

Bootstrap JavaScript Components: Objectives and Outcomes

In this module, we will be learning about Bootstrap's JavaScript components. These components have two parts, the CSS classes, and the JavaScript support. In this module we will learn about using the JS components without writing any JavaScript Code. This is possible using the data-* attributes that Bootstrap provides for us to control the components. At the end of this module, you will be able to:

- Understand the various Bootstrap components that require JavaScript support in order to function
- Use the data-* attributes that Bootstrap's JS API provides for you to control the JS components without writing a single line of JS code

Bootstrap JavaScript Components: Additional Resources

PDFs of Presentations

1-Bootstrap JS.pdfPDF File

Bootstrap Resources

- [Bootstrap and JavaScript](#)
- [Bootstrap JS Data Attributes](#)
- [Bootstrap Programmatic API](#)

Tabs and Tabbed Navigation: Objectives and Outcomes

In this lesson, we examine tabs and tabbed navigation. Tabs require Javascript support to be enabled for navigating the content. At the end of this lesson you will be able to:

- Design a tabbed navigation for your content
- Use tab panes to organize the content and navigate the content using tabbed navigation

Exercise (Instructions): Tabs

Objectives and Outcomes

In this exercise we will explore Bootstrap tabs and tabbed navigation. In particular we will learn about the use of tabs for organizing the content. At the end of this exercise you will be able to:

- Design a web page to use tabbed navigation for organizing the content
- Use tab panes and organize the content into the panes
- Facilitate navigation among the tab panes using the tabbed navigation elements

Adding Tab Navigation Elements

- Open the *aboutus.html* page and move to the second content row containing the details of the corporate leadership of the restaurant.
- Right after the Corporate Leadership heading, introduce the following code to set up the tabbed navigation:

```
<ul class="nav nav-tabs">
  <li class="nav-item">
    <a class="nav-link active" href="#peter"
      role="tab" data-toggle="tab">Peter Pan, CEO</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#danny" role="tab"
      data-toggle="tab">Danny Witherspoon, CFO</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#agumbe" role="tab"
      data-toggle="tab">Agumbe Tang, CTO</a>
  </li>
  <li class="nav-item">
    <a class="nav-link" href="#alberto" role="tab"
      data-toggle="tab">Alberto Somayya, Exec. Chef</a>
  </li>
</ul>
```

Note the use of the `` tag with the *nav* and *nav-tabs* classes to set up the tab navigation. Each list item within the list acts as the tab element. Within each list item, note that we set up the `<a>` tags with the *href* pointing to the *id* of the tab pane of content to be introduced later. Also note that the `<a>` tag contains the *data-toggle=tab* attribute. The first list element's `<a>` tag contains the class *active*. This tab will be the open tab when we view the web page. We can switch to the other tabs using the tabbed navigation that we just set up.

Adding Tab Content

- The details about the various corporate leaders should now be organized into various tab panes. To begin this, we will enclose the entire content into a div element with the class `tab-content` as specified below:

```
<div class="tab-content">
```

```
...
```

```
</div>
```

- Then we take the name and description of the CEO of the company and enclose it within a tab-pane as follows:

```
<div role="tabpanel" class="tab-pane fade show active" id="peter">
```

```
<h3>Peter Pan <small>Chief Epicurious Officer</small></h3>
```

```
<p> ... </p>
```

```
</div>
```

Note the use of the *tab-pane*, *fade*, *show*, and *active* classes and with *peter* as the id. This is the same id used as the *href* in the `<a>` link in the navigation.

- The remaining content is also similarly enclosed inside appropriate divs with the correct ids and the classes specified as above. Only the first tab pane will have the *show* and *active* classes specified to indicate that the content should be visible on the web page by default.

Modifying the tab-content CSS

- We now modify the CSS styles for the `tab-content` class in the *mystyles.css* file as follows:

```
.tab-content {  
    border-left: 1px solid #ddd;  
    border-right: 1px solid #ddd;  
    border-bottom: 1px solid #ddd;  
    padding: 10px;  
}
```

This modification adds a 1px border to the tab content which joins with the upper border introduced by the tab navigation element to give a clean tab like appearance.

- Finally do a Git commit with the message "Tabs".

Conclusions

In this exercise we learnt the use of tabbed navigation, tab content and tab panes and their use in organizing and navigating within the content in a page.

Tabs and Tabbed Navigation: Additional Resources

PDFs of Presentations

2-Tabs-Pills-Navigation.pdf PDF File

Bootstrap Resources

- [Bootstrap Navs](#)
- [Bootstrap Tabs](#)
- [Bootstrap Pills](#)
- [Bootstrap Tabs Javascript Behavior](#)

Hide and Seek: Objectives and Outcomes

In this lesson we learn about the collapse javascript plugin that allows us to hide and reveal content. In particular we explore its use in creating an accordion. At the end of this lesson, you will be able to:

- Use the collapse plugin to hide/reveal content
- Construct the accordion using cards

Exercise (Instructions): Accordion

Objectives and Outcomes

In this exercise we explore the use of the collapse Javascript plugin together with card component to create an accordion to show/hide content in a web page. At the end of this exercise, you will be able to:

- Design an accordion using the collapse plugin together with the card component.

Converting Tabs to Accordion

- First delete the `` class that was introduced for the tabbed navigation.
- Then turn the *tab-content* div into a *accordion* div. Use the code structure as shown below:

```
<div id="accordion">
```

```
...
```

```
</div>
```

- Then, convert the first tab-pane into a card such that the name appears as a card heading, and the <p> will be in the card body. Use the structure of the code as shown below:

```
<div class="card">
  <div class="card-header" role="tab" id="peterhead">
    <h3 class="mb-0">
      <a data-toggle="collapse" data-target="#peter">
        Peter Pan <small>Chief Epicurious Officer</small>
      </a>
    </h3>
  </div>
  <div class="collapse show" id="peter" data-parent="#accordion">
    <div class="card-body">
      <p class="d-none d-sm-block">...</p>
    </div>
  </div>
</div>
```

- For the remaining three leaders, use the same structure as above, with the appropriate ids set up for the cards, as shown in the code structure below:

```
<div class="card">
  <div class="card-header" role="tab" id="dannyhead">
    <h3 class="mb-0">
      <a class="collapsed" data-toggle="collapse" data-target="#danny">
        Dhanasekaran Witherspoon <small>Chief Food Officer</small>
      </a>
    </h3>
  </div>
  <div class="collapse" id="danny" data-parent="#accordion">
    <div class="card-body">
      <p class="d-none d-sm-block">...</em></p>
    </div>
  </div>
```

</div>

</div>

<div class="card">

<div class="card-header" role="tab" id="agumbehead">

<h3 class="mb-0">

Agumbe Tang <small>Chief Taste Officer</small>

</h3>

</div>

<div class="collapse" id="agumbe" data-parent="#accordion">

<div class="card-body">

<p class="d-none d-sm-block">...</p>

</div>

</div>

</div>

<div class="card">

<div class="card-header" role="tab" id="albertohead">

<h3 class="mb-0">

Alberto Somayya <small>Executive Chef</small>

</h3>

</div>

<div class="collapse" id="alberto" data-parent="#accordion">

<div class="card-body">

<p class="d-none d-sm-block">...</p>

</div>

</div>

</div>

- After completing the update, check the behavior of the accordion on the web page.
- Finally do a Git commit with the message "Accordion".

Conclusions

In this exercise we constructed the accordion using the collapse plugin together with the card component.

Hide and Seek: Additional Resources

PDFs of Presentations

3-Collapse.pdfPDF File

Bootstrap Resources

- [Bootstrap Collapse](#)
- [Bootstrap Accordion Example](#)

Revealing Content: Objectives and Outcomes

In this lesson we look at several ways of presenting information to users by overlaying the information on top of the page. In particular, we look at tooltips, popovers and modals. At the end of this lesson, you will be able to:

- Set up a tooltip to be displayed when the user hovers over an area of the page
- Enable popovers when the user clicks on a link or button
- Reveal and hide modals when the user clicks on a link or button

Exercise (Instructions): Tooltips and Modals

Objectives and Outcomes

In this exercise we will examine how to add tooltips to a web page. In addition we look at adding modals to a web page. At the end of this exercise, you will be able to:

- Add tooltips to a web page
- Add modals that are revealed when the user clicks on a link or a button in the web page.

Adding a Tooltip

- Let us now switch to the *index.html* page. We will now add a tooltip to this page. The tooltip will be added to the "Reserve Table" button that is in the jumbotron. We will update the `<a>` tag for the button as follows:

```
<a role="button" class="btn btn-block nav-link btn-warning"
```

```

        data-toggle="tooltip" data-html="true" title="Or Call us at <br><strong>+852
12345678</strong>"

        data-placement="bottom" href="#reserveform">Reserve Table</a>

```

As you can see from the code, we add a *data-toggle*, *data-placement* and a *title* attribute to the `<a>` tag in order to introduce a tooltip.

- The tooltip needs to be activated by adding a small Javascript code to the bottom of the page as follows:

```

<script>

$(document).ready(function(){

    $('[data-toggle="tooltip"]').tooltip();

});

</script>

```

This script is added right after the line that imports the `bootstrap.min.js` file.

Adding a Modal

- In the next step we introduce the modal to the web page. To set up the modal, add the following code right after the navbar at the top of the page:

```

<div id="loginModal" class="modal fade" role="dialog">

    <div class="modal-dialog modal-lg" role="content">

        <!-- Modal content-->

        <div class="modal-content">

            <div class="modal-header">

                <h4 class="modal-title">Login </h4>

                <button type="button" class="close" data-dismiss="modal">&times;</button>

            </div>

            <div class="modal-body">

                <form>

                    <div class="form-row">

                        <div class="form-group col-sm-4">

                            <label class="sr-only" for="exampleInputEmail3">Email
address</label>

```



```

        <input type="email" class="form-control form-control-sm mr-1"
id="exampleInputEmail3" placeholder="Enter email">
    </div>
    <div class="form-group col-sm-4">
        <label class="sr-only" for="exampleInputPassword3">Password</label>
        <input type="password" class="form-control form-control-sm mr-1"
id="exampleInputPassword3" placeholder="Password">
    </div>
    <div class="col-sm-auto">
        <div class="form-check">
            <input class="form-check-input" type="checkbox">
            <label class="form-check-label"> Remember me
        </label>
        </div>
    </div>
</div>
<div class="form-row">
    <button type="button" class="btn btn-secondary btn-sm ml-auto" data-
dismiss="modal">Cancel</button>
    <button type="submit" class="btn btn-primary btn-sm ml-1">Sign
in</button>
</div>
</form>
</div>
</div>
</div>

```

- Next we introduce another link on the right side of the navbar in order to trigger the display of the modal. To do this, add the following code in the navbar after the :

```

<span class="navbar-text">
    <a data-toggle="modal" data-target="#loginModal">

```

```
<span class="fa fa-sign-in"></span> Login</a>
</span>
```

We are introducing another link to the right of the navbar using the *navbar-text*. This contains a link with an `<a>` tag with the `data-toggle="modal"` and `data-target="#loginModal"` attributes.

- Save all the changes and do a Git commit with the message "Tooltip and Modal".

Conclusions

In this exercise we explored tooltips and modals as two ways of revealing content for the user upon clicking on a button or a link.

Revealing Content: Additional Resources

PDFs of Presentations

4-Tooltips-Popovers-Modals.pdfPDF File

Bootstrap Resources

- [Bootstrap Tooltips](#)
- [Bootstrap Popovers](#)
- [Bootstrap Modals](#)

Carousel: Objectives and Outcomes

In this lesson we will examine the use of the Carousel component in our web page. We will examine the configuration of the various aspects of the carousel. At the end of this lesson you will be able to:

- Use a carousel component in your web page
- Configure various aspects of the carousel
- Add controls to the carousel to manually control it

Exercise (Instructions): Carousel

Objectives and Outcomes

In this exercise we will examine the carousel component and add it to the web page. We will examine the configuration of the carousel and adding controls to the carousel. At the end of this exercise you will be able to:

- Use a carousel component in your web page
- Configure various aspects of the carousel
- Add controls to the carousel to manually control it

Adding a row for the carousel

- The carousel will be added to the *index.html* page. In this page, go to the top of the container div that contains the content of the page and add a new content row and an inner div spanning all the 12 columns as follows:

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

```
  <div class="col">
```

```
  </div>
```

```
</div>
```

Adding a Carousel

- Next, add the basic carousel div inside the content row that you just added as follows:

```
<div id="mycarousel" class="carousel slide" data-ride="carousel">
```

```
</div>
```

Adding Carousel Content

- Next add the content inside the carousel as follows:

```
<div class="carousel-inner" role="listbox">
```

```
  <div class="carousel-item active">
```

```
    
```

```
    <div class="carousel-caption d-none d-md-block">
```

```
      <h2>Uthappizza <span class="badge badge-danger">HOT</span> <span  
class="badge badge-pill badge-default">$4.99</span></h2>
```

```
      ...
```

```
    </div>
```

```
  </div>
```

```
<div class="carousel-item">
```

```
  ...
```

```
</div>
```

```
<div class="carousel-item">

    ...

</div>

</div>
```

Note that the first item has been set up partially. Fill in the remaining parts from the content rows below.

Adding CSS Classes

- Add the following CSS classes to the *styles.css* file:

```
.carousel {
    background:#512DA8;
}
```

```
.carousel-item {
    height: 300px;
}
```

```
.carousel-item img {
    position: absolute;
    top: 0;
    left: 0;
    min-height: 300px;
}
```

Adding Carousel Controls

- Next, we will add manual controls to the carousel so that we can manually move among the slides. Add the following code to the bottom after the carousel items in the div of the carousel to add slide indicators that enable us to select a specific slide:

```
<ol class="carousel-indicators">

    <li data-target="#mycarousel" data-slide-to="0" class="active"></li>

    <li data-target="#mycarousel" data-slide-to="1"></li>

    <li data-target="#mycarousel" data-slide-to="2"></li>
```


- Then, add the left and right controls to the carousel that enable us to move to the previous and next slide manually. Add this to the bottom of the carousel div:

```
<a class="carousel-control-prev" href="#mycarousel" role="button" data-  
slide="prev">
```

```
<span class="carousel-control-prev-icon"></span>
```

```
</a>
```

```
<a class="carousel-control-next" href="#mycarousel" role="button" data-  
slide="next">
```

```
<span class="carousel-control-next-icon"></span>
```

```
</a>
```

- Do a Git commit with the message "Carousel".

Conclusions

In this exercise we learnt about the carousel component and how to add it to a web page. We also learnt about introducing manual controls to the carousel.

Carousel: Additional Resources

PDFs of Presentations

5-Carousel.pdfPDF File

Bootstrap Resources

- [Bootstrap Carousel](#)

Bootstrap Resources

- [Bootstrap Buttons Checkbox/Radio](#)
- [Bootstrap Modals](#)
- [Bootstrap Forms](#)

In this assignment you will remove the tooltip from the Reserve table button. Then you will move the table reservation form into a modal that will be shown when the Reserve Table button is clicked. The updated reservation form will include a new radio button group allowing you to select the smoking/non-smoking section of the restaurant.

Objectives and Outcomes

In this assignment, you will continue to work with the website that you have been developing in the exercises. You will edit the home page (index.html). You will start with the current home page at the

end of the last exercise in this module. At the end of this assignment, you should have completed the following tasks:

- Moved the table reservation form from the last content row into a modal.
- Included a radio button group in the table reservation form to enable diners to ask for a table in the smoking/non-smoking section of the restaurant.
- Removed the tooltip from the Reserve Table button.
- Updated the Reserve Table button to show the modal containing the table reservation form when the button is clicked.

Assignment Requirements

This assignment requires you to complete the following four tasks. Detailed instructions for each task are given below. The picture of the completed web page included below indicates the location within the web page that will be updated by the four tasks.

Task 1

In this task you will move the table reservation form from the last content row into a modal. You should also remove the last content row.

- The form should be completely shifted to a modal.
- Add a Cancel button in the form that will dismiss the modal when clicked.
- The modal header should contain a X button to dismiss the modal.

Task 2

In this task you will be adding a radio button group to the form to allow the selection of the smoking/non-smoking section of the restaurant.

- The radio button group should start out with the non-smoking section selected by default.
- The row containing the button group will have the label Section displayed preceding it in the form.

Note: Read [Bootstrap Buttons Checkbox/Radio](#) for more information on how to design checkbox/radio buttons.

Task 3

In this task you will be updating the Reserve Table button in the Jumbotron:

- Remove the tooltip from the button. This is to facilitate the button to be used to trigger the modal containing the table reservation form in the later tasks. A single button can support only one Javascript plugin via the data-* attributes. Make sure to remove the JavaScript script at the bottom of the page. Also remove the corresponding JavaScript code for the tooltip from the bottom of the page.
- You will update the Reserve Table button to show the modal containing the table reservation form when the button is clicked.

At the end of this assignment, your *index.html* file should look like this:

[Home](#)[About](#)[Menu](#)[Contact](#)

Ristora

We take inspiration
create a unique
creations will tickle

Reserve a Table

Task 1

Number of
Guests

☐ 1 ☐ 2 ☒ 3 ☐ 4 ☐ 5 ☐ 6

Section

Non-Smoking

Smoking

Date and
Time

Date

Time

Cancel

Reserve

Submission

1. You should submit the updated index.html file with all the tasks completed. A reviewer should easily be able to take your file and substitute it into their own web project and see it working correctly.
2. Also upload a screenshot of your browser window showing the completed index.html page with the modal containing the table reservation form overlayed on top of the web page in png or jpg format.

Review criteria

Upon completion of the assignment, your submission will be reviewed based on the following criteria:

Task 1:

- The form has been moved into a modal.
- The modal HTML code is included towards the top of the body of the page near the other modal code.
- The modal includes a Cancel button to dismiss the modal.
- The modal header includes a X button to dismiss the modal.

Task 2:

- A correct radio button group with the labels non-smoking in green and smoking in red is included in the form. Use the correct button color classes for the buttons.
- The label of the row containing the buttons is set to Section.
- The non-smoking button is checked by default.

Task 3:

- The tooltip has been removed from the Reserve Table button and the corresponding JavaScript code has been removed from the bottom of the page.
- The Reserve Table button will show the modal containing the table reservation form when clicked.