

Unit – IV: Bootstrap Plug-Ins

Understand importance of various bootstrap plug-ins

Bootstrap comes bundled with 12 jQuery plugins that extend the features and can add more interaction to your site. To get started with the Bootstrap's JavaScript plugins, you don't need to be an advanced JavaScript developer. By utilizing Bootstrap Data API, most of the plugins can be triggered without writing a single line of code.

Bootstrap Plugins can be included on your site in two forms –

- **Individually** – Using Bootstrap's individual *.js files. Some plugins and CSS components depend on other plugins. If you include plugins individually, make sure to check for these dependencies in the docs.
- **Or compiled (all at once)** – Using *bootstrap.js* or the minified *bootstrap.min.js*. Do not attempt to include both, as both *bootstrap.js* and *bootstrap.min.js* contain all plugins in a single file.

Use different modal dialog box give different popup windows, messages on dialog mode.

Bootstrap Alerts

Bootstrap provides an easy way to create predefined alert messages:

Alerts are created with the **.alert** class, followed by one of the four contextual classes **.alert-success**, **.alert-info**, **.alert-warning** or **.alert-danger**:

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <h2>Alerts</h2>
  <div class="alert alert-success">
    <strong>Success!</strong> This alert box could indicate a successful or positive action.
  </div>
  <div class="alert alert-info">
    <strong>Info!</strong> This alert box could indicate a neutral informative change or action.
```

```
</div>
<div class="alert alert-warning">
  <strong>Warning!</strong> This alert box could indicate a warning that might need attention.
</div>
<div class="alert alert-danger">
  <strong>Danger!</strong> This alert box could indicate a dangerous or potentially negative action.
</div>
</div>

</body>
</html>
```

Output:**Apply Scrollspy, Tooltip, popover plug-in, alert and button plug-ins while designing page to make it more interactive.**

Automatically update Bootstrap navigation or list group components based on scroll position to indicate which link is currently active in the viewport.

Scrollspy has a few requirements to function properly:

- If you're building our JavaScript from source, it requires util.js.
- It must be used on a Bootstrap nav component or list group.
- Scrollspy requires position: relative; on the element you're spying on, usually the <body>.
- When spying on elements other than the <body>, be sure to have a height set and overflow-y: scroll; applied.
- Anchors (<a>) are required and must point to an element with that id.

When successfully implemented, your nav or list group will update accordingly, moving the .active class from one item to the next based on their associated targets.

Example

```

<nav id="navbar-example2" class="navbar navbar-light bg-light">
  <a class="navbar-brand" href="#">Navbar</a>
  <ul class="nav nav-pills">
    <li class="nav-item">
      <a class="nav-link" href="#fat">@fat</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#mdo">@mdo</a>
    </li>
    <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-
haspopup="true" aria-expanded="false">Dropdown</a>
      <div class="dropdown-menu">
        <a class="dropdown-item" href="#one">one</a>
        <a class="dropdown-item" href="#two">two</a>
        <div role="separator" class="dropdown-divider"></div>
        <a class="dropdown-item" href="#three">three</a>
      </div>
    </li> </ul> </nav>
<div data-spy="scroll" data-target="#navbar-example2" data-offset="0">
  <h4 id="fat">@fat</h4>
  <p>...</p>
  <h4 id="mdo">@mdo</h4>
  <p>...</p>
  <h4 id="one">one</h4>
  <p>...</p>
  <h4 id="two">two</h4>
  <p>...</p>
  <h4 id="three">three</h4>
  <p>...</p>
</div>

```

Output:

Documentation and examples for adding custom Bootstrap tooltips with CSS and JavaScript using CSS3 for animations and data-attributes for local title storage.

Things to know when using the tooltip plugin:

- Tooltips rely on the 3rd party library [Popper.js](#) for positioning. You must include [popper.min.js](#) before bootstrap.js or use bootstrap.bundle.min.js / bootstrap.bundle.js which contains Popper.js in order for tooltips to work!
- If you're building our JavaScript from source, it [requires](#) util.js.
- Tooltips are opt-in for performance reasons, so **you must initialize them yourself**.
- Tooltips with zero-length titles are never displayed.
- Specify container: 'body' to avoid rendering problems in more complex components (like our input groups, button groups, etc).
- Triggering tooltips on hidden elements will not work.
- Tooltips for .disabled or disabled elements must be triggered on a wrapper element.
- When triggered from hyperlinks that span multiple lines, tooltips will be centered. Use white-space: nowrap; on your <a>s to avoid this behavior.
- Tooltips must be hidden before their corresponding elements have been removed from the DOM.

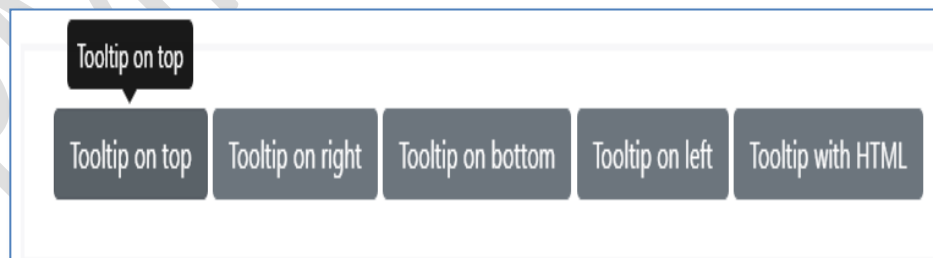
Example: Enable tooltips everywhere

One way to initialize all tooltips on a page would be to select them by their **data-toggle** attribute:

Example

```
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="top" title="Tooltip on top">  
    Tooltip on top  
</button>
```

Output:



Design home screen more attractive by using Carousel plug-in.

The Carousel plugin is a component for cycling through elements, like a carousel (slideshow).

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <h2>Carousel Example</h2>
  <div id="myCarousel" class="carousel slide" data-ride="carousel">
    <!-- Indicators -->
    <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>
    </ol>

    <!-- Wrapper for slides -->
    <div class="carousel-inner">
      <div class="item active">
        
      </div>

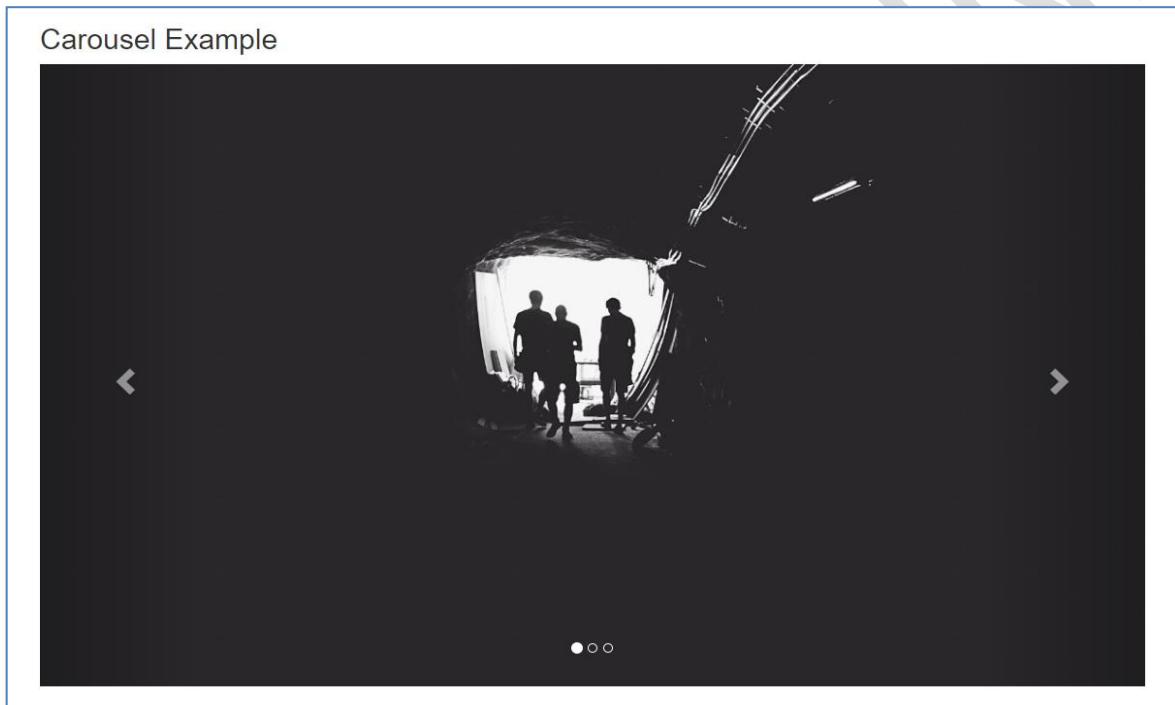
      <div class="item">
        
      </div>

      <div class="item">
        
      </div>

    </div>

    <!-- Left and right controls -->
    <a class="left carousel-control" href="#myCarousel" data-slide="prev">
      <span class="glyphicon glyphicon-chevron-left"></span>
      <span class="sr-only">Previous</span>
    </a>
```

```
<a class="right carousel-control" href="#myCarousel" data-slide="next">  
  <span class="glyphicon glyphicon-chevron-right"></span>  
  <span class="sr-only">Next</span>  
</a>  
</div>  
</div>  
  
</body>  
</html>
```

Output:

Tip: Plugins can be included individually (using Bootstrap's individual "carousel.js" file), or all at once (using "bootstrap.js" or "bootstrap.min.js").