

1 Bootstrap Introduction

- Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first web sites.
- Bootstrap is a front-end framework for faster and easier web development
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs
- It is Mobile first approach.
- Bootstrap is free to download and use.

Advantages of Bootstrap

- Save lots of time — You can save lots of time and efforts using the Bootstrap predefined design templates and classes and concentrate on other development work.
- Responsive features — Using Bootstrap you can easily create responsive websites that appear more appropriately on different devices and screen resolutions without any change in markup.
- Consistent design — All Bootstrap components share the same design templates and styles through a central library, so the design and layout of your web pages will be consistent.
- Easy to use — Bootstrap is very easy to use. Anybody with the basic working knowledge of HTML, CSS and JavaScript can start development with Bootstrap.
- Compatible with browsers — Bootstrap is created with modern web browsers in mind and it is compatible with all modern browsers such as Chrome, Firefox, Safari, Internet Explorer, etc.
- Open Source — And the best part is, it is completely free to download and use.
- Platform Independence: It creates Platform-independent web-pages
- It works with any editor.
- It works with any server side technology.

What is Responsive Web Design or Layout?

- Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.
- Responsive web design is a process of designing and building websites to provide better accessibility and optimal viewing experience to the user.
- Responsive layouts automatically adjust and adapts to any device screen size, whether it is a desktop, a laptop, a tablet, or a mobile phone.



Setting bootstrap reference:

To set reference to bootstrap folder, use *href* attribute of `<link>` element.

Example

<link rel="stylesheet" href="bootstrap.min.css"> link to bootstrap folder.

2 Bootstrap Grid Structure

- Bootstrap includes a responsive, mobile first fluid grid system. It scales up to 12 columns as the device or viewport size increases.

The columns will re-arrange automatically depending on the output device. The figure 1 shows grid structure.

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Figure : Bootstrap grid structure

Grid Classes

- The Bootstrap grid system support following classes:

Class-prefix	Applicable Device	Screen size
col-xs-*	phones	<768px
col-sm-*	small screen devices like tablets	≥768px
col-md-*	medium size devices like desktops	≥992px
col-lg-*	larger device	≥1200px

Working of Bootstrap Grid System

- Rows must be placed within a .container (fixed-width) or .container-fluid (full-width) for proper alignment and padding
- Use rows to create horizontal groups of columns.
- Content should be placed within the columns
- Grid columns are created by specifying the number of twelve available columns you wish to span. For example, three equal columns would use three .col-xs-4.

Examples:

How to get a three equal-width columns starting starting at tablets and scaling to large desktops. On mobile phones, the columns will automatically stack:

Three Equal Columns

.col-sm-4	.col-sm-4	.col-sm-4
-----------	-----------	-----------

```
<div class="row">
  <div class="col-sm-4">.col-sm-4</div>
  <div class="col-sm-4">.col-sm-4</div>
  <div class="col-sm-4">.col-sm-4</div>
</div>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Bootstrap Example</title>
```

```
  <link rel="stylesheet" href="bootstrap_folder/bootstrap.min.css">
```

```
</head>
```

```
<body>
```

```
<div">
```

```
  <h1>Hello World!</h1>
```

```
  <p>Resize the browser window to see the effect.</p>
```

```
  <div class="row">
```

```
    <div class="col-sm-4" style="background-color:lavender;">.col-sm-4</div>
```

```
    <div class="col-sm-4" style="background-color:lavenderblush;">.col-sm-4</div>
```

```
    <div class="col-sm-4" style="background-color:lavender;">.col-sm-4</div>
```

```
  </div>
```

```
</div>
```

```
</body>
```

```
</html>
```

Three column

(1) equal column

```
<!--Row with three equal columns-->
```

```
<div class="row">
```

```
  <div class="col-md-4">!--Column left--</div>
```

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

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

```
</div>
```

2) <!--Row with three columns divided in 1:4:1 ratio-->

```
<div class="row">
```

```
  <div class="col-md-2">--Column left--</div>
```

```
  <div class="col-md-8">--Column middle--</div>
```

```
  <div class="col-md-2">--Column right--</div>
```

```
</div>
```

3) <!--Row with three columns divided unevenly-->

```
<div class="row">
```

```
  <div class="col-md-3">--Column left--</div>
```

```
  <div class="col-md-7">--Column middle--</div>
```

```
  <div class="col-md-2">--Column right--</div>
```

```
</div>
```

Two column
<pre> <div class="row"> <div class="col-sm-4">.col-sm-4</div> <div class="col-sm-8">.col-sm-8</div> </div> </pre>
Two equal column
<pre> <div class="container"> <!--Row with two equal columns--> <div class="row"> <div class="col-sm-6">--Column left--</div> <div class="col-sm-6">--Column right--</div> </div> <!--Row with two columns divided in 1:2 ratio--> <div class="row"> <div class="col-sm-4">--Column left--</div> <div class="col-sm-8">--Column right--</div> </div> <!--Row with two columns divided in 1:3 ratio--> <div class="row"> <div class="col-sm-3">--Column left--</div> <div class="col-sm-9">--Column right--</div> </div> </div> </pre>
Two unequal column
<pre> <div class="row"> <div class="col-sm-4">.col-sm-4</div> <div class="col-sm-8">.col-sm-8</div> </div> </pre>

3.1 Bootstrap - Tables

Bootstrap provides classes for tables to improve the appearance of table. A basic Bootstrap table has effect of light padding and contains horizontal separators.

{{

Supported Table elements:

Tag	Description
<table>	Wrapping element for displaying data in a tabular format
<thead>	Container element for table header rows (<tr>) to label table columns.
<tbody>	Container element for table rows (<tr>) in the body of the table.
<tr>	Container element for a set of table cells (<td> or <th>) that appears on a single row.

<td>	Default table cell.
<th>	Special table cell for column (or row, depending on scope and placement) labels. Must be used within a <thead>
<caption>	Description or summary of what the table holds.

}}

Bootstrap <table> Classes

Bootstrap provides classes to make table responsive.

Use the classes below to style any table:

Class	Description
.table	Adds basic styling (light padding and only horizontal dividers) to any <table>
.table-striped	Adds zebra-striping to any table row within <tbody>. You can create table with alternate background like zebra-stripes
.table-bordered	Adds border on all sides of the table and cells
.table-hover	Enables a hover state on table rows within a <tbody>
.table-condensed	Makes table more compact by cutting cell padding in half and save the space
.table-responsive	make the table scroll horizontally up to small devices

<tr>, <th> and <td> Classes

Use the classes below to color table rows or cells:

Class	Description
.active	Applies the hover color to a particular row or cell
.success	Indicates a successful or positive action
.info	Indicates a neutral informative change or action
.warning	Indicates a warning that might need attention
.danger	Indicates a dangerous or potentially negative action

Example:

Creating a Simple Table with Bootstrap

A basic Bootstrap table has a light padding and only horizontal dividers.

The `.table` class adds basic styling to a table:

```
<table class = "table">
  <caption>Basic Table Layout</caption>

  <thead>
    <tr>
      <th>Name</th>
      <th>City</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td>jmp</td>
      <td>vadodara</td>
    </tr>

    <tr>
      <td>abc</td>
      <td>surat</td>
    </tr>
  </tbody>
</table>

</body>
</html>
```

output

Basic Table Layout	
Name	City
jmp	vadodara
abc	surat

Example of contextual class

Change first line in previous example

`<table class = "table table-striped">` instead of `<table class = "table">`

Apply bootstrap on table row /column

```
<table class = "table">
  <caption>Contextual Table Layout</caption>

  <thead>
    <tr>
```

```

        <th>Product</th>
        <th>Payment Date</th>

    </tr>
</thead>

<tbody>
    <tr class = "active">
        <td>Product1</td>

        <td>Pending</td>
    </tr>

    <tr class = "success">
        <td>Product2</td>

        <td>Delivered</td>
    </tr>

    <tr class = "warning">
        <td>Product3</td>

        <td>In Call to confirm</td>
    </tr>

    <tr class = "danger">
        <td>Product4</td>

        <td>Declined</td>
    </tr>
</tbody>
</table>

```

Example: Responsive table

To make table responsive, in above example write `<table class = "table table responsive">`

3.2 Bootstrap colour

- Bootstrap colour classes are used to apply colour to element depending on their context.
- Bootstrap provides following text colour classes: text-primary, text-secondary, text-success, text-danger, text-warning, text-info, text-light, text-dark, text-muted, text-white.
- Bootstrap provides following background colour classes bg-primary, bg-success, bg-warning, bg-info, bg-danger, bg-secondary, bg-dark and bg-light.

Example:

Following line display light text on danger background colour.

`<p class="bg-danger text-light "> light text on danger background color </p>`

3.3 Bootstrap Alerts

- Bootstrap provides an easy way to create predefined alert messages
- support following class.

alert-success	show a successful or positive activity.
alert-info	show a informative change or action.
alert-warning	show a warning that might require attention.
alert-danger	show a dangerous or potentially negative activity.
alert-primary	show an important activity
alert-secondary	show a less important activity
alert-light	displays dark grey alert box
alert-dark	displays Light grey alert box.

Example:

The following example demonstrates various Bootstrap Alerts classes.

```
<body>
  <div class="alert alert-warning" >
    This is a warning alert
  </div>

  <div class="alert alert-primary">
    This is a primary alert
  </div>

  <div class="alert alert-success" >
    This is a success alert
  </div>

  <div class="alert alert-danger" >
    This is a danger alert
  </div>

  <div class="alert alert-secondary" >
    This is a secondary alert
  </div>

  <div class="alert alert-dark" >
    This is a dark alert
  </div>

  <div class="alert alert-info" >
    This is a info alert
  </div>

  <div class="alert alert-light" >
    This is a light alert
  </div>
</body>
```

Output:

This is a warning alert

This is a primary alert

This is a success alert

This is a danger alert

This is a secondary alert

This is a dark alert

This is a info alert

This is a light alert

Closing Alerts box:

- The alert box does not close automatically.
- To close the alert message, add a .alert-dismissible class to the alert container. Then add class="close" and data-dismiss="alert" to a link or a button element.

Example:

The following example creates closing alert box. when you click close button, this the alert box will disappear/ closed. "×" indicate x sign.

```
<div class="alert alert-dismissible">  
  <button type="button" class="close" data-dismiss="alert">  
    &times;  
  </button>  
  Click on the "x" symbol to close alert box  
</div>
```

Output: Alert box will look like this:

Click on the "x" symbol to close alert box



Animated alert

- It is possible to give animation when closing the alert message. The *fade* and *in* classes are used to add a fading effect.
- Following line create animated alerts.

```
<div class="alert alert-dismissible fade in">
```

3.4 Bootstrap Form controls

- Bootstrap form controls are used to customize form. It allows to give different styles of form controls like labels, buttons, input, select, checkbox, textarea, etc.
- With Bootstrap, some global styling automatically applied to form controls.

Rules for using bootstrap form:

1. Wrap form controls in `<div class="form-group">`
2. Add class `form-control` to `<input>`, `<select>`, and `<textarea>` elements

List of bootstrap form class.

Classes	Description
form-group	group form controls
form-control	make element width is 100%
form-horizontal	create horizontal form
form-inline	create inline form
form-control-file	for File inputs
form-check, form-check- inline	for Checkboxes and radios
input-lg	set height bigger than default-size
input-sm	set height smaller than default-size

Bootstrap provides three different types of form layouts:

1. Vertical Form (default)
2. Horizontal Form
3. Inline Form

1. Vertical or Basic Form

This is the default Bootstrap form layout in which styles are applied to form controls without adding any base class to the `<form>` element.

Example of Form in vertical layout.

```
<body>
<div>
  <h2>Vertical (basic) form</h2>
  <form>
  <div>
```

```

        <label for="email">Email:</label>
        <input type="email" class="form-control" id="email" placeholder="Enter email">
    </div>
    <div class="form-group">
        <label for="pwd">Password:</label>
        <input type="password" class="form-control" id="pwd" placeholder="Enter
password">
    </div>

    <button type="submit" class="btn btn-default">Submit</button>
</form>
</div>
</body>

```

Output:

(ii)

Horizontal Form Layout

In horizontal form layout labels are right aligned and floated to left to make them appear on the same line as form controls.

To create a form that uses the horizontal layout, do the following –

- Add a class of *.form-horizontal* to the parent `<form>` element.
- Wrap labels and controls in a `<div>` with class *.form-group*.
- Add a class of *.control-label* to the labels.

Example:

Following code display Form in Horizontal layout.

```

<body>
    <form class="form-horizontal">
        <div class="form-group">
            <label class="control-label col-sm-2"> User id:</label>
            <div class="col-sm-10">
                <input type="text" class="form-control">
            </div>
        </div>
    </form>

```

```

<div class="form-group">
  <label class="control-label col-sm-2"> Password:</label>
  <div class="col-sm-10">
    <input type="password" class="form-control">
  </div>
</div>
<div class="form-group">
  <div class="col-sm-offset-2 col-sm-10">
  </div>
</div>
<div class="form-group">
  <div class="col-sm-offset-2 col-sm-10">
    <button type="submit" class="btn btn-default"> Submit</button>
  </div>
</div>
</form>
</body>

```

Output:

User id:

Password:

3. Inline Form Layout

- In an inline form layout, all elements are inline and left-aligned. All elements display in single line.
- It create compact layout by placing the form controls side-by-side.
- It will be stacked automatically.
- It is created using Bootstrap class *form-inline* with the <form> element.

Example:

Following code display Inline Form layout.

```

<body>
  <form class="form-inline">
    <div class="form-group">
      <label>User Id:</label>
      <input type="text" class="form-control">
    </div>
    <div class="form-group">
      <label>Password:</label>
      <input type="password" class="form-control">
    </div>
  </form>
</body>

```

output

User Id: Password:

4 Working with buttons

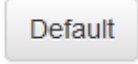
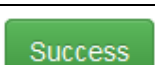
4.1 Buttons

Buttons are the integral part of a website and application. They are used for various purposes like, submit or reset an HTML form, performing interactive actions such as showing or hiding something on a web page on click of the button, etc. The Bootstrap buttons classes apply different styles to buttons.

Button Styles

{{it is applied normally to the <a>, <input>, and <button> elements for the best rendering.}}

Bootstrap support following classes to apply styles to button:

Button	Class	Description
	btn btn-default	Default gray button with gradient.
	btn btn-primary	Provides extra visual weight to indicate primary action button in a set of buttons.
	btn btn-info	Can be used as an alternative to the default button.
	btn btn-success	Indicates a successful or positive action.
	btn btn-warning	Indicates caution should be taken with this action.
	btn btn-danger	Indicates a dangerous or potentially negative action.
	btn btn-link	Deemphasize a button by making it look like a link while maintaining button behavior.

<code><button type="button" class="btn btn-default">Default</button></code>
<code><button type="button" class="btn btn-primary">Primary</button></code>
<code><button type="button" class="btn btn-info">Info</button></code>
<code><button type="button" class="btn btn-success">Success</button></code>
<code><button type="button" class="btn btn-warning">Warning</button></code>
<code><button type="button" class="btn btn-danger">Danger</button></code>
<code><button type="button" class="btn btn-link">Link</button></code>

Following line apply success style to button:

```
<button type="button" class="btn btn-success"> Success </button>
```

Button Sizes (Changing the Sizes of Buttons)

Bootstrap gives you option to scaling a button up or down. You can make buttons larger or smaller through adding an extra class .btn-lg, .btn-sm, or .btn-xs.

Class	Description
.btn-lg	This makes the button size large.
.btn-sm	This makes the button size small.
.btn-xs	This makes the button size extra small.
btn-md	This makes the button size medium.
.btn-block	This creates block level buttons—those that span the full width of a parent.

Example: Following line display button in various size

```
<button type="button" class="btn btn-lg">large</button>
<button type="button" class="btn btn-xs">extra small</button>
<button type="button" class="btn btn-sm">small</button>
<button type="button" class="btn btn-block">block</button>
<button type="button" class="btn btn-md">medium</button>
```

Button State

- A button can be set to an active (appear pressed) or a disabled (unclickable) state:
 - Active Primary
 - Disabled Primary

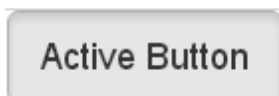
(i) Active State

Buttons will appear pressed (with a darker background, darker border, and inset shadow) when active.

Classes used to make button elements and anchor elements active –

Element	Class
Button element	Use .active class to show that it is activated.
Anchor element	Use .active class to <a> buttons to show that it is activated.

```
<button type="button" class="btn active">active button</button>
```



ii)Disabled State

- Sometimes we need to disable a button for certain reasons like, a user in case is not eligible to perform this particular action, or we want to ensure that user should performed all other required actions before proceed to this particular action.
- It is unclickable
- When you disable a button, it will fade in color by 50%, and lose the gradient.
- classes used to make button element and anchor element disabled –

Element	Class
Button element	Add the disabled <i>attribute</i> to <button> buttons.
Anchor element	Add the disabled <i>class</i> to <a> buttons.

<button type="button" class="btn disabled">disable button</button>



4.2 Button Group

- Group a series of buttons together on a single line in a button group
- Bootstrap provides following button groups class:

Class	Description
btn-group	It is used to group buttons.
btn-group-lg, btn-group-xs, btn-group-sm,	apply button sizing classes to each button in group.
btn-group-vertical	create a vertical button group. i.e. buttons appear vertically stacked instead of horizontally.
btn-toolbar	combine sets of button groups into button toolbars

Example:

example of class *btn-group*

```
<body>
<div class = "btn-group">

    <button type = "button" class = "btn btn-default"> sem 1 </button>
    <button type = "button" class = "btn btn-default"> sem 2 </button>
```

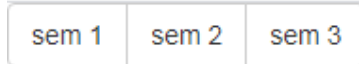
```

        <button type = "button" class = "btn btn-default"> sem 3 </button>
    </div>
</body>

```

Output:

Button will display as shown in figure:



= = == = = == = =

5 Bootstrap: media object

Bootstrap provides an easy way to align media objects (like images or videos) to the left or to the right of some content. This can be used to display blog comments, tweets and so on

5.1 Images

- Using the Bootstrap built-in classes you can easily style images such as making the round cornered or circular images, or give them effect like thumbnails.
- Image can be displayed in circle, thumbnail or Rounded Corners.
- Bootstrap provides three classes that can be used to apply some simple styles to images:

class	description
img-circle	display image in circle.
img-rounded	display image in Rounded corners.
img-thumbnail	gives effect of Thumbnail. It adds padding and a gray border. It is also helpful in creating images or videos gallery.

Example:

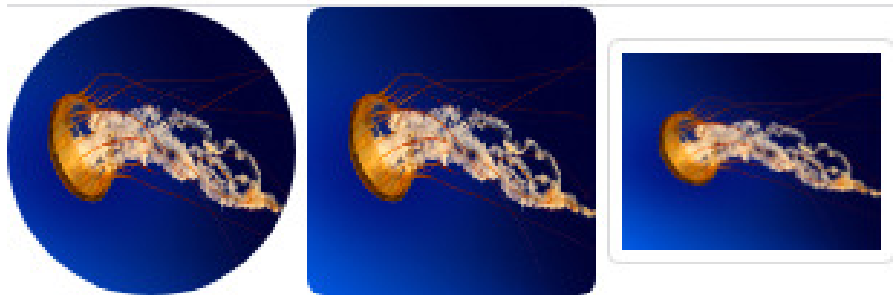
```





```

Output:



Circle

Rounded corner

Thumbnail

Responsive Images

- in Bootstrap you can make the images responsive too. Images come in all sizes. Responsive images automatically adjust to fit the size of the screen.
- .Use *img-responsive* class to make image responsive.

Example:

```

```

5.2 Media Objects

- Bootstrap gives simple way to align media objects such as images or videos, so media objects appear to the left (left-aligned) or to the right (right-aligned) of some content.
- This type of alignment is necessary while writing blog comments, tweets and so on. The media object is used to make the code for creating this type of alignment shorter.

Media objects supported classes:

class	description
media	Define media group
media-body	Define body for media object
media-object	Define media object
media-heading	Set heading
media-left	Set media object left align to text
media-right	Set media object right align to text
media-top	Set media object align to top
media-bottom	Set media object align to bottom
media-middle	Set media object align to middle
Media-list	Define media list

Steps for creating media objects left align:

- 1: Add *media* class to the parent *<div>* element.
- 2: Use child *<div>* inside *media <div>*. Use the *media-left* class to align the media object.
- 3: Place Text inside *<div>* with *class="media-body"*.
- 4: Inside this *<div>*, place headline using *media-heading* class. This is optional.
5. Optionally, You can use add margin and padding.

Example: following code demonstrate how media object left

```
<div class="container">
  <!-- Left-aligned -->
  <div class="media">
    <div class="media-left">
      
    </div>
    <div class="media-body">
      <h3 class="media-heading"> demo fo Media object:left align</h3>
      <p> Concept of web technology html css javascript, Concept of web
technology html css javascript</p>
    </div>
  </div>
</div>
```



demo fo Media object:left align

Concept of web technology html css javascript, Concept of web technology html css javascript

Example:

Following code show how media object left & right align.

```
<body>
  <div class="container">
    <!-- Left-aligned -->
    <div class="media">
      <div class="media-left">
        
      </div>
      <div class="media-body">
        <h3 class="media-heading"> demo of Media object:left align</h3>
        <p> This is demo of media object left alignment. This is demo of media
object left alignment</p>
      </div>
    </div>
    <hr>
    <!-- Right-aligned -->
    <div class="media">
      <div class="media-body">
        <h3 class="media-heading"> Media object: right align</h3>
        <p> This is demo of media object right alignment. This is demo of media
object right alignment</p>
      </div>
      <div class="media-right">
        
      </div>
    </div>
  </div>
</body>
```

Output:



demo fo Media object:left align

This is demo of media object left alignment. This is demo of media object left alignment

Media object: right align

This is demo of media object right alignment. This is demo of media object right alignment



Nesting of media object:

Media objects can also be nested inside other media object. It can be very useful for creating comment threads in a blog post

Example:

Following example shows nesting of media object.

```
<body>
  <h2>Nested Media Objects</h2>

  <div class="media">
    <div class="media-left">
      
    </div>
    <div class="media-body">
      <h4 class="media-heading">XYZ: Posted on 3/6/21</h4>
      <p>How are you? Learning is great!!!</p>

      <!-- Nested media object -->
      <div class="media">
        <div class="media-left">
          
        </div>
        <div class="media-body">
          <h4 class="media-heading">ABC replied on 3/7/21</h4>
          <p>I am fine. Yes learning is great experience</p>

        </div>

      </div>

    </div>
  </div>
</body>
```

Output:

Nested Media Objects



XYZ: Posted on 3/6/21

How are you? Learning is great!!!



ABC replied on 3/7/21>

I am fine. Yes learning is great experience

6. Pagination

- Pagination is the links given to multiple pages within website or application.
- If you have a web site with lots of pages, you may wish to add some sort of pagination to each page. Bootstrap pagination looks like this:



- Bootstrap provides following pagination classes.

Class	Description
pagination	adds the pagination on page.
disabled	make link disabled. A disabled link cannot be click.
active	indicate the current or active page.
pagination-lg	makes pagination block larger size.
pagination-sm	makes pagination block smaller size.

- Pagination is built with list HTML elements.
- create a basic pagination, add the *.pagination* class
- *disabled* and *active* classes are used to customize link or show status of link. *pagination-lg* and *pagination-sm* are used for sizing.

Example

various pagination classes example:

```
<p> pagination normal size </p>
<ul class="pagination">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
</ul>

<p> pagination smaller size: </p>
<ul class="pagination pagination-sm">
  <li class="active"> <a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
</ul>

<p> pagination larger size: </p>
<ul class="pagination pagination-lg">
  <li><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
```

```
<li><a href="#">4</a></li>
```

```
</ul>
```

Output:

pagination normal size

1	2	3
---	---	---

pagination smaller size:

1	2	3
---	---	---

pagination larger size:

1	2	3
---	---	---