
  <input type="text" class="form-control mb-2 mr-sm-2" id="inlineFormInputName2" placeholder="Name" />

  <label class="sr-only" for="inlineFormInputGroupUsername2">Username</label>
  <div class="input-group mb-2 mr-sm-2">
    <div class="input-group-prepend">
      <div class="input-group-text">@</div>
    </div>
    <input type="text" class="form-control" id="inlineFormInputGroupUsername2" placeholder="Username" />
  </div>

  <div class="form-check mb-2 mr-sm-2">
    <input class="form-check-input" type="checkbox" id="inlineFormCheck">
    <label class="form-check-label" for="inlineFormCheck">
      Remember me
    </label>
  </div>

  <button type="submit" class="btn btn-primary mb-2">Submit</button>
</form>
```



28



Validation



- Provide valuable, actionable feedback to your users with HTML5 form validation—available in all our supported browsers. Choose from the browser default validation feedback, or implement custom messages with our built-in classes and starter JavaScript.





Validation



– 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. This way, you may choose when to activate them (typically after form submission is attempted).





Validation



- 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.
- Due to constraints in how CSS works, we cannot (at present) apply styles to a **<label>** that comes before a form control in the DOM without the help of custom JavaScript.
- All modern browsers support the **constraint validation API**, a series of JavaScript methods for validating form controls.





Validation




– Browser defaults

First name

Last name

Username

City

 Please fill out this field.



32



Validation



- Custom styles
 - For custom Bootstrap form validation messages, you'll need to add the novalidate boolean attribute to your <form>. This disables the browser default feedback tooltips, but still provides access to the form validation APIs in JavaScript.

The screenshot shows a form with five input fields. The first two, 'First name' (containing 'Mark') and 'Last name' (containing 'Otto'), have green borders and green feedback messages 'Looks good!'. The 'Username' field, which has a dropdown menu showing '@', has a red border and a red feedback message 'Please choose a username.'. The 'City' field has a red border and a red feedback message 'Please provide a valid city.'.



33



Modal



- Use Bootstrap's JavaScript modal plugin to add dialogs to your site for lightboxes, user notifications, or completely custom content.





Modal - How it works



- Before getting started with Bootstrap's modal component, be sure to read the following as our menu options have recently changed.
 - Modals are built with HTML, CSS, and JavaScript. They're positioned over everything else in the document and remove scroll from the `<body>` so that modal content scrolls instead.
 - Clicking on the modal "backdrop" will automatically close the modal.
 - Bootstrap only supports one modal window at a time. Nested modals aren't supported as we believe them to be poor user experiences.





Modal - How it works



- Modals use position: fixed, which can sometimes be a bit particular about its rendering. Whenever possible, place your modal HTML in a top-level position to avoid potential interference from other elements. You'll likely run into issues when nesting a .modal within another fixed element.
- Once again, due to position: fixed, there are some caveats with using modals on mobile devices. See our browser support docs for details.
- Due to how HTML5 defines its semantics, the autofocus HTML attribute has no effect in Bootstrap modals. To achieve the same effect, use some custom JavaScript:

```
$('#myModal').on('shown.bs.modal', function () {  
  $('#myInput').trigger('focus')  
})
```



36



Modal



Launch demo modal

Copy

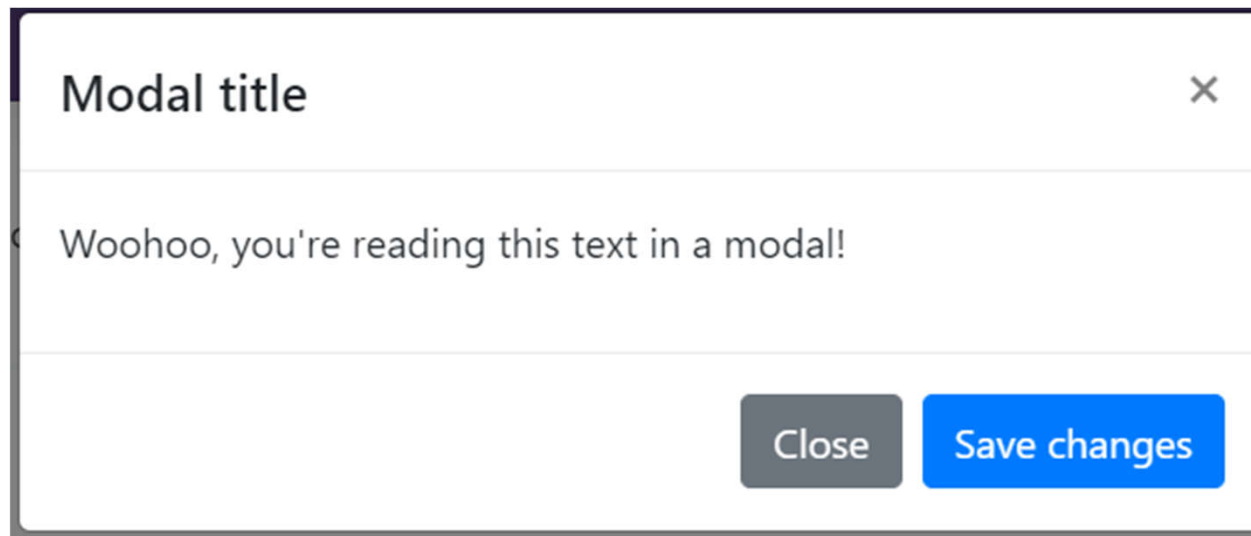
```
<!-- Button trigger modal -->
<button type="button" class="btn btn-primary" data-toggle="modal" data-target="#exampleModal">
  Launch demo modal
</button>

<!-- Modal -->
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        ...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```

37



Modal



38



LAYOUT





Container



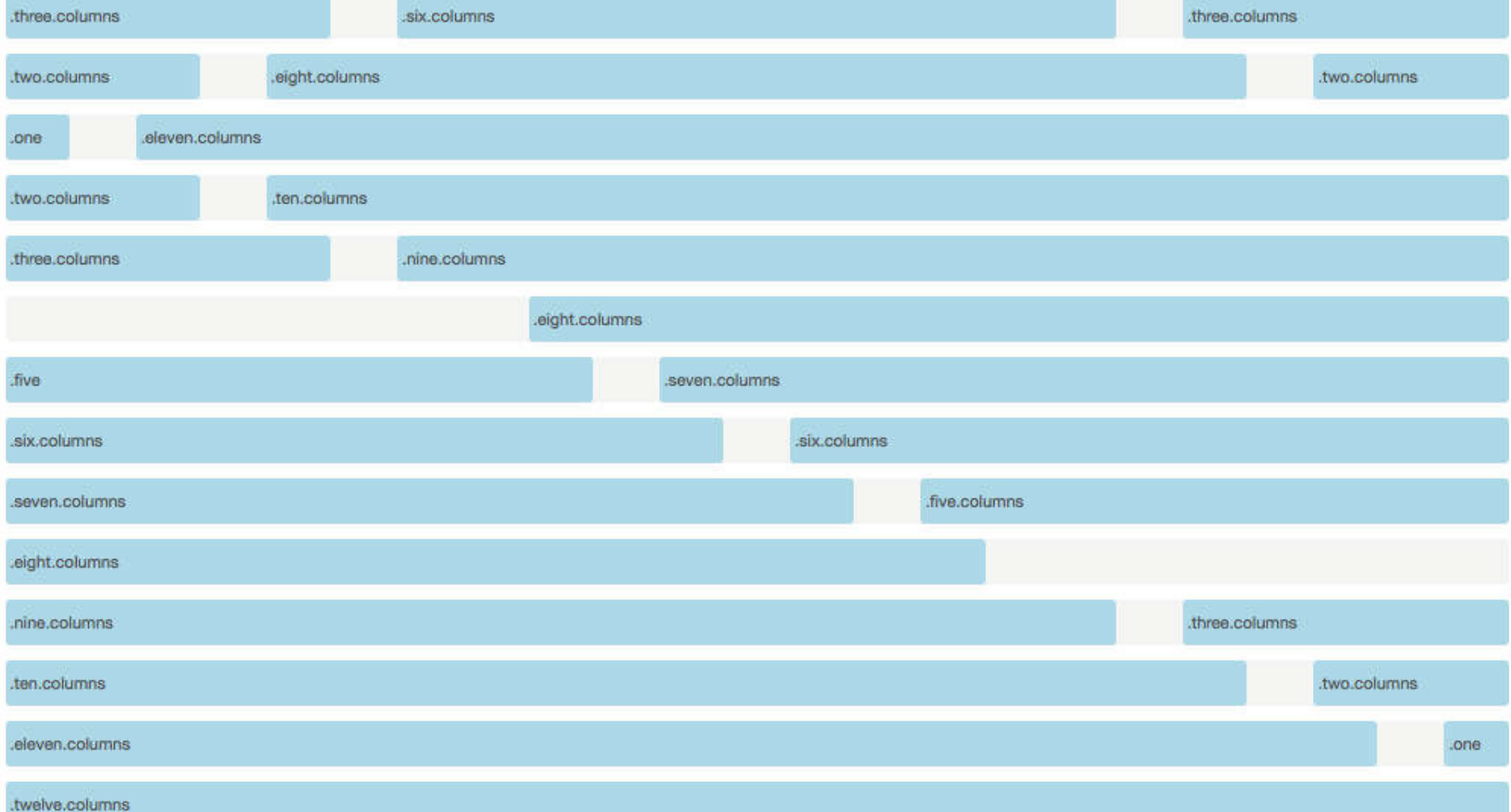
- Bootstrap also requires a containing element to wrap site contents.
- There are two container classes to choose from:
 - The `.container` class provides a responsive **fixed width container**.
 - The `.container-fluid` class provides a **full width container**, spanning the entire width of the viewport.
- **Note:** Containers are not nestable (you cannot put a container inside another container).



41



Bootstrap Grids



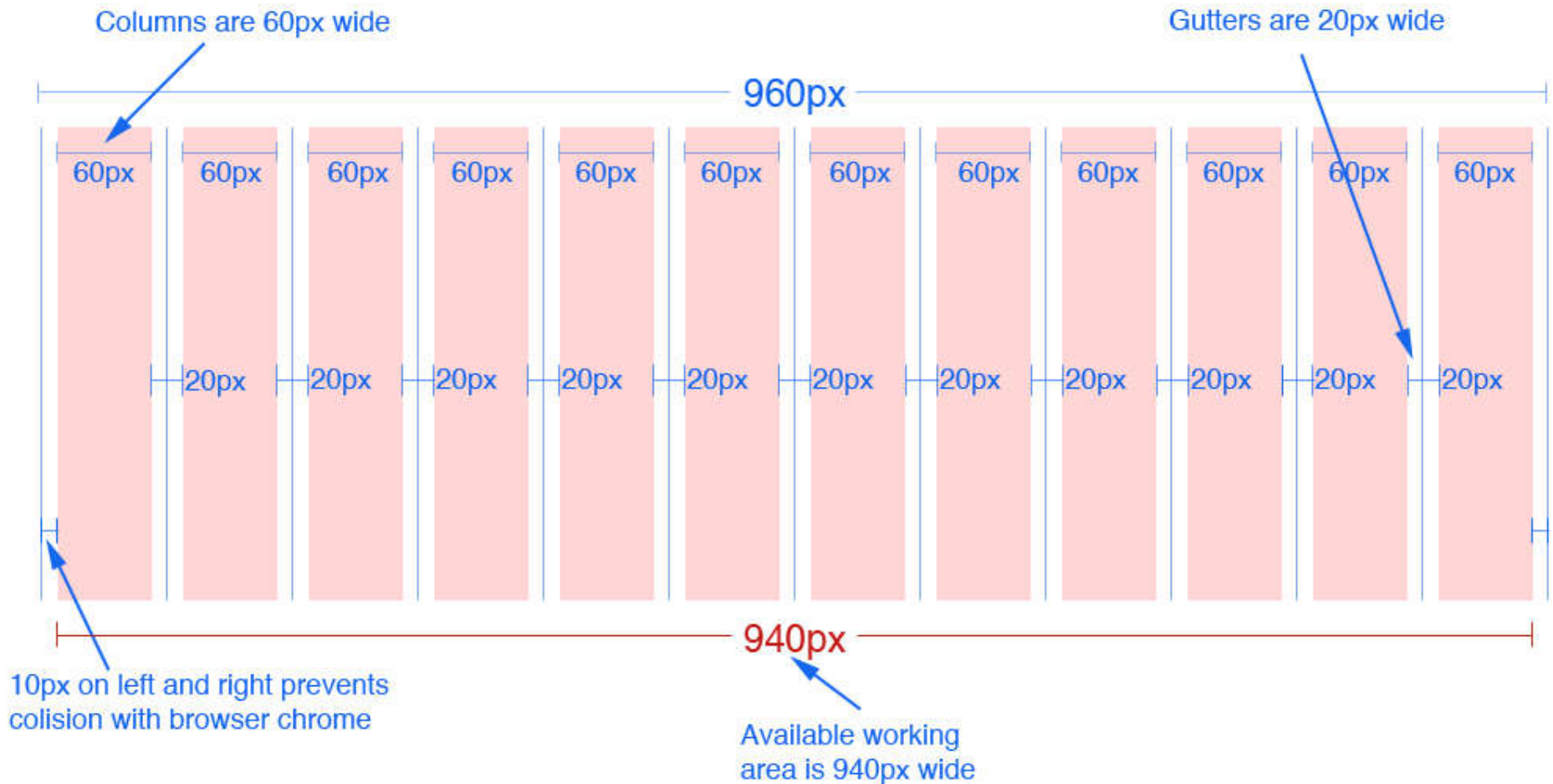
```
8</div>
```

```
<div class="col-md-4">Span 4</div><div class="col-md-4">Span  
4</div>
```

```
<div class="col-md-4">Span 4</div>
```



12 column version





Grid Classes



- The Bootstrap grid system has four classes:
 - xs (for phones)
 - sm (for tablets)
 - md (for desktops)
 - lg (for larger desktops)
- The classes above can be combined to create more dynamic and flexible layouts.





Basic Structure of a Bootstrap Grid



- First; create a row (<div class="row">). Then, add the desired number of columns (tags with appropriate .col-*-* classes). Note that numbers in .col-*-* should always add up to 12 for each row.

```
<div class="container">
  <div class="row">
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
  </div>
</div>
```



Responsive breakpoints



- Since Bootstrap is developed to be mobile first, we use a handful of **media queries** to create sensible breakpoints for our layouts and interfaces. These breakpoints are mostly based on minimum viewport widths and allow us to scale up elements as the viewport changes.
- Bootstrap primarily uses the following media query ranges—or breakpoints—in our source Sass files for our layout, grid system, and components.



46



Responsive breakpoints



```
// Extra small devices (portrait phones, less than  
576px)  
// No media query for `xs` since this is the default in  
Bootstrap  
  
// Small devices (landscape phones, 576px and up)  
@media (min-width: 576px) { ... }  
  
// Medium devices (tablets, 768px and up)  
@media (min-width: 768px) { ... }  
  
// Large devices (desktops, 992px and up)  
@media (min-width: 992px) { ... }  
  
// Extra large devices (large desktops, 1200px and up)  
@media (min-width: 1200px) { ... }
```



47



Q & A



48