Learning Outcome

Bootstrap

## 

## 

## 

# Overview of Bootstrap

Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites. It solves many problems which we had once, one of which is the cross-browser compatibility issue. Nowadays, the websites are perfect for all the browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones). All thanks to Bootstrap developers -Mark Otto and Jacob Thornton of Twitter, though it was later declared to be an open-source project.

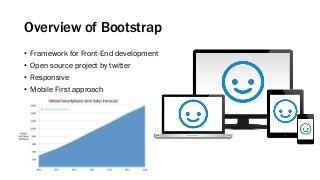


Image : Overview of Bootstrap

Reference: <https://image.slidesharecdn.com/bootstrap-150705164322-lva1-app6892/85/bootstrap-3-320.jpg?cb=1436114649>

## Advantages of Bootstrap

The first thing is that now days people use mobiles, laptops, tabs, personal computers. If we have to reach our website to all people, we should spread our business more and more. The website should be such that everyone can read and view the content of your website well. So now we will not make different designs for different devices because may be a new Mobile Screen Ratio may come tomorrow, or if a new laptop screen size is launched, then our website will not work well on it. So to avoid all this mess Bootstrap is there. In which you have to create a website just once, then whatever the screen viewer opens the website on any screen, the UI (User Interface) of the website will be shown well. And people create and sell templates from bootstrap which people use in WordPress, Joomla, Drupal etc.



Image : Benefits of Bootstrap

Reference: <https://pbs.twimg.com/media/EonHh_wUwAEcsP2?format=png&name=large>

### Saving time

This is the biggest advantage of Bootstrap that using it increases the development speed and gets the work done in a very short time. On the contrary, if you want to create a responsive design without bootstrap by yourself, then you have a lot of time in it. May take

### Easy to use

If you have basic knowledge of HTML and CSS, then you can easily use Bootstrap.

### Responsive Design

Through bootstrap you can easily create responsive design. If your website is responsive then it adjusts itself according to the screen size in any platform or device like desktop, laptop, mobile etc.

### Cross Browser Compatible

Bootstrap is designed in such a way that our web-page will look the same in almost all modern browsers like Firefox, Chrome, Internet Explorer, Opera etc.

### Open Source

The best thing about it is that you can use it for free.

### Customization

Bootstrap is easy to customize, customize means that if you do not want all the features of bootstrap, you can only use a few features. On Bootstrap’s website give you a lot of options (Nav bar, Table, Form, Button, Model, Dropdown, Badges, etc.) Let you tick what you want, remove the tick from what you don’t want and download and use it. Do it All the CSF classes are already created in Bootstrap, we just have to use those classes in our website and very easily and in a very short time. We create a beautiful website in this way, Bootstrap reduces the time taken in our website development.

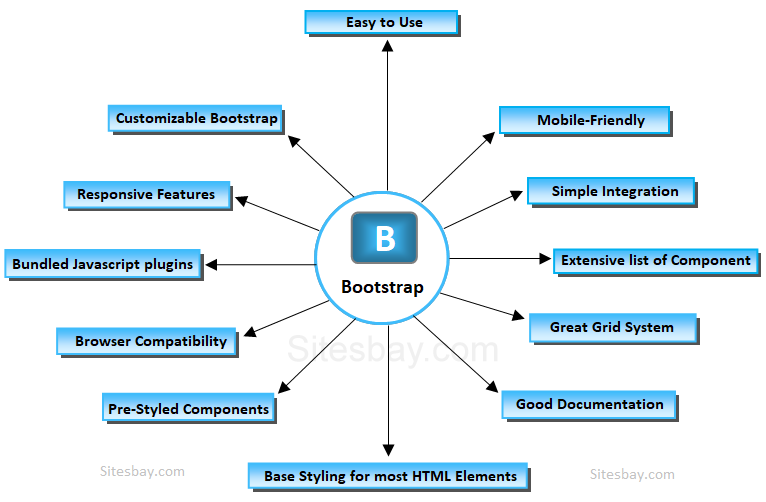


Image : Advantages of Bootstrap

Reference: <https://www.sitesbay.com/bootstrap/images/features-of-bootstrap.png>

## Bootstrap Container

In Bootstrap, container is used to set the content's margins dealing with the responsive behaviors of your layout. It contains the row elements and the row elements are the container of columns (known as grid system).

The container class is used to create boxed content.

Containers are used to pad the content inside of them, and there are two container classes available:

* 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

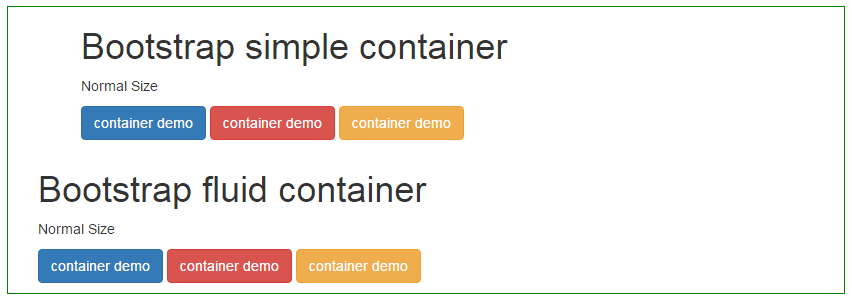


Image : Bootstrap containers

Reference: <https://www.jquery-az.com/wp-content/uploads/2016/11/12.1-Bootstrap-container.png>

### Fixed Container

Use the .container class to create a responsive, fixed-width container.

Note that its width (max-width) will change on different screen sizes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Extra small <576px | Small≥576px | Medium≥768px | Large≥992px | Extra large≥1200px |
| Max-width | 100% | 540px | 720px | 960px | 1140px |

### Fluid Container

Use the .container-fluid class to create a full width container, that will always span the entire width of the screen (width is always 100%)

## How to use Bootstrap 5 on a webpage:

There are two ways to include Bootstrap on the website.

* Include Bootstrap from the CDN link.
* Download Bootstrap from getbootstrap.com and use it.

### Bootstrap 5 CDN

If you don't want to download and host Bootstrap 5 yourself, you can include it from a CDN (Content Delivery Network).

jsDelivr provides CDN support for Bootstrap's CSS and JavaScript:

### MaxCDN:

<!-- Latest compiled and minified CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Latest compiled JavaScript -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>

### Downloading Bootstrap 5

If you want to download and host Bootstrap 5 yourself, go to <https://getbootstrap.com/>, and follow the instructions there.

## Bootstrap Components

Bootstrap provides a variety of customizable and reusable components which makes the development faster and easier. They are heavily based on the base modifier nomenclature i.e. the base class has many groups of shared properties together while the modifier class has a group of individual styles. For example, .btn is a base class and .btn-primary or .btn-success is a modifier class. The bootstrap components range from alerts, buttons, badges, cards to various other components.



Image : Bootstrap components

Reference: <https://media.geeksforgeeks.org/wp-content/uploads/20210819203105/202108199-660x372.jpg>

### List of components:

### Jumbotron

It simply put extra attention to particular content or information by making it larger and more eye-catching.

### Alerts

It is a popup with a predefined message that appears after a particular action.

### Buttons

It is customized buttons that are used to perform an action in the form, dialogue box, etc. They are in multiple states, sizes and have predefined styles.

### Button group

It is a group of buttons aligned in a single line and they can be arranged both vertically as well as horizontally.

### Badge

It Is a labeling component that is used to add additional information.

### Progress Bar

It is used to show the progress of a particular operation with a custom progress bar. They have text labels, stacked bars, and animated backgrounds.

### Spinner

The spinner displays the loading state of websites or projects. They are built with HTML, CSS and don’t require any JavaScript.

### Scrollspy

It keeps updating the navigation bar to the currently active link based on the scroll position in the viewport.

### List group

It is used to display an unordered series of content in a proper way.

### Card

It provides a customizable, extensible, and flexible content container.

### Dropdown

It is used to drop the menu in the format of a list of links, they are contextual and toggleable overlays.

### Navs

It is used to create a basic and simple navigation menu with a .nav base class.

### Navbar

The navigation bar is the headers at the top of a website or webpage.

### Forms

Forms are used to take multiple inputs at once from the user. Bootstrap has two layouts available stacked and inline.

### Input groups

They have extended form controls by adding a button, button group or text on either side of inputs.

### Toast

It displays a message for a small amount of time, a few seconds. They are alert messages designed to imitate push notifications popular in desktop and mobile systems.

### Tooltip

It provides small information about the element/link when the mouse hovers over the element.

### Popovers

It displays extra information about the element/link when clicked on it.

### Collapse

It is a JavaScript plugin that is used to show or hide the content.

### Modal

It is a small popup window positioned over the actual window.

### Pagination

It is used to easily navigate between different pages, a large block of connected links is used for making them accessible.

### Media Object

The Media object is used for repetitive and complex components like tweets or blogs. The images or videos are placed/aligned to the left or the right of the content.

## Advance Components of Bootstrap

Bootstrap is already pre-packaged with a huge selection of useful tools, extensions and components, and is more than enough to kick-start most web design or web application projects. But there will be times when the bundled basic components are not quite enough for what you need. That is where this post comes in. We have assembled 45 extensions, plugins, addons and components that will allow you to extend Bootstrap even further.

### Accessibility

This plugin adds accessibility mark-up to the default components of Bootstrap. Components include: Alert, Tooltip, Popover, Modal Dialog, Dropdown Menu, Tab Panel, Collapse and Carousel.

### Breadcrumbs

Bootstrap-Breadcrumb: A Bootstrap JavaScript plugin that allows you to programmatically manipulate breadcrumb navigation.

### Calendar

Bootstrap-Calendar: A Full view calendar based on Bootstrap.

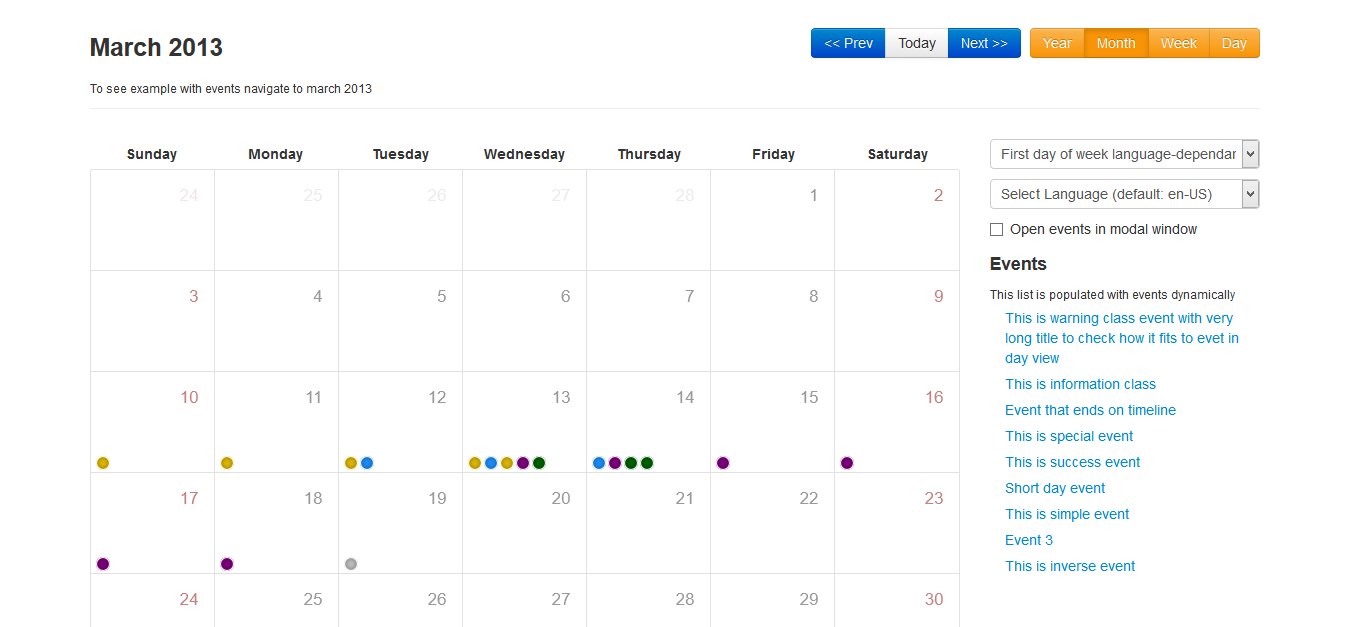


Image : Advance components of Bootstrap

Reference: <https://www.drupal.org/files/project-images/Twitter%20Bootstrap%20jQuery%20Calendar.png>

### Carousel

Bootstrap Modal Carousel – A collection of plugins for displaying a carousel in fullscreen modal window.

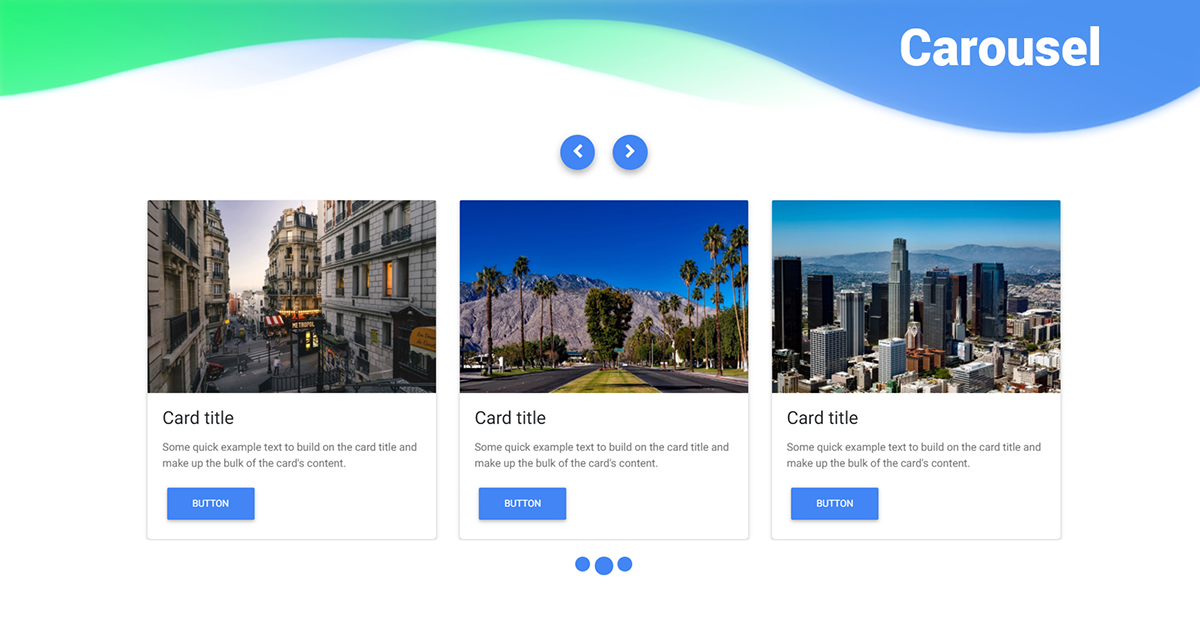


Image : Advance components of Bootstrap

Reference: <https://mdbcdn.b-cdn.net/wp-content/uploads/2017/12/carousel.jpg>

### Checkbox

prettyCheckable – A jQuery plugin for replacing the default checkboxes and radio inputs.

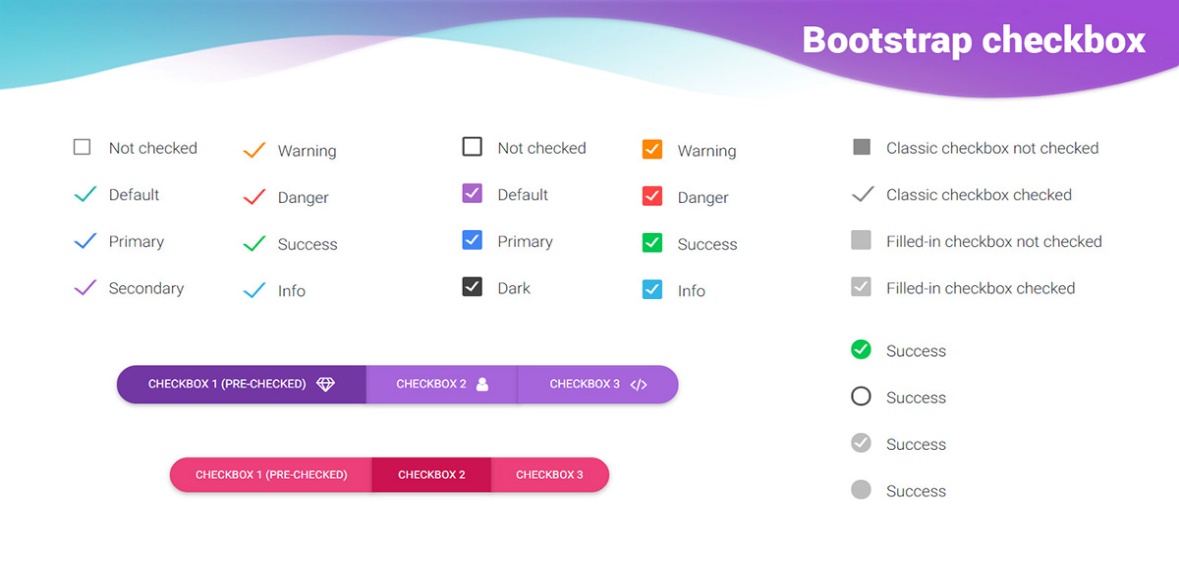


Image : Advance components of Bootstrap

Reference: <https://mdbcdn.b-cdn.net/wp-content/uploads/2017/07/bootstrap-checkbox.jpg>

### Color Picker

Simple Color Picker – A very simple and lightweight (200 lines of JavaScript and 100 lines of CSS) jQuery color picker for Bootstrap.

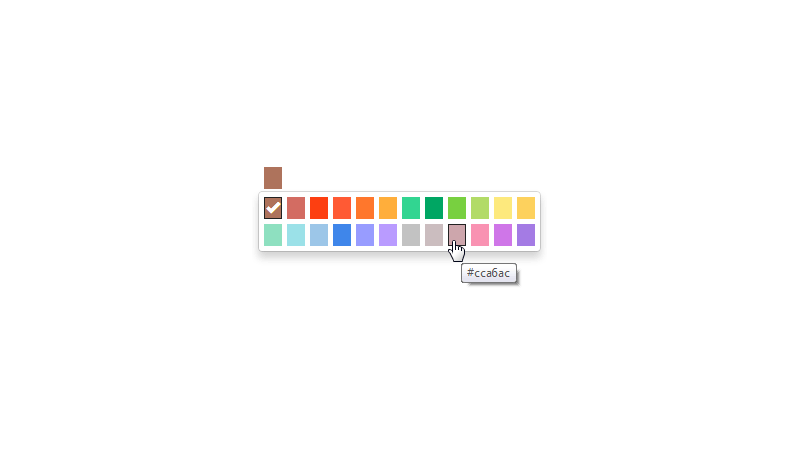


Image : Advance components of Bootstrap

Reference: <https://freefrontend.com/assets/img/jquery-color-picker-plugins/jquery-simplecolorpicker.png>

### Combobox

Bootstrap Combobox – A combobox plugin that integrates well with Bootstrap.

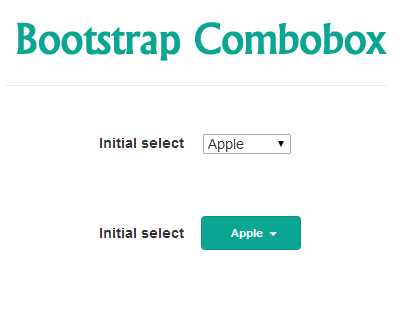


Image : Advance components of Bootstrap

Reference: <https://jquery-plugins.net/image/plugin/bootstrap-combobox.png>

### Contact Form

Bootstrap-Contact – A simple PHP contact form using Bootstrap and the jQuery validation plugin.

jQuery Gridform – A jQuery plugin for creating complex table-based forms with Bootstrap.

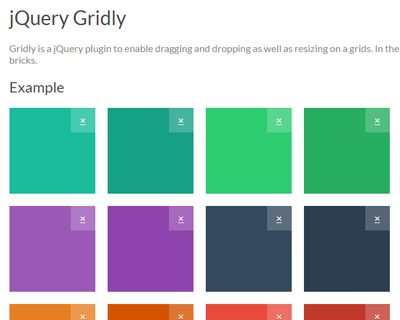


Image : Advance components of Bootstrap

Reference: <https://www.htmllion.com/img/jquery-plugins/jQuery-gridly.jpg>

### Datepicker

Datepicker – A plugin for adding a datepicker field to any element.

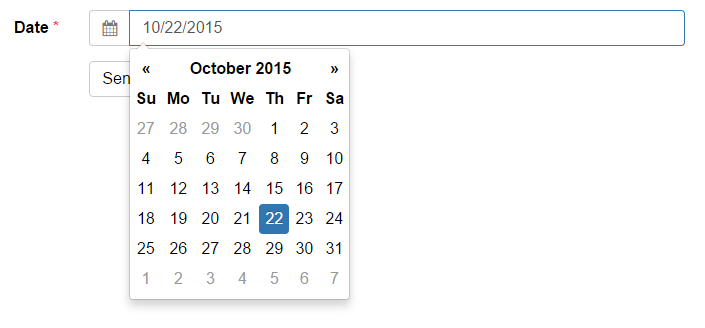


Image : Advance components of Bootstrap

Reference: <https://formden.com/static/assets/img/posts/date-picker/example_date.png>

Date Range Picker – This date range picker component creates a drop-down from which you can select a range of dates.

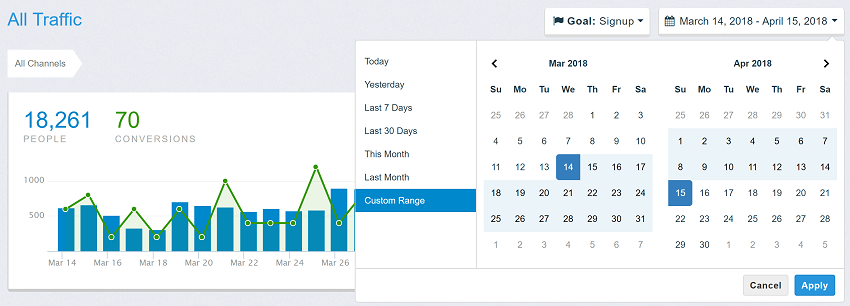


Image : Advance components of Bootstrap

Reference:<https://camo.githubusercontent.com/d7265660ea14a97c52ab3f6ce8684294b1df30892fcbd596f2c961ca8d1bfce2/68747470733a2f2f692e696d6775722e636f6d2f5554526c6161722e706e67>

ClockPicker – A unique clock-style timepicker for Bootstrap.

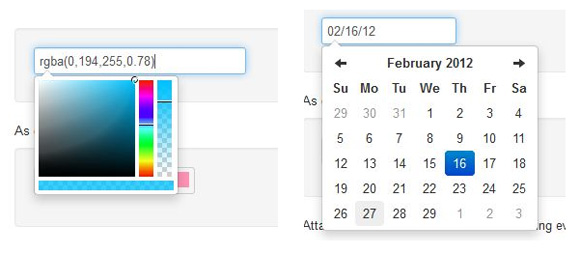


Image : Advance components of Bootstrap

Reference: <http://www.webappers.com/img/2012/07/bootstrap-datepicker-calendar.jpg>

### Dialog Boxes & Alerts

Bootbox.js – A small JS library that allows you to create programmatic dialog boxes using Bootstrap’s modals.



Image : Advance components of Bootstrap

Reference: <https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRk-zNcmg5V4KIh1eaWwKwdTlFjAMBEETQ1RA&usqp=CAU>

Bootstrap Prompts – An plugin to replace the alert(), prompt(),confirm() notifications on the browser when using Twitter Bootstrap with modals.

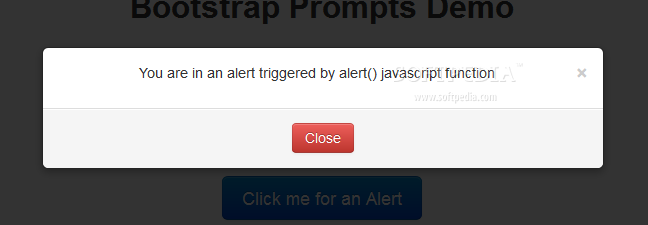


Image : Advance components of Bootstrap

Reference: <https://scripts-cdn.softpedia.com/screenshots/bootstrap-prompts_1.png>

Bootstrap Confirmation – A plugin that replaces popovers with confirmation dialogs.

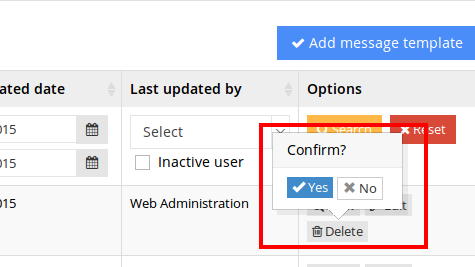


Image : Advance components of Bootstrap

Reference: <https://i.stack.imgur.com/cut2D.png>

### File Upload

jQuery File Upload – A file upload widget which features multiple file selection, drag & drop, progress bars, validation and preview images.

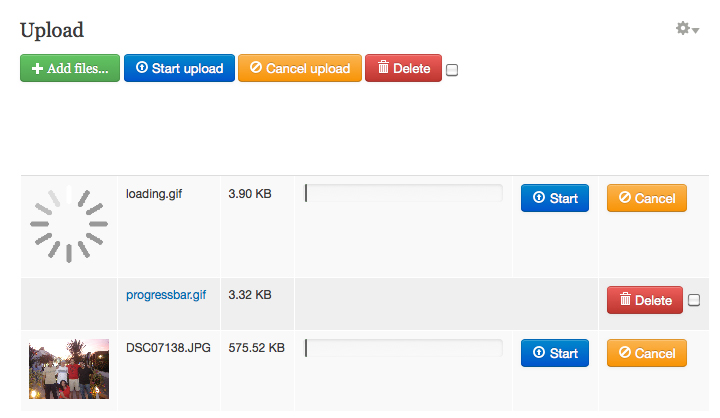


Image : Advance components of Bootstrap

Reference: <https://www.drupal.org/files/project-images/jquery_file_upload.jpg>

### Form Validation

BootstrapValidator – A jQuery plugin for vaildating forms within Bootstrap.

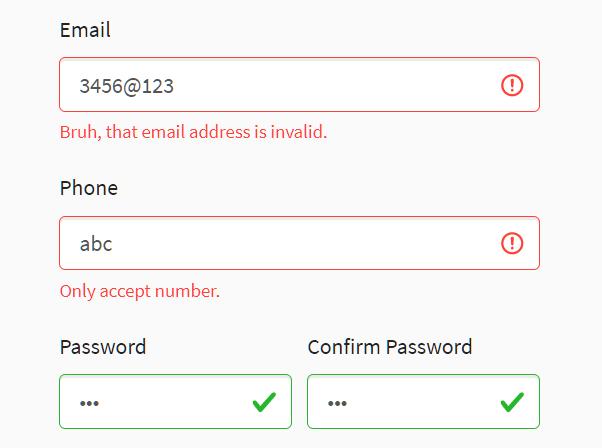


Image : Advance components of Bootstrap

Reference: <https://i0.wp.com/www.cssscript.com/wp-content/uploads/2021/02/Bootstrap-Form-Validation-Library-Without-jQuery-Native-Validator.png?fit=602%2C448&ssl=1>

jqBootstrapValidation – Another jQuery validation framework for Bootstrap forms.

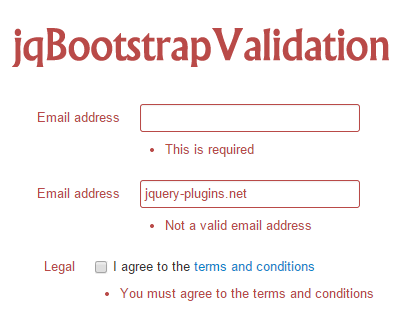


Image : Advance components of Bootstrap

Reference: <https://jquery-plugins.net/image/plugin/jqbootstapvalidation-jquery-validation-plugin-for-bootstrap.png>

Validator – A simple and user-friendly form validator plugin for Bootstrap.

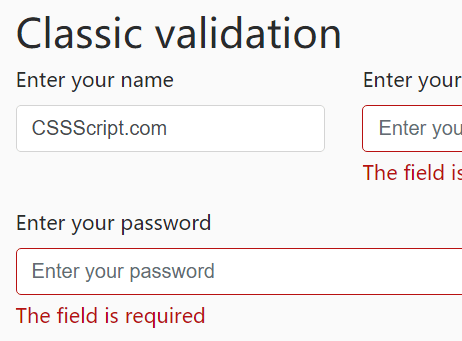


Image : Advance components of Bootstrap

Reference: <https://i0.wp.com/www.cssscript.com/wp-content/uploads/2017/03/Just-validate.png?fit=462%2C341&ssl=1>

HTML Tables

Tablecloth.js – Building off Bootstrap, this is is a jQuery plugin that helps you easily style HTML tables.

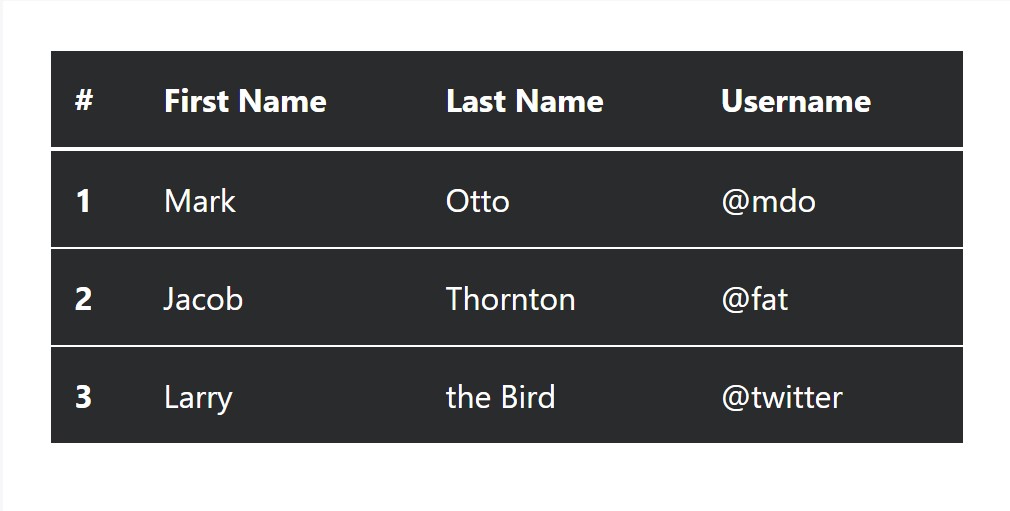


Image : Advance components of Bootstrap

Reference: <http://formoid.com/articles/data/upload/2017/03/1inverse-bootstrap-table.jpg>

Navigable Table – A Bootstrap plugin for smooth navigation across table inputs.

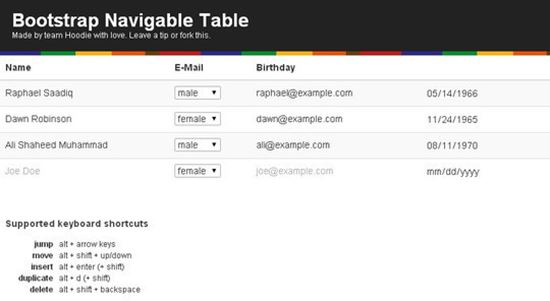


Image : Advance components of Bootstrap

Reference: <https://designposts.net/wp-content/uploads/2014/09/Bootstrap-Navigable-Table.jpg>

### Image Gallery

Image Gallery – This plugin shows images and videos in the modal dialog of the Bootstrap. It features swipe, mouse & keyboard navigation, transition effects, Fullscreen support and on-demand content loading.

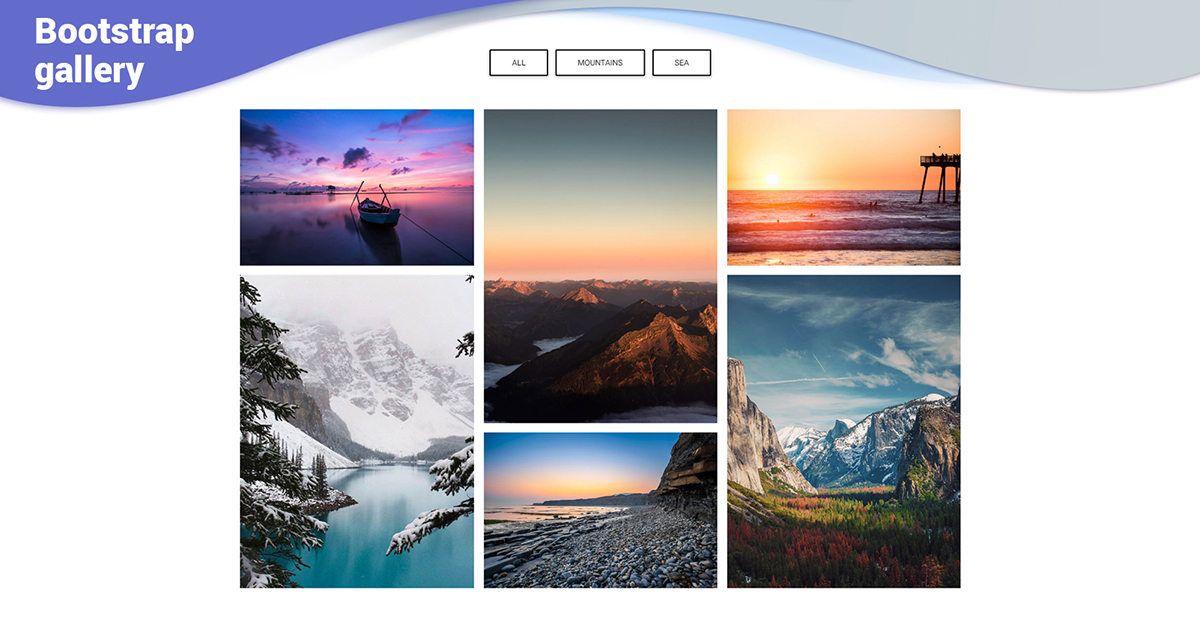


Image : Advance components of Bootstrap

Reference: <https://mdbcdn.b-cdn.net/wp-content/uploads/2017/09/gallery-fb.jpg>

### In-Place Editing

X-editable – A library that allows you to create editable elements on your Bootstrap page.

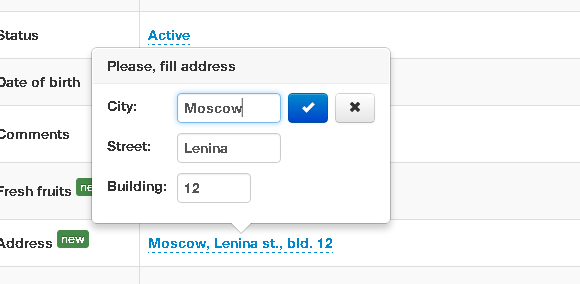


Image : Advance components of Bootstrap

Reference: <http://www.webappers.com/img/2012/12/editable-fields.png>

### Layout Grid

jQDrawBootstrapGrid – A simple jQuery plugin that draws grid columns to a Bootstrap enabled layout.

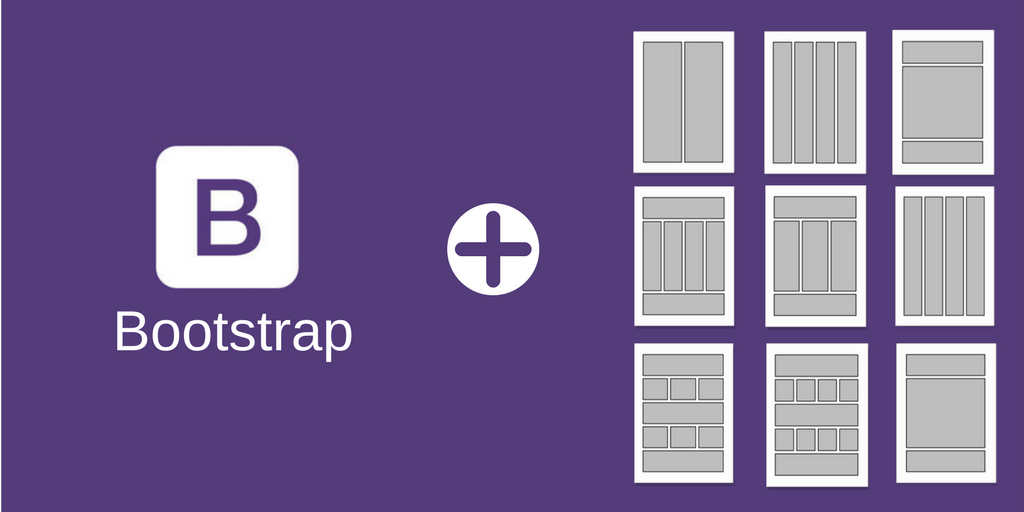


Image : Advance components of Bootstrap

Reference: <https://www.webwash.net/wp-content/uploads/2018/01/d8-bootstrap-layouts-feature.png>

### Magnify

Magnify – A JS plugin for adding a magnifying glass to images on mouseover.



Image : Advance components of Bootstrap

Reference: <https://camo.githubusercontent.com/9dca757e32e3a73c530bae322463bda634071a20d75e4e4e4307181f922cfb57/68747470733a2f2f7261772e6769746875622e636f6d2f6d617263617562652f626f6f7473747261702d6d61676e6966792f6d61737465722f6578616d706c652f73637265656e73686f742e706e67>

### Modal Windows

Bootstrap Modal – This plugin extends Bootstrap’s native modals to provide additional functionality (responsive, stackable, Ajax…).

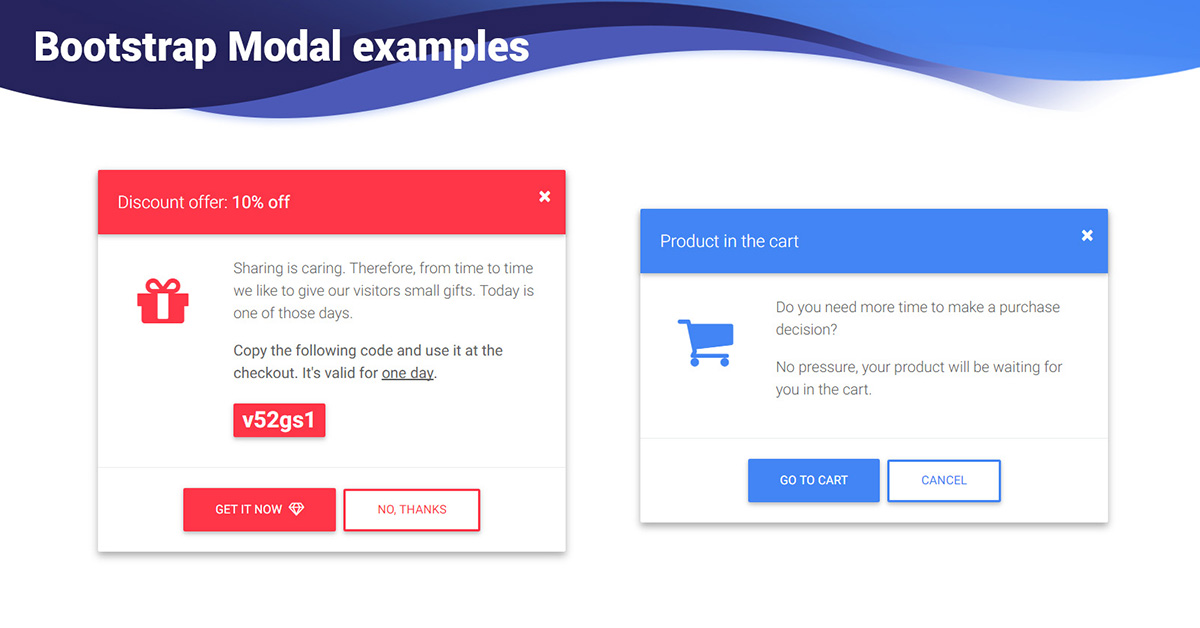


Image : Advance components of Bootstrap

Reference: <https://mdbcdn.b-cdn.net/wp-content/uploads/2018/02/modal-examples.jpg>

Bootstrap Scroll Modal – A modification of the Bootstrap Modal plugin that allows for unlimited modal height with full page scrolling.

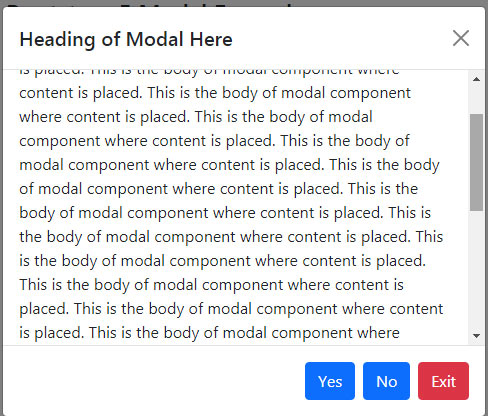


Image : Advance components of Bootstrap

Reference: <https://www.jquery-az.com/wp-content/uploads/2021/06/1-2-modal-sccrollable.jpg>

### Navigation

Contextmenu – A context menu plugin for Bootstrap.



Image : Advance components of Bootstrap

Reference: <https://www.codehim.com/wp-content/uploads/2019/08/context-menu-bootstrap.jpg>

### Pagination

bootpag – A jQuery plugin helps you create dynamic pagination with Bootstrap.

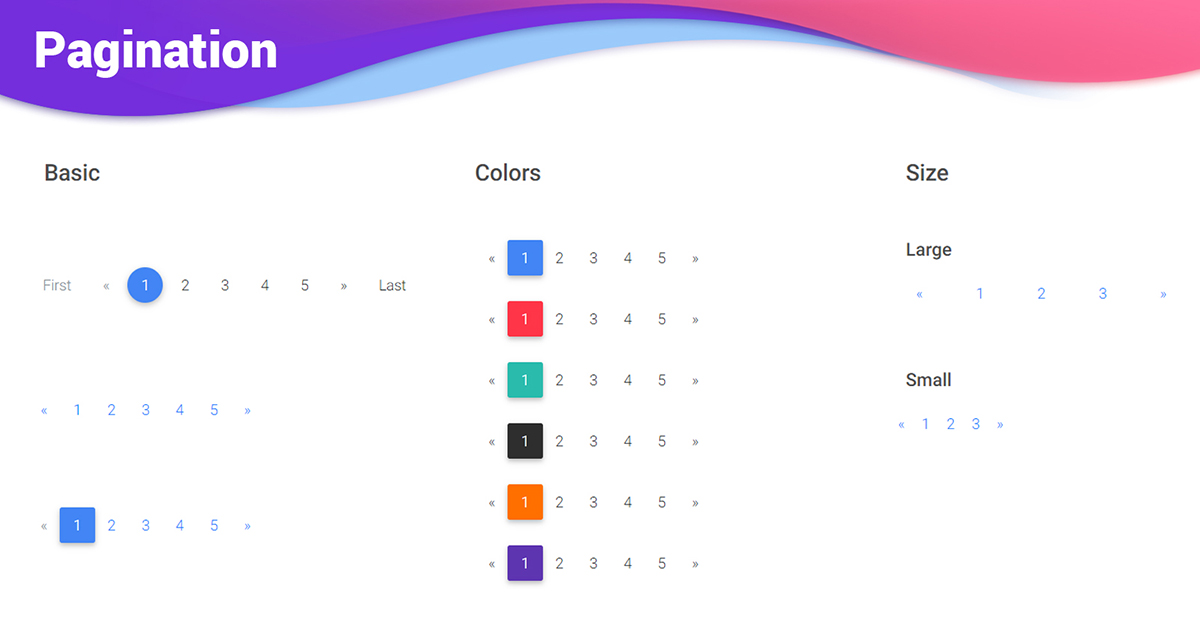


Image : Advance components of Bootstrap

Reference: <https://mdbcdn.b-cdn.net/wp-content/uploads/2016/08/pagination.jpg>

### Progress Bars

Bootstrap Progressbar – A multi-color progress bar component for Bootstrap.

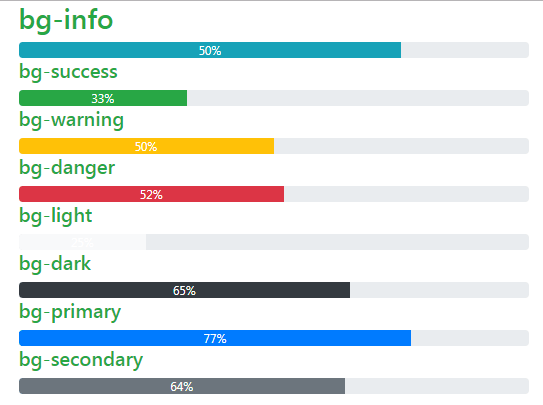


Image : Advance components of Bootstrap

Reference: <https://www.jquery-az.com/wp-content/uploads/2018/01/19-2-Bootstra-4p-progress-colors.png>

### Ratings

Bootstrap Star Rating – A jQuery star rating plugin for Bootstrap that supports fractional star fill and RTL input support.

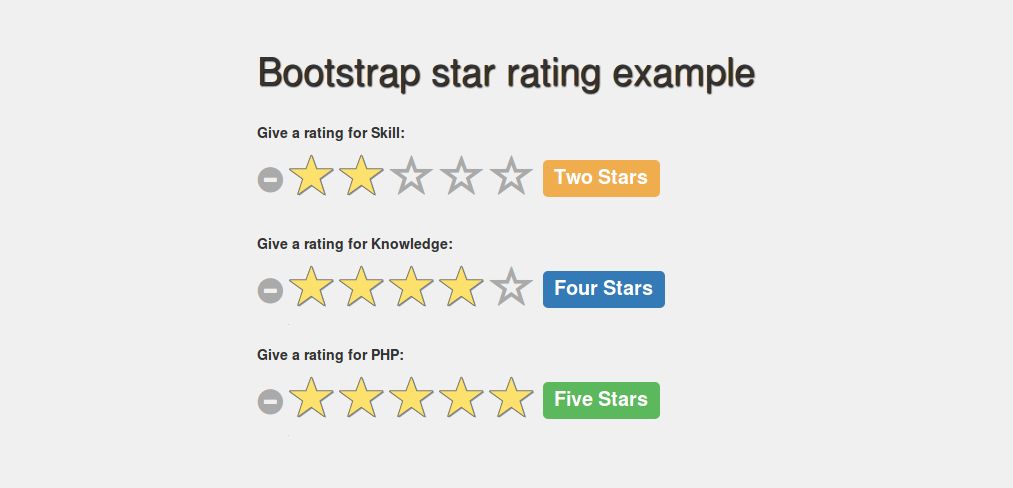


Image : Advance components of Bootstrap

Reference: <https://www.itsolutionstuff.com/upload/bootstrap-rating.png>

### Social Buttons

Social Buttons for Bootstrap – A pure CSS social sign-in button library.

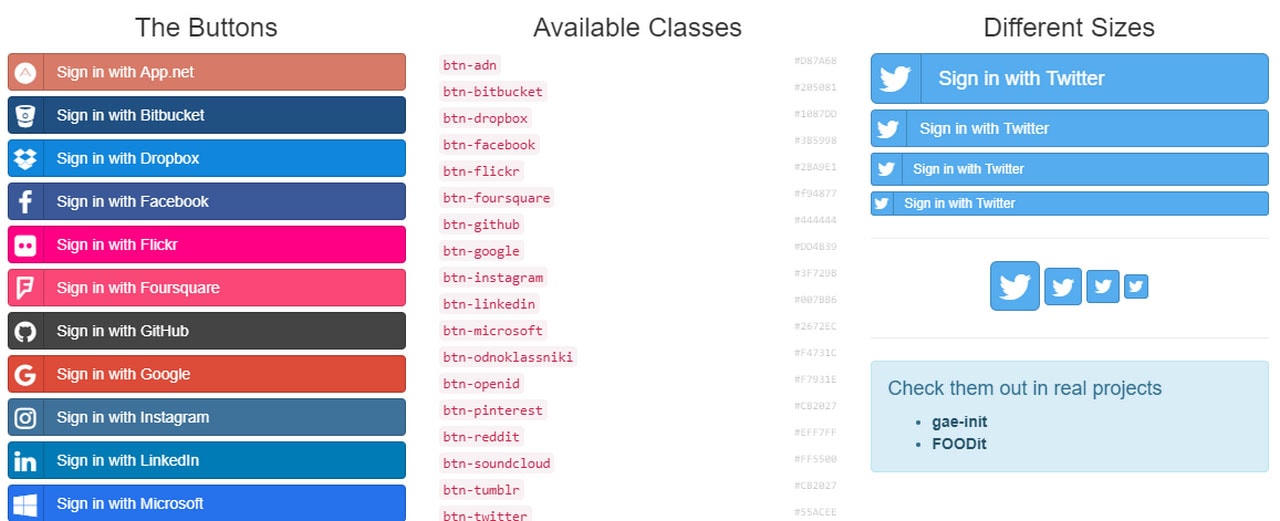


Image : Advance components of Bootstrap

Reference: <https://designmodo.com/wp-content/uploads/2019/08/6-Social-Buttons-for-Bootstrap.jpg>

### Tabs

Tabcordion.js – A simple jQuery plugin that transforms a set of Bootstrap tabs into a Bootstrap accordion.

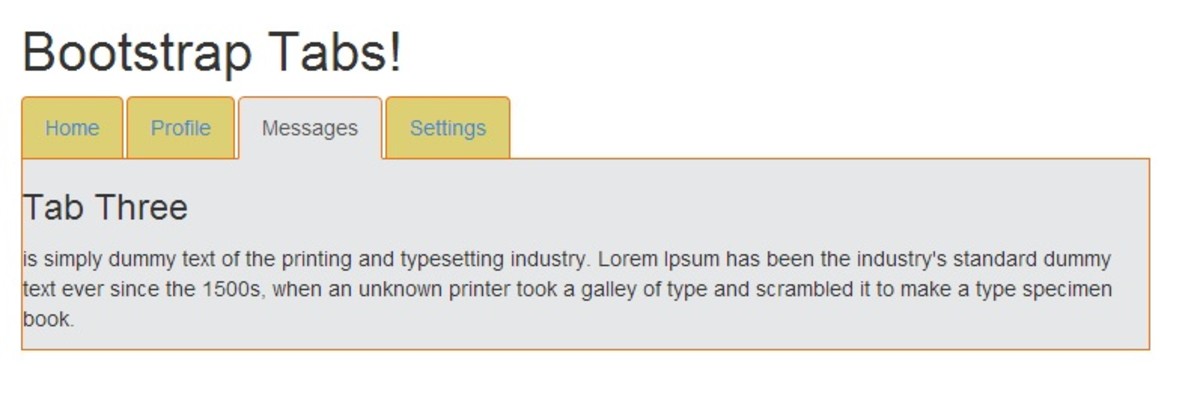


Image : Advance components of Bootstrap

Reference: <https://images.saymedia-content.com/.image/t_share/MTc0Mjk3NDI1Njk4MzY2OTcy/apply-custom-styles-to-bootastrap-tabs-step-by-step.jpg>

## Bootstrap 5 Utilities

### Background

With bootstrap, it’s easy to add some background-color CSS rule in an element to convey a specific connotation using its predefined contextual background color classes which follow its built-in theme colors. These are composed of a subset of color palettes for generating color schemes.

Note: Background color utility classes do not set color or text color. You can use the contextual text color utility classes via .text-\* color utilities if you want to achieve this style according to your constraint (more on this below).

The following are the supported background color utility classes that you can use when adding background-color:

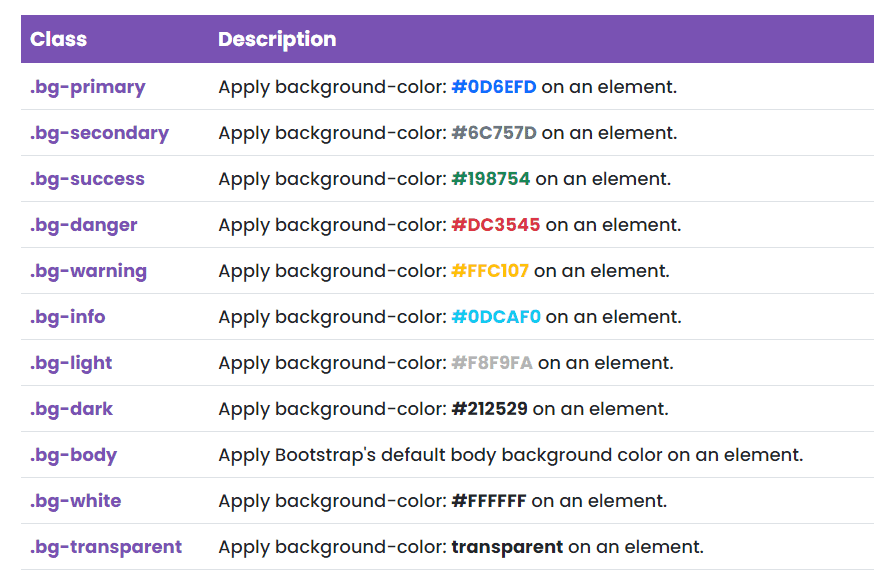


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/1.png>

### Borders

Another CSS style that is regularly used in any layout design is border. The border properties allow you to define the style, width, and color of an element’s border. With bootstrap, you can quickly style the border and border-radius of an element by using the predefined border utility classes.

The following are the supported border utility classes that you can use when adding a border to an element:

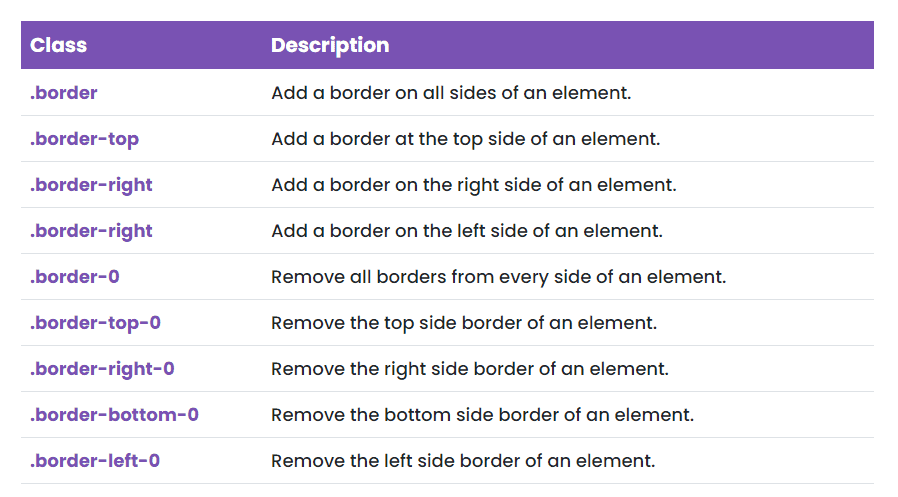


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/4.png>

The following are the border-color utilities built on bootstrap theme colors:

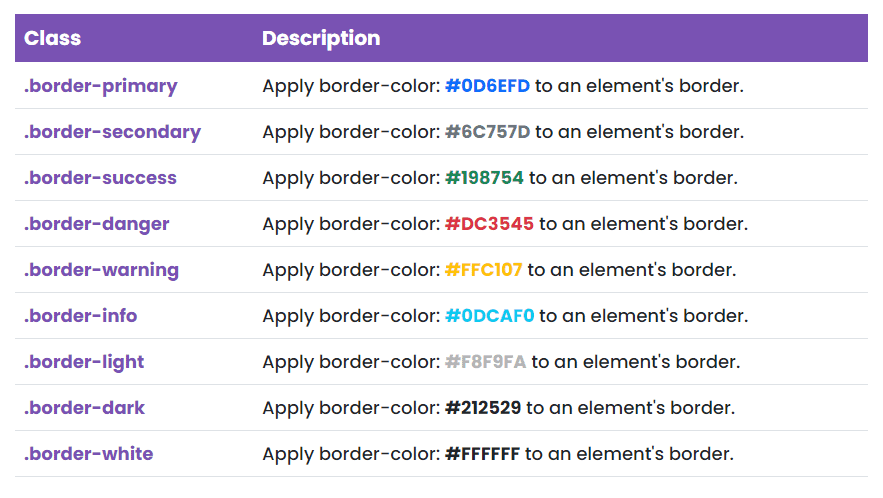


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/5.png>

Additionally, you can also define the border thickness of an element using the border width utility classes:

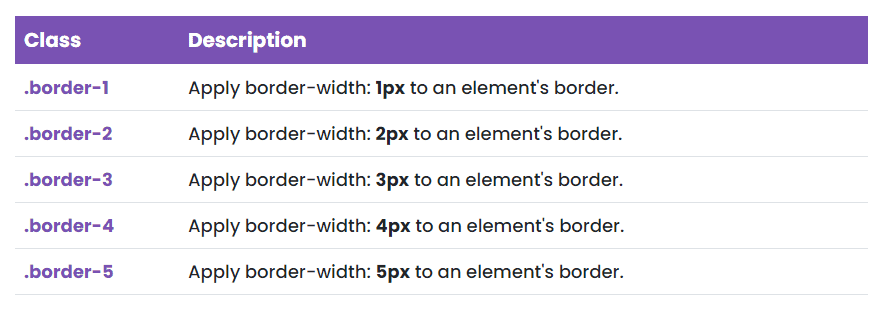


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/6.png>

You can also use the border-radius utility classes to quickly add a radius or rounded corner(s) to an element:

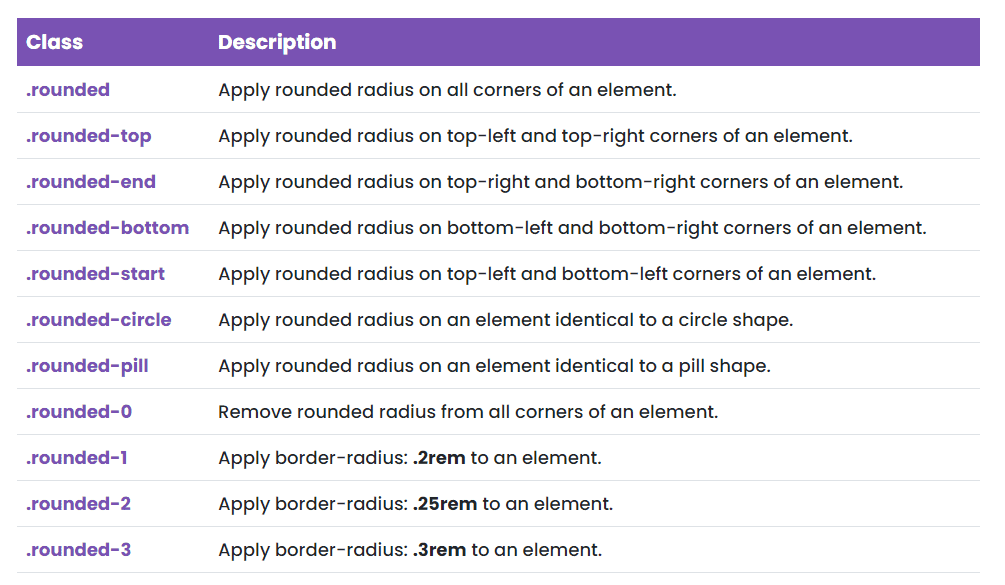


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/7.png>

### Color

You can also apply the same contextual colors that we used for the background and border color to every text element through bootstrap text color utility classes. These are frequently used for conveying meaning for a particular action or situation on your website or app.

The use of these utility classes is very straightforward. To do this, you can simply utilize the following classes and apply these classes directly to any of your text elements:

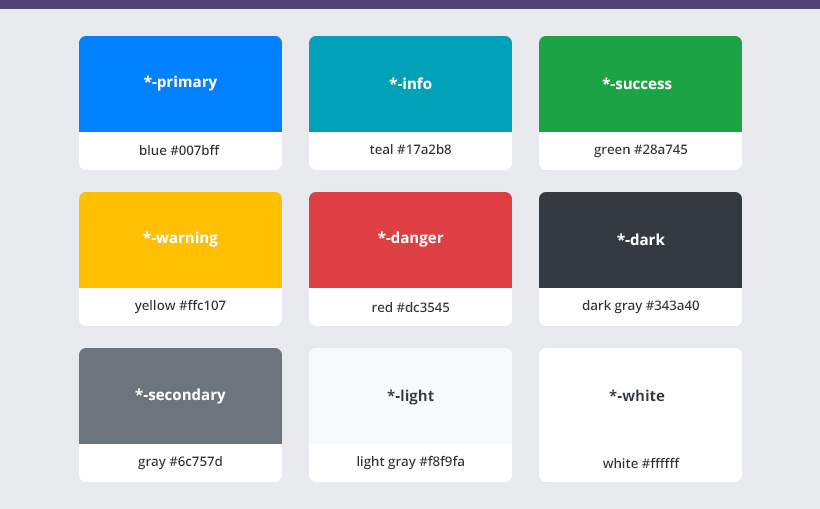


Image : Bootstrap 5 Utilities

Reference: <https://www.bitdegree.org/learn/storage/media/images/7383b588-563f-4117-82bd-4c866b5490bd.jpg>

### Display

Another helpful set of bootstrap utility classes that lets you easily and responsively toggle display value of a specific element in a specific breakpoint or viewport are the display utility classes.

These classes consist of a subset of classes that mostly use breakpoint or viewport abbreviation in them to control the responsive display by showing and hiding elements by screen resolution or device viewport width.

The following are the main supported display utility classes that you can use when dealing with a component’s responsive toggle:

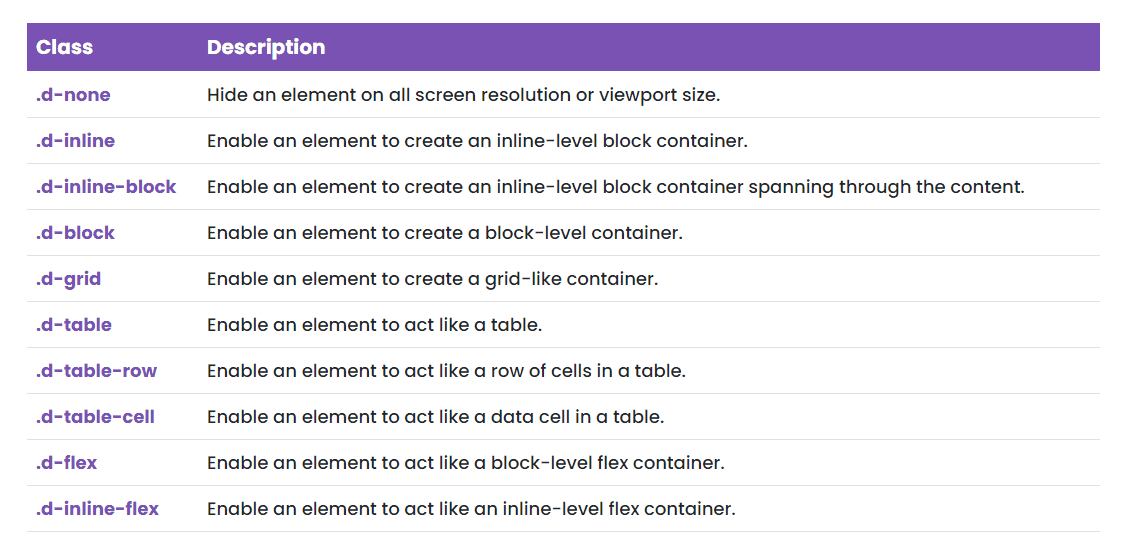


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/10.png>

Additionally, display utility classes can be applied to all breakpoints, from xs to xxl. You can use these utility classes to show or hide elements on your preferred screen resolution or viewport width using the format .d-{breakpoint}-{value} for sm, md, lg, xl, and xxl. For instance, you can use .d-md-none, .d-sm-block, .d-md-flex and the list go on.

Lastly, you can change the display value of an element when printing via print display utility classes. These display in print utility classes support for the same display values as the responsive .d-\* utilities. You can do this simply by adding the print- after .d-\*.

The following are the list of all the display in print utility classes.

* .d-print-none
* .d-print-inline
* .d-print-inline-block
* .d-print-block
* .d-print-grid
* .d-print-table
* .d-print-table-row
* .d-print-table-cell
* .d-print-flex
* .d-print-inline-flex

### Flex

Flexbox offers a better way to organize elements in a web page in a predictable manner. While it sometimes performs like a float, it offers a lot more than that such as reordering elements and avoiding known issues of float.

In version 4 of Bootstrap, flexbox support has finally arrived. Using display properties via display utilities, it’s easy to create a flexbox container and transform direct children elements into flex items.

Bootstrap comes with the following main flexbox utility classes:

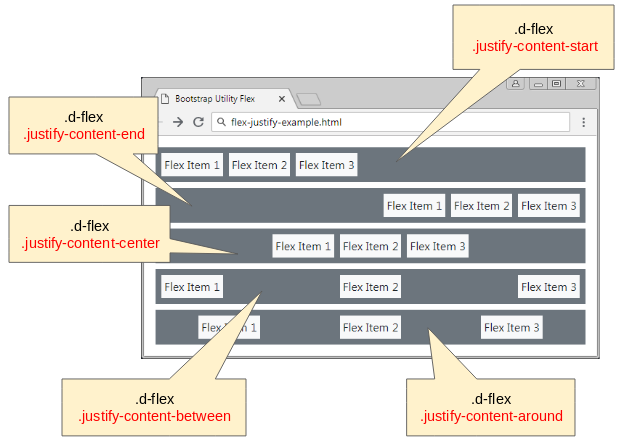


Image : Bootstrap 5 Utilities

Reference: <https://s1.o7planning.com/en/12023/images/22980304.png>

Additionally, responsive variations exist on most of these flexbox utility classes. As an example for .d-flex, you can use classes on specific breakpoints or viewports such as .d-sm-flex or .d-xl-flex.

### Interactions

Bootstrap 5 also provides CSS property controls that allow users to interact with content. This determines whether the user can select text or not and if a specific pointer event is active in a text element. This doesn’t have any effect on content loaded as part of a browser’s user interface except in textboxes.

The following are the supported interactions utility classes:

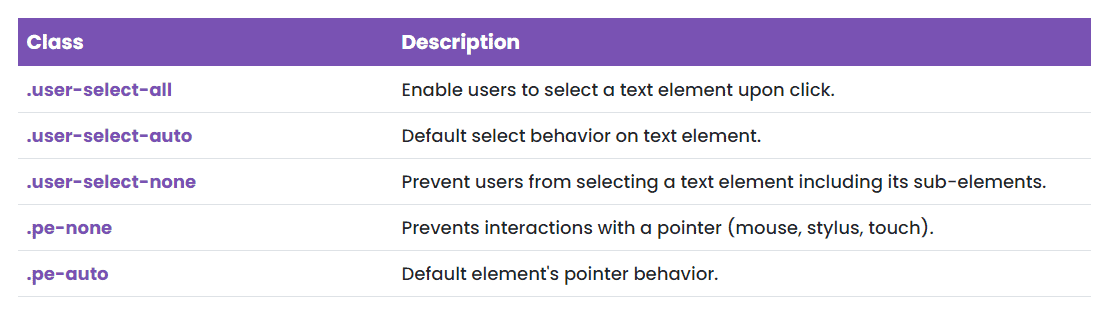


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/14.png>

### Overflow

With bootstrap, it’s also easy to set your preferred behavior for an element’s overflow using the overflow utility classes. For instance, when an element’s content is too big to fit in its container context, you can specify whether to clip content in both directions or add a scrollbar on it.

Bootstrap provides the following overflow utility classes that you can use on your elements right out of the box:

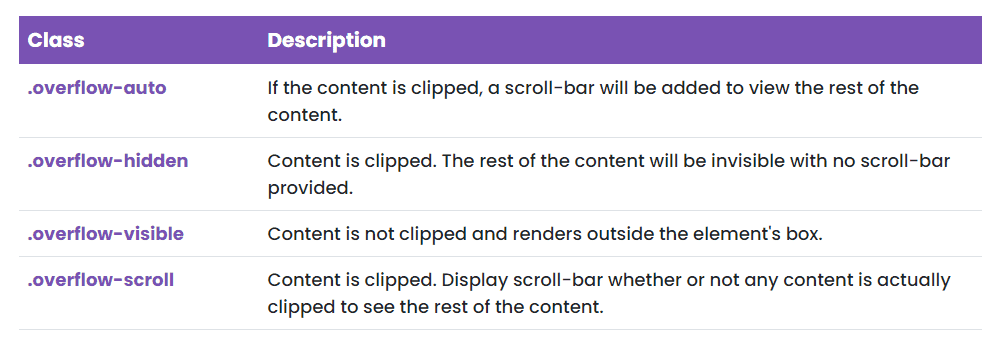


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/16.png>

### Position

Another useful set of bootstrap utility classes are the position utilities. These classes allow you to define the type of positioning method and final location you want an element to behave in a web page.

Bootstrap offers the following position and edge positioning utility classes on the fly:

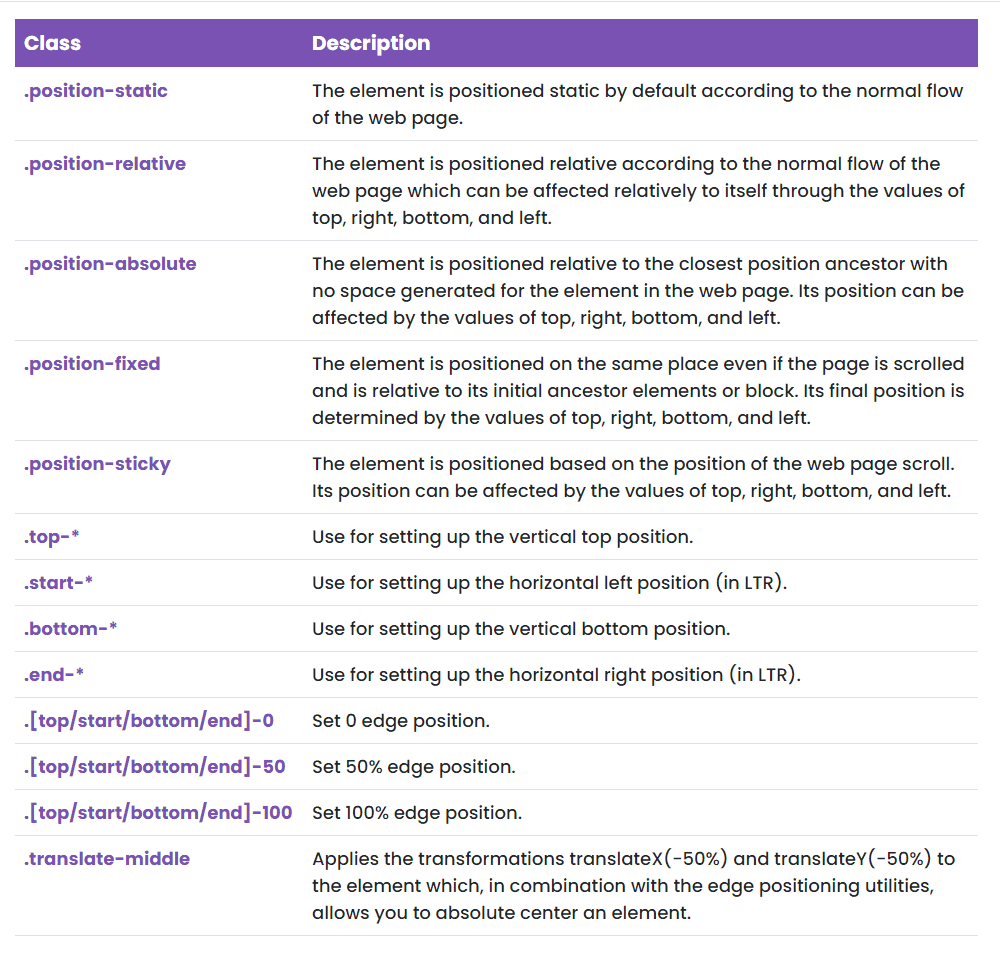


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/18.png>

### Box Shadow

With the box-shadow CSS property, you can cast shadow effects around an element’s frame which is determined by X and Y offsets. Bootstrap 5 also has its own out-of-the-box utility classes to quickly add box shadows to your elements.

The following are the supported box-shadow utility classes:

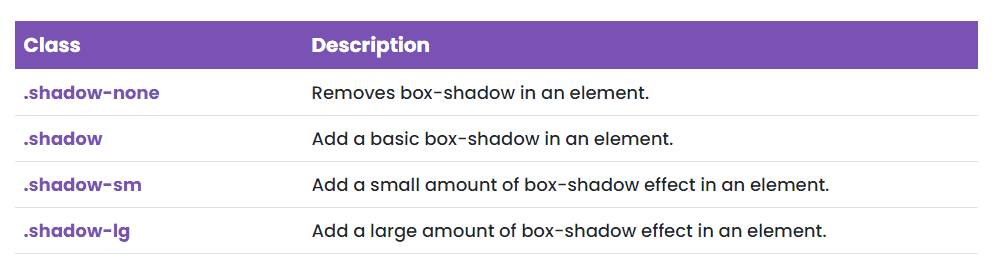


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/20.png>

### Sizing

One of the important factors in web design is the responsive sizes of each element that can span or shrink in size across different screen resolutions or viewport widths.

Aside from the responsive columns and flexbox classes, bootstrap also offers sizing utility classes that can make an element as wide or tall depending on the layout you are trying to achieve.

This became possible through the sizing utility classes provided by bootstrap out of the box.

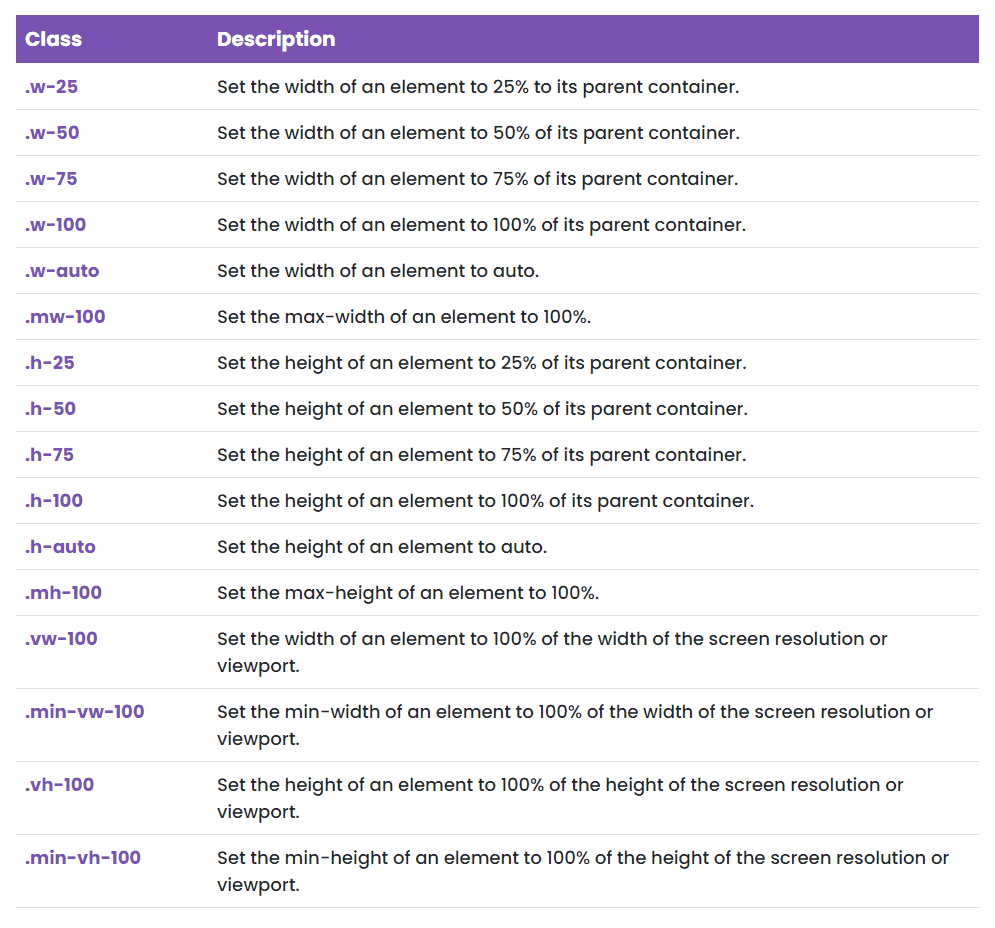


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/22.png>

### Spacing

Bootstrap offers a variety of shorthand responsive margin, padding, and gap utility classes to modify an element position or appearance. These spacing utility classes can be used on any supported breakpoints, from xs to xxl and don’t have any breakpoint abbreviation in them. The standard measurement for each margin, padding and gap classes are ranging from .25rem to 3rem.

These classes are formatted {property}{sides}-{size} for xs and {property}{sides}-{breakpoint}-{size} for sm, md, lg, xl, and xxl. Available sizes starts from 0 up to 5 and an extra auto keyword to set the margin to auto.

The following are supported spacing utility classes:

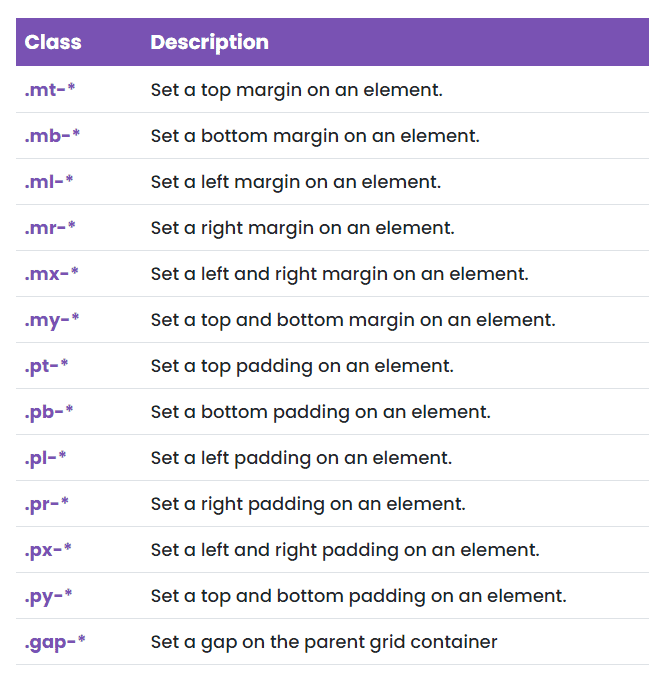


Image : Bootstrap 5 Utilities

Reference: <https://designmodo.com/wp-content/uploads/2021/04/24.png>

### Text

Bootstrap 5 also added a few new extra utility classes for common text or link control. With these classes, you can easily realign text to components, wrap text, modify the font size or weight, transform case and more. Just like the previous set of utility classes that I introduced and demonstrated above, responsive classes are also available on some of these text utility classes which you can use with viewport width breakpoints as the grid system.

The text alignment classes used for aligning text:

* .text-start - It places the text to the left on all types of the viewport.
* .text-end - It places the text to the right on all typ4es of the viewport.
* .text-center - It places the text to the center on all types of the viewport
* .text-sm-start - It places the text to the left on small sizes viewports.
* .text-md-end - It places the text to the right on medium size viewport.

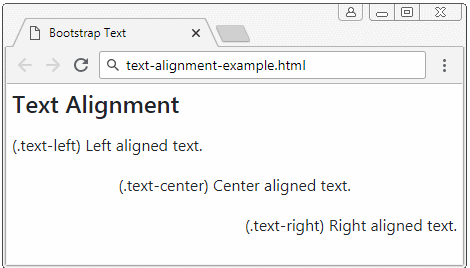


Image : Bootstrap 5 Utilities

Reference: <https://s1.o7planning.com/en/12071/images/24027841.gif>

# Introduction to Jquery

jQuery is an open source JavaScript library that simplifies the interactions between an HTML/CSS document, or more precisely the Document Object Model (DOM), and JavaScript.

Elaborating the terms, jQuery simplifies HTML document traversing and manipulation, browser event handling, DOM animations, Ajax interactions, and cross-browser JavaScript development.

jQuery is widely famous with its philosophy of “Write less, do more.” This philosophy can be further elaborated as three concepts:

Finding some elements (via CSS selectors) and doing something with them (via jQuery methods) i.e. locate a set of elements in the DOM, and then do something with that set of elements.

Chaining multiple jQuery methods on a set of elements

Using the jQuery wrapper and implicit iteration

## What is jQuery?

jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

* HTML/DOM manipulation
* CSS manipulation
* HTML event methods
* Effects and animations
* AJAX
* Utilities

## Basic syntax for any jQuery function is:

|  |
| --- |
| $(selector).action() |

* A $ sign is to define/access jQuery
* A (selector) is to “query (or find)” HTML elements in html page
* A jQuery action() is the action to be performed on the selected element(s)

## Why jQuery?

Some of the key points which support the answer for why to use jQuery:

* It is incredibly popular, which is to say it has a large community of users and a healthy amount of contributors who participate as developers and evangelists.
* It normalizes the differences between web browsers so that you don’t have to.
* It is intentionally a lightweight footprint with a simple yet clever plugin architecture.
* Its repository of plugins is vast and has seen steady growth since jQuery’s release.
* Its API is fully documented, including inline code examples, which in the world of JavaScript libraries is a luxury. Heck, any documentation at all was a luxury for years.
* It is friendly, which is to say it provides helpful ways to avoid conflicts with other JavaScript libraries.

## Advantages:

* Wide range of plug-ins. jQuery allows developers to create plug-ins on top of the JavaScript library.
* Large development community
* It has a good and comprehensive documentation
* It is a lot more easy to use compared to standard javascript and other javascript libraries.
* JQuery lets users develop Ajax templates with ease, Ajax enables a sleeker interface where actions can be performed on pages without requiring the entire page to be reloaded.
* Being Light weight and a powerful chaining capabilities makes jQuery more strong.

## Disadvantages:

* While JQuery has an impressive library in terms of quantity, depending on how much customization you require on your website, the functionality may be limited thus using raw javascript may be inevitable in some cases.
* The JQuery javascript file is required to run JQuery commands, while the size of this file is relatively small (25-100KB depending on the server), it is still a strain on the client computer and maybe your web server as well if you intend to host the JQuery script on your own web server.

## Adding jQuery to Your Web Pages

There are several ways to start using jQuery on your web site. You can:

* Download the jQuery library from jQuery.com
* Include jQuery from a CDN, like Google

### Downloading jQuery

There are two versions of jQuery available for downloading:

* Production version - this is for your live website because it has been minified and compressed
* Development version - this is for testing and development (uncompressed and readable code)

Both versions can be downloaded from jQuery.com.

The jQuery library is a single JavaScript file, and you reference it with the HTML <script> tag (notice that the <script> tag should be inside the <head> section):

|  |
| --- |
| <head>  <script src="jquery-3.5.1.min.js"></script>  </head> |

### jQuery CDN

If you don't want to download and host jQuery yourself, you can include it from a CDN (Content Delivery Network).

Google is an example of someone who host jQuery:

|  |
| --- |
| <head>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  </head> |

## jQuery Syntax

The jQuery syntax is tailor-made for selecting HTML elements and performing some action on the element(s).

Basic syntax is:

|  |
| --- |
| $(selector).action() |

* A $ sign to define/access jQuery
* A (selector) to "query (or find)" HTML elements
* A jQuery action() to be performed on the element(s)

Examples:

1. $(this).hide() - hides the current element.
2. $("p").hide() - hides all <p> elements.
3. $(".test").hide() - hides all elements with class="test".
4. $("#test").hide() - hides the element with id="test".

### The Document Ready Event

You might have noticed that all jQuery methods in our examples, are inside a document ready event:

|  |
| --- |
| $(document).ready(function(){  // jQuery methods go here...  }); |

This is to prevent any jQuery code from running before the document is finished loading (is ready).

It is good practice to wait for the document to be fully loaded and ready before working with it. This also allows you to have your JavaScript code before the body of your document, in the head section.

|  |
| --- |
| $("p") |

Here are some examples of actions that can fail if methods are run before the document is fully loaded:

* Trying to hide an element that is not created yet
* Trying to get the size of an image that is not loaded yet

|  |
| --- |
| $(function(){  // jQuery methods go here...  }); |

Tip: The jQuery team has also created an even shorter method for the document ready event:

## jQuery Selectors

jQuery selectors allow you to select and manipulate HTML element(s).

jQuery selectors are used to "find" (or select) HTML elements based on their name, id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors.

All selectors in jQuery start with the dollar sign and parentheses: $().

### The element Selector

The jQuery element selector selects elements based on the element name.

You can select all <p> elements on a page like this:

Example

When a user clicks on a button, all <p> elements will be hidden:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">  </script>  <script>  $(document).ready(function(){  $("button").click(function(){  $("p").hide();  });  });  </script>  </head>  <body>  <h2>This is a heading</h2>  <p>This is a paragraph.</p>  <button>Click me to hide paragraphs</button>  </body>  </html> |

### The #id Selector

The jQuery #id selector uses the id attribute of an HTML tag to find the specific element.

An id should be unique within a page, so you should use the #id selector when you want to find a single, unique element.

To find an element with a specific id, write a hash character, followed by the id of the HTML element:

|  |
| --- |
| $("#test") |

Example

When a user clicks on a button, the element with id="test" will be hidden:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js">  </script>  <script>  $(document).ready(function(){  $("button").click(function(){  $("#test").hide();  });  });  </script>  </head>  <body>  <h2>This is a heading</h2>  <p>This is a paragraph.</p>  <p id="test">This is another paragraph.</p>  <button>Click me</button>  </body>  </html> |

### The .class Selector

The jQuery .class selector finds elements with a specific class. To find elements with a specific class, write a period character, followed by the name of the class:

|  |
| --- |
| $(".test") |

Example

When a user clicks on a button, the elements with class="test" will be hidden:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  <script>  $(document).ready(function(){  $("button").click(function(){  $(".test").hide();  });  });  </script>  </head>  <body>  <h2 class="test">This is a heading</h2>  <p class="test">This is a paragraph.</p>  <p>This is another paragraph.</p>  <button>Click me</button>  </body>  </html> |

### More Examples of jQuery Selectors

|  |  |
| --- | --- |
| **Syntax** | **Description** |
| $("\*") | Selects all elements |
| $(this) | Selects the current HTML element |
| $("p.intro") | Selects all <p> elements with class="intro" |
| $("p:first") | Selects the first <p> element |
| $("ul li:first") | Selects the first <li> element of the first <ul> |
| $("ul li:first-child") | Selects the first <li> element of every <ul> |
| $("[href]") | Selects all elements with an href attribute |
| $("a[target='\_blank']") | Selects all <a> elements with a target attribute value equal to "\_blank" |
| $("a[target!='\_blank']") | Selects all <a> elements with a target attribute value NOT equal to "\_blank" |
| $(":button") | Selects all <button> elements and <input> elements of type="button" |
| $("tr:even") | Selects all even <tr> elements |
| $("tr:odd") | Selects all odd <tr> elements |

## jQuery Attributes

Some of the most basic components we can manipulate when it comes to DOM elements are the properties and attributes assigned to those elements.

Most of these attributes are available through JavaScript as DOM node properties. Some of the more common properties are −

* className
* tagName
* id
* href
* title
* rel
* src

Consider the following HTML markup for an image element –

|  |
| --- |
| <img id = "imageid" src = "image.gif" alt = "Image" class = "myclass" |

In this element's markup, the tag name is img, and the markup for id, src, alt, class, and title represents the element's attributes, each of which consists of a name and a value.

jQuery gives us the means to easily manipulate an element's attributes and gives us access to the element so that we can also change its properties.

### Get Attribute Value

The attr() method can be used to either fetch the value of an attribute from the first element in the matched set or set attribute values onto all matched elements.

Example

Following is a simple example which fetches title attribute of <em> tag and set <div id = "divid"> value with the same value −

|  |
| --- |
| <html>  <head>  <title>The jQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  var title = $("em").attr("title");  $("#divid").text(title);  });  </script>  </head>    <body>  <div>  <em title = "Bold and Brave">This is first paragraph.</em>  <p id = "myid">This is second paragraph.</p>  <div id = "divid"></div>  </div>  </body>  </html> |

This will produce following result –

|  |
| --- |
| *This is first paragraph.*  This is second paragraph.  Bold and Brave |

### Set Attribute Value

The attr(name, value) method can be used to set the named attribute onto all elements in the wrapped set using the passed value.

Example

Following is a simple example which set src attribute of an image tag to a correct location –

|  |
| --- |
| <html>  <head>  <title>The jQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  $("#myimg").attr("src", "/jquery/images/jquery.jpg");  });  </script>  </head>    <body>  <div>  <img id = "myimg" src = "/images/jquery.jpg" alt = "Sample image" />  </div>  </body>  </html> |

This will produce following result −



### Applying Styles

The addClass( classes ) method can be used to apply defined style sheets onto all the matched elements. You can specify multiple classes separated by space.

Example

Following is a simple example which sets class attribute of a para <p> tag –

|  |
| --- |
| <html>  <head>  <title>The jQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  $("em").addClass("selected");  $("#myid").addClass("highlight");  });  </script>    <style>  .selected { color:red; }  .highlight { background:yellow; }  </style>  </head>    <body>  <em title = "Bold and Brave">This is first paragraph.</em>  <p id = "myid">This is second paragraph.</p>  </body>  </html> |

This will produce following result –

|  |
| --- |
| *This is first paragraph.*  This is second paragraph. |

### Attribute Methods

Following table lists down few useful methods which you can use to manipulate attributes and properties –

|  |  |  |
| --- | --- | --- |
| S.No | Methods | Description |
| 1 | attr(properties) | Set a key/value object as properties to all matched elements. |
| 2 | attr(key, fn) | Set a single property to a computed value, on all matched elements. |
| 3 | removeAttr(name) | Remove an attribute from each of the matched elements. |
| 4 | hasClass(class) | Returns true if the specified class is present on at least one of the set of matched elements. |
| 5 | removeClass(class) | Removes all or the specified class(es) from the set of matched elements. |
| 6 | toggleClass(class) | Adds the specified class if it is not present, removes the specified class if it is present. |
| 7 | html() | Get the html contents (innerHTML) of the first matched element. |
| 8 | html(val) | Set the html contents of every matched element. |
| 9 | text() | Get the combined text contents of all matched elements. |
| 10 | text(val) | Set the text contents of all matched elements. |
| 11 | val() | Get the input value of the first matched element. |
| 12 | val(val) | Set the value attribute of every matched element if it is called on <input> but if it is called on <select> with the passed <option> value then passed option would be selected, if it is called on check box or radio box then all the matching check box and radiobox would be checked. |

## jQuery - DOM Traversing

jQuery is a very powerful tool which provides a variety of DOM traversal methods to help us select elements in a document randomly as well as in sequential method. Most of the DOM Traversal Methods do not modify the jQuery object and they are used to filter out elements from a document based on given conditions.

### Find Elements by Index

Following is a simple example which adds the color to second list item.

|  |
| --- |
| <html>  <head>  <title>The JQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  $("li").eq(2).addClass("selected");  });  </script>    <style>  .selected { color:red; }  </style>  </head>    <body>  <div>  <ul>  <li>list item 1</li>  <li>list item 2</li>  <li>list item 3</li>  <li>list item 4</li>  <li>list item 5</li>  <li>list item 6</li>  </ul>  </div>  </body>  </html> |

This will produce following result –

|  |
| --- |
| * list item 1 * list item 2 * list item 3 * list item 4 * list item 5 * list item 6 |

### Filtering out Elements

The filter( selector ) method can be used to filter out all elements from the set of matched elements that do not match the specified selector(s). The selector can be written using any selector syntax.

Example

Following is a simple example which applies color to the lists associated with middle class –

|  |
| --- |
| <html>  <head>  <title>The JQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  $("li").filter(".middle").addClass("selected");  });  </script>    <style>  .selected { color:red; }  </style>  </head>    <body>  <div>  <ul>  <li class = "top">list item 1</li>  <li class = "top">list item 2</li>  <li class = "middle">list item 3</li>  <li class = "middle">list item 4</li>  <li class = "bottom">list item 5</li>  <li class = "bottom">list item 6</li>  </ul>  </div>  </body>  </html> |

This will produce following result –

|  |
| --- |
| * list item 1 * list item 2 * list item 3 * list item 4 * list item 5 * list item 6 |

### Locating Descendant Elements

The find( selector ) method can be used to locate all the descendant elements of a particular type of elements. The selector can be written using any selector syntax.

Example

Following is an example which selects all the <span> elements available inside different <p> elements –

|  |
| --- |
| <html>  <head>  <title>The JQuery Example</title>  <script type = "text/javascript"  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">  </script>    <script type = "text/javascript" language = "javascript">  $(document).ready(function() {  $("p").find("span").addClass("selected");  });  </script>    <style>  .selected { color:red; }  </style>  </head>    <body>  <p>This is 1st paragraph and <span>THIS IS RED</span></p>  <p>This is 2nd paragraph and <span>THIS IS ALSO RED</span></p>  </body>  </html> |

This will produce following result −

|  |
| --- |
| This is 1st paragraph and THIS IS RED  This is 2nd paragraph and THIS IS ALSO RED |

### JQuery DOM Filter Methods

Following table lists down useful methods which you can use to filter out various elements from a list of DOM elements –

|  |  |  |
| --- | --- | --- |
| S.No | Methods | Description |
| 1 | eq(index) | Reduce the set of matched elements to a single element. |
| 2 | filter(selector) | Removes all elements from the set of matched elements that do not match the specified selector(s). |
| 3 | filter(fn) | Removes all elements from the set of matched elements that do not match the specified function. |
| 4 | is(selector) | Checks the current selection against an expression and returns true, if at least one element of the selection fits the given selector. |
| 5 | map(callback) | Translate a set of elements in the jQuery object into another set of values in a jQuery array (which may, or may not contain elements). |
| 6 | not(selector) | Removes elements matching the specified selector from the set of matched elements. |
| 7 | slice(start,[end]) | Selects a subset of the matched elements. |

### JQuery DOM Traversing Methods

Following table lists down other useful methods which you can use to locate various elements in a DOM –

|  |  |  |
| --- | --- | --- |
| S.No | Methods | Description |
| 1 | add(selector) | Adds more elements, matched by the given selector, to the set of matched elements. |
| 2 | andSelf() | Add the previous selection to the current selection. |
| 3 | children([selector]) | Get a set of elements containing all of the unique immediate children of each of the matched set of elements. |
| 4 | closest(selector) | Get a set of elements containing the closest parent element that matches the specified selector, the starting element included. |
| 5 | contents() | Find all the child nodes inside the matched elements (including text nodes), or the content document, if the element is an iframe. |
| 6 | end() | Revert the most recent 'destructive' operation, changing the set of matched elements to its previous state. |
| 7 | find(selector) | Searches for descendant elements that match the specified selectors. |
| 8 | next([selector]) | Get a set of elements containing the unique next siblings of each of the given set of elements. |
| 9 | nextAll(selector) | Find all sibling elements after the current element. |
| 10 | offsetParent() | Returns a jQuery collection with the positioned parent of the first matched element. |
| 11 | parent([selector]) | Get the direct parent of an element. If called on a set of elements, parent returns a set of their unique direct parent elements. |
| 12 | parents([selector]) | Get a set of elements containing the unique ancestors of the matched set of elements (except for the root element). |
| 13 | prev([selector]) | Get a set of elements containing the unique previous siblings of each of the matched set of elements. |
| 14 | prevAll([selector]) | Find all sibling elements in front of the current element. |
| 15 | siblings([selector]) | Get a set of elements containing all of the unique siblings of each of the matched set of elements. |

## jQuery - CSS Selectors Methods

The jQuery library supports nearly all of the selectors included in Cascading Style Sheet (CSS) specifications 1 through 3, as outlined on the World Wide Web Consortium's site.

Using JQuery library developers can enhance their websites without worrying about browsers and their versions as long as the browsers have JavaScript enabled. Most of the JQuery CSS Methods do not modify the content of the jQuery object and they are used to apply CSS properties on DOM elements.

### Apply CSS Properties

This is very simple to apply any CSS property using JQuery method css( PropertyName, PropertyValue ).

Here is the syntax for the method −

selector.css( PropertyName, PropertyValue );

Here you can pass PropertyName as a javascript string and based on its value, PropertyValue could be string or integer.

### Apply Multiple CSS Properties

You can apply multiple CSS properties using a single JQuery method CSS( {key1:val1, key2:val2....). You can apply as many properties as you like in a single call.

Here is the syntax for the method −

selector.css( {key1:val1, key2:val2....keyN:valN})

Here you can pass key as property and val as its value as described above.

Example

Following is an example which adds font color as well as background color to the second list item.

<html>

<head>

<title>The jQuery Example</title>

<script type = "text/javascript"

src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">

</script>

<script type = "text/javascript" language = "javascript">

$(document).ready(function() {

$("li").eq(2).css({"color":"red", "background-color":"green"});

});

</script>

</head>

<body>

<div>

<ul>

<li>list item 1</li>

<li>list item 2</li>

<li>list item 3</li>

<li>list item 4</li>

<li>list item 5</li>

<li>list item 6</li>

</ul>

</div>

</body>

</html>

This will produce following result-

* list item 1
* list item 2
* list item 3
* list item 4
* list item 5
* list item 6

### Setting Element Width & Height

The width( val ) and height( val ) method can be used to set the width and height respectively of any element.

Example- Following is a simple example which sets the width of first division element where as rest of the elements have width set by style sheet

<html>

<head>

<title>The jQuery Example</title>

<script type = "text/javascript"

src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">

</script>

<script type = "text/javascript" language = "javascript">

$(document).ready(function() {

$("div:first").width(100);

$("div:first").css("background-color", "blue");

});

</script>

<style>

div {

width:70px; height:50px; float:left;

margin:5px; background:red; cursor:pointer;

}

</style>

</head>

<body>

<div></div>

<div>d</div>

<div>d</div>

<div>d</div>

<div>d</div>

</body>

</html>

### JQuery CSS Methods

Following table lists down all the methods which you can use to play with CSS properties −

|  |  |  |
| --- | --- | --- |
| S.No | Methods | Description |
| 1 | css(name) | Return a style property on the first matched element. |
| 2 | css(name, value) | Set a single style property to a value on all matched elements. |
| 3 | css(properties) | Set a key/value object as style properties to all matched elements. |
| 4 | height(val) | Set the CSS height of every matched element. |
| 5 | height() | Get the current computed, pixel, height of the first matched element. |
| 6 | innerHeight() | Gets the inner height (excludes the border and includes the padding) for the first matched element. |
| 7 | innerWidth() | Gets the inner width (excludes the border and includes the padding) for the first matched element. |
| 8 | offset() | Get the current offset of the first matched element, in pixels, relative to the document. |
| 9 | offsetParent() | Returns a jQuery collection with the positioned parent of the first matched element. |
| 10 | outerHeight([margin]) | Gets the outer height (includes the border and padding by default) for the first matched element. |
| 11 | outerWidth([margin]) | Get the outer width (includes the border and padding by default) for the first matched element. |
| 12 | position() | Gets the top and left position of an element relative to its offset parent. |
| 13 | scrollLeft([val]) | When a value is passed in, the scroll left offset is set to that value on all matched elements. |
| 14 | scrollLeft() | Gets the scroll left offset of the first matched element. |
| 15 | scrollTop(val) | When a value is passed in, the scroll top offset is set to that value on all matched elements. |
| 16 | scrollTop() | Gets the scroll top offset of the first matched element. |
| 17 | width(val) | Set the CSS width of every matched element. |
| 18 | width() | Get the current computed, pixel, width of the first matched element. |

## jQuery Effects

jQuery provides a trivially simple interface for doing various kind of amazing effects. jQuery methods allow us to quickly apply commonly used effects with a minimum configuration. This tutorial covers all the important jQuery methods to create visual effects.

### jQuery hide() and show()

With jQuery, you can hide and show HTML elements with the hide() and show() methods:

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#hide").click(function(){

$("p").hide();

});

$("#show").click(function(){

$("p").show();

});

});

</script>

</head>

<body>

<p>If you click on the "Hide" button, I will disappear.</p>

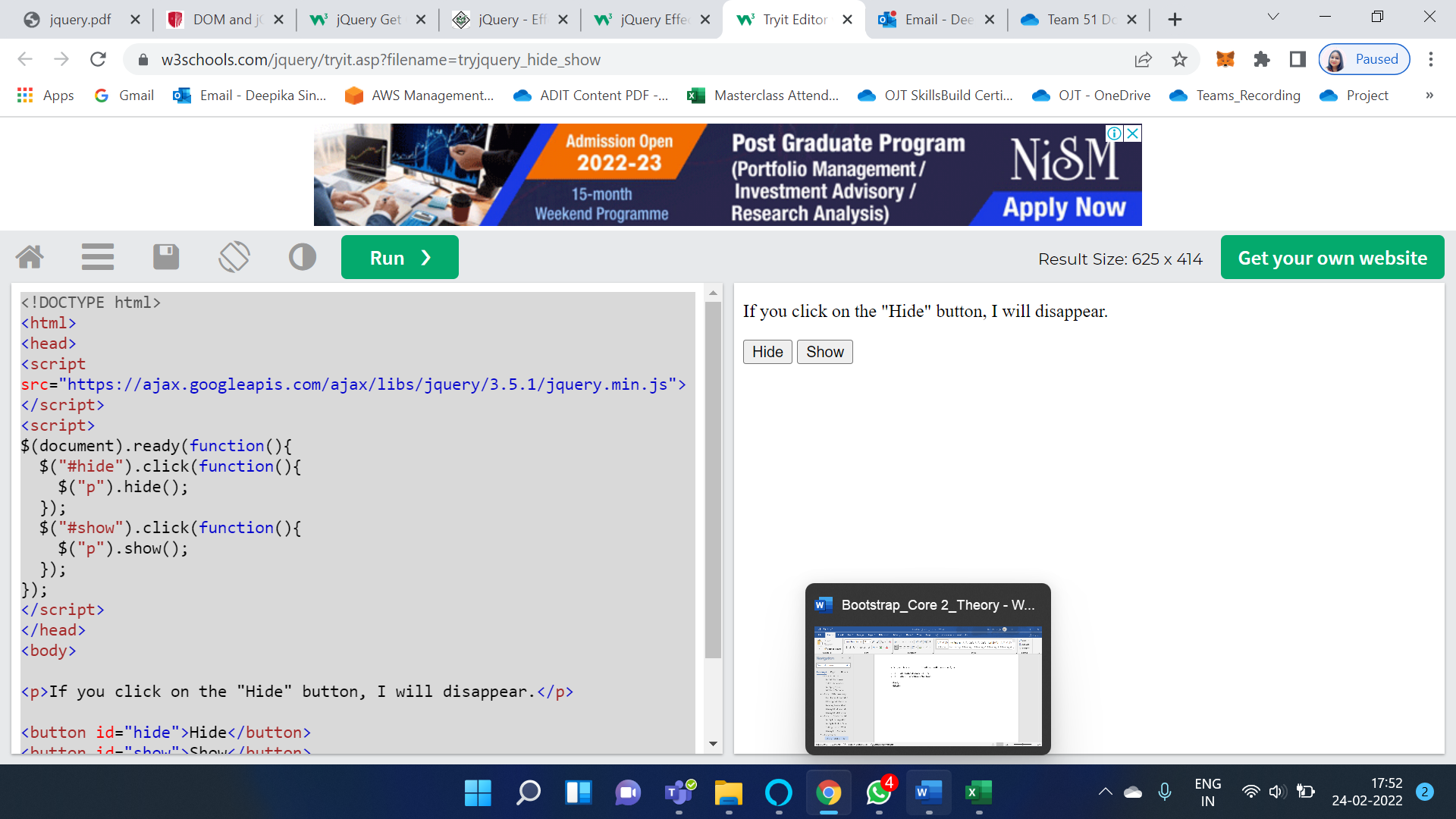
<button id="hide">Hide</button>

<button id="show">Show</button>

</body>

</html>

This will produce following result-



### jQuery toggle()

You can also toggle between hiding and showing an element with the toggle() method.

Shown elements are hidden and hidden elements are shown:

<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("button").click(function(){

$("p").toggle();

});

});

</script>

</head>

<body>

<button>Toggle between hiding and showing the paragraphs</button>

<p>This is a paragraph with little content.</p>

<p>This is another small paragraph.</p>

</body>

</html>

### jQuery Fade

With jQuery you can fade elements in and out of visibility.

#### jQuery Fading Methods

With jQuery you can fade an element in and out of visibility. jQuery has the following fade methods:

* fadeIn()
* fadeOut()
* fadeToggle()
* fadeTo()

**jQuery fadeIn() Method**

The jQuery fadeIn() method is used to fade in a hidden element.

Syntax:

$(selector).fadeIn(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The optional callback parameter is a function to be executed after the fading completes.

**jQuery fadeOut() Method**

The jQuery fadeOut() method is used to fade out a visible element.

Syntax:

$(selector).fadeOut(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The optional callback parameter is a function to be executed after the fading completes.

**jQuery fadeToggle() Method**

The jQuery fadeToggle() method toggles between the fadeIn() and fadeOut() methods.

If the elements are faded out, fadeToggle() will fade them in. If the elements are faded in, fadeToggle() will fade them out.

Syntax:

$(selector).fadeToggle(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The optional callback parameter is a function to be executed after the fading completes.

**jQuery fadeTo() Method**

The jQuery fadeTo() method allows fading to a given opacity (value between 0 and 1).

Syntax:

$(selector).fadeTo(speed,opacity,callback);

The required speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The required opacity parameter in the fadeTo() method specifies fading to a given opacity (value between 0 and 1).

The optional callback parameter is a function to be executed after the function completes.

### jQuery slide

jQuery Sliding Methods- With jQuery you can create a sliding effect on elements.jQuery has the following slide methods:

* slideDown()
* slideUp()
* slideToggle()
* jQuery slideDown() Method

The jQuery slideDown() method is used to slide down an element.

Syntax:

$(selector).slideDown(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The optional callback parameter is a function to be executed after the sliding completes.

**jQuery slideUp() Method**

The jQuery slideUp() method is used to slide up an element.

Syntax:

$(selector).slideUp(speed,callback);

The optional speed parameter specifies the duration of the effect. It can take the following values: "slow", "fast", or milliseconds. The optional callback parameter is a function to be executed after the sliding completes.

**jQuery slideToggle() Method**

The jQuery slideToggle() method toggles between the slideDown() and slideUp() methods.If the elements have been slid down, slideToggle() will slide them up.

If the elements have been slid up, slideToggle() will slide them down.

$(selector).slideToggle(speed,callback);

The optional speed parameter can take the following values: "slow", "fast", milliseconds. The optional callback parameter is a function to be executed after the sliding completes.

## jQuery Animations

The jQuery animate() method performs custom animation using element's style properties. The animate() method changes existing style properties to the specified properties with motion.

Specify a selector to get the reference of an element to which you want to add animation effect and then call animate() method with JSON object for style properties, speed of animation and other options.

$('selector expression').animate({ stylePropertyName : 'value'}, duration, easing, callback);

$('selector expression').animate({ propertyName : 'value'},{ options });

Parameters

|  |  |
| --- | --- |
| Parameters | Details |
| properties | An object of CSS properties and values that the animation will move toward |
| duration | (default: 400) A string or number determining how long the animation will run |
| easing | (default: swing) A string indicating which easing function to use for the  Transition |
| complete | A function to call once the animation is complete, called once per matched  element. |
| start | specifies a function to be executed when the animation begins. |
| step | specifies a function to be executed for each step in the animation. |
| queue | a Boolean value specifying whether or not to place the animation in the  effects queue. |
| progress | specifies a function to be executed after each step in the animation. |
| done | specifies a function to be executed when the animation ends. |
| fail | specifies a function to be executed if the animation fails to complete. |
| specialEasing | a map of one or more CSS properties from the styles parameter, and their  corresponding easing functions. |
| always | specifies a function to be executed if the animation stops without completing. |

### Animation with callback

Sometimes we need to change words position from one place to another or reduce size of the words and change the color of words automatically to improve the attraction of our website or web apps. JQuery helps a lot with this concept using fadeIn(), hide(), slideDown() but its functionality are limited and it only done the specific task which assign to it.

Jquery fix this problem by providing an amazing and flexible method called .animate(). This

method allows to set custom animations which is used css properties that give permission to fly

over borders. for example if we give css style property as width:200; and current position of the

DOM element is 50, animate method reduce current position value from given css value and

animate that element to 150.But we don't need to bother about this part because animation engine

will handle it.

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<script>

$("#btn1").click(function(){

$("#box").animate({width: "200px"});

});

</script>

<button id="btn1">Animate Width</button>

<div id="box" style="background:#98bf21;height:100px;width:100px;margin:6px;"></div>

References

1. <https://getbootstrap.com/>
2. <https://www.w3schools.com/bootstrap/>
3. <https://www.freecodecamp.org/news/learn-bootstrap-4-in-30-minute-by-building-a-landing-page-website-guide-for-beginners-f64e03833f33/>
4. <https://dev.to/bootstrap/ui-components-5a9k>
5. <https://webpixels.io/blog/bootstrap-5-components-new-website-and-open-source-css-design-system>
6. <https://shuffle.dev/blog/2021/06/new-library-to-quickly-create-bootstrap-5-templates/>
7. <https://icons.getbootstrap.com/>
8. <https://www.codegrepper.com/code-examples/whatever/glyphicons+bootstrap+5>
9. <https://www.w3schools.com/jquery/jquery_intro.asp>
10. <https://www.geeksforgeeks.org/jquery-introduction/>
11. <https://www.javatpoint.com/jquery-selectors>
12. <https://www.tutorialrepublic.com/jquery-reference/jquery-selectors.php>