Orchard| Mindtree

Material Design, Bootstrap & Media Query

MAY 15, 2018

by
Ravi Kant (M1044571)

Bootstrap

- Bootstrap is a free 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

➔ Advantages of Bootstrap

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

➔ Where to Get Bootstrap?

- In angular cli you can install using command:-

```
npm install bootstrap --save
```

- Include Bootstrap from a CDN (Content Delivery Network).

❖ Create First Web Page With Bootstrap

➔ 1. Add the HTML5 doctype

- Bootstrap uses HTML elements and CSS properties that require the HTML5 doctype.
- Always include the HTML5 doctype at the beginning of the page, along with the lang attribute and the correct character set:

➔ 2. Bootstrap 3 is mobile-first

- Bootstrap 3 is designed to be responsive to mobile devices. Mobile-first styles are part of the core framework.
- To ensure proper rendering and touch zooming, add the following <meta> tag inside the <head> element:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- The `width=device-width` part sets the width of the page to follow the screen-width of the device .
- The `initial-scale=1` part sets the initial zoom level when the page is first loaded by the browser.

➔ 3. Containers

There are two container classes to choose from:

- The `.container` class provides a responsive fixed width container
- The `.container-fluid` class provides a full width container, spanning the entire width of the viewport

```
<!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/3.3
.7/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min
.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.
min.js"></script>
</head>
<body>

<div class="container">
  <h1>My First Bootstrap Page</h1>
  <p>This is some text.</p>
</div>
```

```
</body>
</html>
```

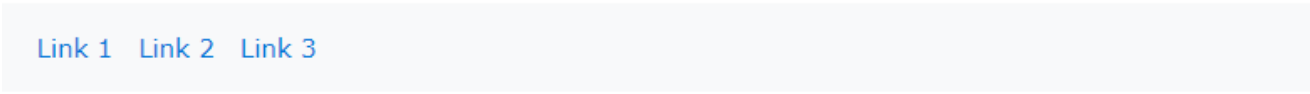
❖ Navigation Bars

- A navigation bar is a navigation header that is placed at the top of the page.
- With Bootstrap, a navigation bar can extend or collapse, depending on the screen size.
- A standard navigation bar is created with the `.navbar` class, followed by a responsive collapsing class: `.navbar-expand-xl|lg|md|sm` (stacks the navbar vertically on extra large, large, medium or small screens).
- To add links inside the navbar, use a `<ul>` element with `class="navbar-nav".` Then add `<li>` elements with a `.nav-item` class followed by an `<a>` element with a `.nav-link` class:

```
<nav class="navbar navbar-expand-sm bg-light">

  <!-- Links -->
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link" href="#">Link 1</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link 2</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link 3</a>
    </li>
  </ul>

</nav>
```

- There are different classes of Navbar, inorder to learn more about these classes please go through the link:

https://www.w3schools.com/bootstrap4/bootstrap_navbar.asp

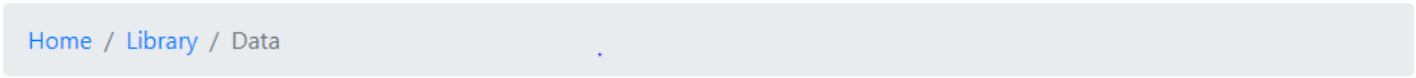


❖ Breadcrumb

- Indicate the current page's location within a navigational hierarchy.
- Separators are automatically added in CSS through `::before` and `content`.

Example:-

```
<nav aria-label="breadcrumb">
  <ol class="breadcrumb">
    <li class="breadcrumb-item"><a href="#">Home</a></li>
    <li class="breadcrumb-item"><a href="#">Library</a></li>
    <li class="breadcrumb-item active" aria-current="page">Data</li>
  </ol>
</nav>
```



❖ Button

Bootstrap 4 provides different styles of buttons:

We can use these buttons using `btn` class and their design specific classes like `btn-primary, btn-secondary, btn-success .. etc`

```
<button type="button" class="btn">Basic</button>
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
```

```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-
secondary">Secondary</button>
```

❖ Images

- Bootstrap provides three different classes for images, we can shape images using these classes.
- The `.rounded` class adds rounded corners to an image:

```
<img src="cinqueterre.jpg" class="rounded" alt="Cinque Terre">
```

- The `.rounded-circle` class shapes the image to a circle:

```
<img src="cinqueterre.jpg" class="rounded-circle" alt="Cinque
Terre">
```

- The `.img-thumbnail` class shapes the image to a thumbnail (bordered):

```
<img src="cinqueterre.jpg" class="img-thumbnail" alt="Cinque
Terre">
```

Rounded Corners:

Circle:

Thumbnail:

❖ Headers

The <header> element represents a container for introductory content or a set of navigational links.

A <header> element typically contains:

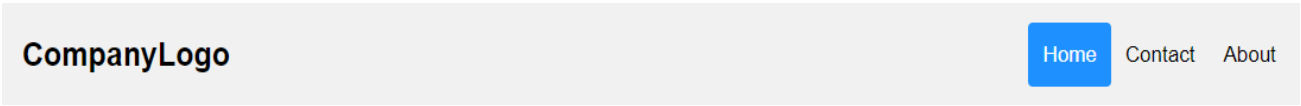
- one or more heading elements (<h1> - <h6>)
- logo or icon
- authorship information

You can have several <header> elements in one document.

Example:- basic header tag

```
<header>
    <h1>Most important heading here</h1>
    <h3>Less important heading here</h3>
    <p>Some additional information here</p>
  </header>
```

We can create custom header using HTML and CSS.




```
<div class="header">
  <a href="#default" class="logo">CompanyLogo</a>
  <div class="header-right">
    <a class="active" href="#home">Home</a>
    <a href="#contact">Contact</a>
    <a href="#about">About</a>
  </div>
</div>
```

Refer-https://www.w3schools.com/howto/howto_css_responsive_header.asp

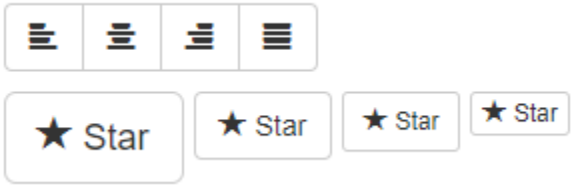
❖ Glyphicons

Bootstrap provides over 250 glyphs . Icon classes cannot be directly combined with other components. They should not be used along with other classes on the same element. Instead, add a nested `<span>`.and apply the icon classes to the `<span>`.

```
<span class="glyphicon glyphicon-search" aria-hidden="true"></span>
```

Example:-

```
<button type="button" class="btn btn-default" aria-label="Left Align">
<span class="glyphicon glyphicon-align-left" aria-
hidden="true"></span>
</button>
 <button type="button" class="btn btn-default btn-lg">
 <span class="glyphicon glyphicon-star" aria-hidden="true"></span>
Star </button>
```



❖ Carousel

The Carousel is a slideshow for cycling through elements.

Example:-

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

  <!-- Indicators -->
```

```
  <ul class="carousel-indicators">
    <li data-target="#demo" data-slide-to="0" class="active"></li>
    <li data-target="#demo" data-slide-to="1"></li>
    <li data-target="#demo" data-slide-to="2"></li>
  </ul>

  <!-- The slideshow -->
  <div class="carousel-inner">
    <div class="carousel-item active">
      <img src="la.jpg" alt="Los Angeles">
    </div>
    <div class="carousel-item">
      <img src="chicago.jpg" alt="Chicago">
    </div>
    <div class="carousel-item">
      <img src="ny.jpg" alt="New York">
    </div>
  </div>

  <!-- Left and right controls -->
  <a class="carousel-control-prev" href="#demo" data-slide="prev">
    <span class="carousel-control-prev-icon"></span>
  </a>
  <a class="carousel-control-next" href="#demo" data-slide="next">
    <span class="carousel-control-next-icon"></span>
  </a>

</div>
```

❖ Grid

- Bootstrap's grid system is built with flexbox and allows up to 12 columns across the page.
- If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:

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											

- The grid system is responsive, and the columns will re-arrange automatically depending on the screen size.
- Make sure that the sum adds up to 12 or fewer (it is not required that you use all 12 available columns).

❖ Grid Classes

The Bootstrap 4 grid system has five classes:

- `.col-` (extra small devices - screen width less than 576px)
- `.col-sm-` (small devices - screen width equal to or greater than 576px)
- `.col-md-` (medium devices - screen width equal to or greater than 768px)
- `.col-lg-` (large devices - screen width equal to or greater than 992px)
- `.col-xl-` (xlarge devices - screen width equal to or greater than 1200px)

The classes above can be combined to create more dynamic and flexible layouts.

Tip: Each class scales up, so if you wish to set the same widths for `sm` and `md`, you only need to specify `sm`.

Example:-

```
<div class="row">
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
  <div class="col-sm-3">.col-sm-3</div>
</div>
```

.col-sm-3	.col-sm-3	.col-sm-3	.col-sm-3

❖ Forms

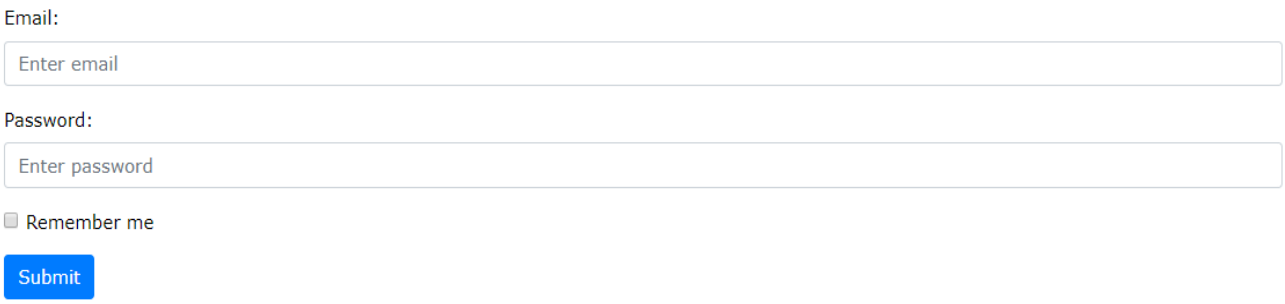
Bootstrap provides two types of form layouts:

- Stacked (full-width) form
- Inline form: using form class- `.form-inline`

All textual <input>, <textarea>, and <select> elements with class `.form-control` have a width of 100%.

Example:-

```
<form action="/action ">
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="form-group form-check">
    <label class="form-check-label">
      <input class="form-check-input" type="checkbox"> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```



❖ Inline Form

- Add class `.form-inline` to the <form> element.

Example:-

```
<form class="form-inline" action="/action_page.php">
  <label for="email">Email address:</label>
  <input type="email" class="form-control" id="email">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd">
  <div class="form-check">
    <label class="form-check-label">
      <input class="form-check-input" type="checkbox"> Remember me
    </label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

lists

❖ Lists

Bootstrap provides various List Groups .

The most basic list group is an unordered list with list items. To create a basic list group, use an `<ul>` element with class `.list-group`, and `<li>` elements with class `.list-group-item`:

```
<ul class="list-group">
  <li class="list-group-item">First item</li>
  <li class="list-group-item">Second item</li>
  <li class="list-group-item">Third item</li>
</ul>
```

Other list groups are list with active states, list with badges, list with linked items.

Refer:- https://www.w3schools.com/bootstrap/bootstrap_list_groups.asp .

Media Query

The `@media` rule, introduced in CSS2, made it possible to define different style rules for different media types.

Media queries can be used to check many things, such as:

- width and height of the viewport
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- resolution

Media Query Syntax:

We will put media query syntax in style tag like normal CSS.

```
@media not|only mediatype and (media feature and|or|not mediafeature) {
    CSS-Code;
}
```

meaning of the **not**, **only** and **and** keywords:

not: The not keyword reverts the meaning of an entire media query.

only: The only keyword prevents older browsers that do not support media queries with media features from applying the specified styles.It has no effect on modern browsers.

and: The and keyword combines a media feature with a media type or other media features.

They are all optional. However, if you use **not** or **only**, you must also specify a media type.

You can also have different *stylesheets* for different media, like this:

```
<link rel="stylesheet" media="screen and (min-width:
900px)" href="widescreen.css">
<link rel="stylesheet" media="screen and (max-width:
600px)" href="smallscreen.css">
```

Media Types:

- All: Default. Used for all media type devices.
- Print: Used for printers.
- Screen: Used for computer screens, tablets, smart-phones etc.
- Speech: Used for screen readers that "reads" the page out loud.

Example:-

If the browser window is 600px or smaller, the background color will be lightblue:

```
@media only screen and (max-width: 600px) {
    body {
        background-color: lightblue;
    }
}
```

Use media queries to create a responsive column layout:

```
/* On screens that are 992px wide or less, go from four columns to two
columns */
@media screen and (max-width: 992px) {
    .column {
        width: 50%;
    }
}

/* On screens that are 600px wide or less, make the columns stack on top of
each other instead of next to each other */
@media screen and (max-width: 600px) {
    .column {
        width: 100%;
    }
}
```

Like this we can use any CSS properties on any elements at any particular condition.

Material Design

Material Design is a visual language that synthesizes the classic principles of good design with the innovation of technology and science.

Install Angular Material and Angular CDK:

```
npm install --save @angular/material @angular/cdk
```

Animations:

Some Material components depend on the Angular animations. If you want these animations to work in your app, you have to install the @angular/animations module and include the BrowserAnimationsModule in your app.

```
npm install --save @angular/animations
```

There are many components provided by material design, few of the important components are:

- **Form Controls**

Autocomplete:

The autocomplete is a normal text input enhanced by a panel of suggested options.

Example:- simple autocomplete.

Html Code:-

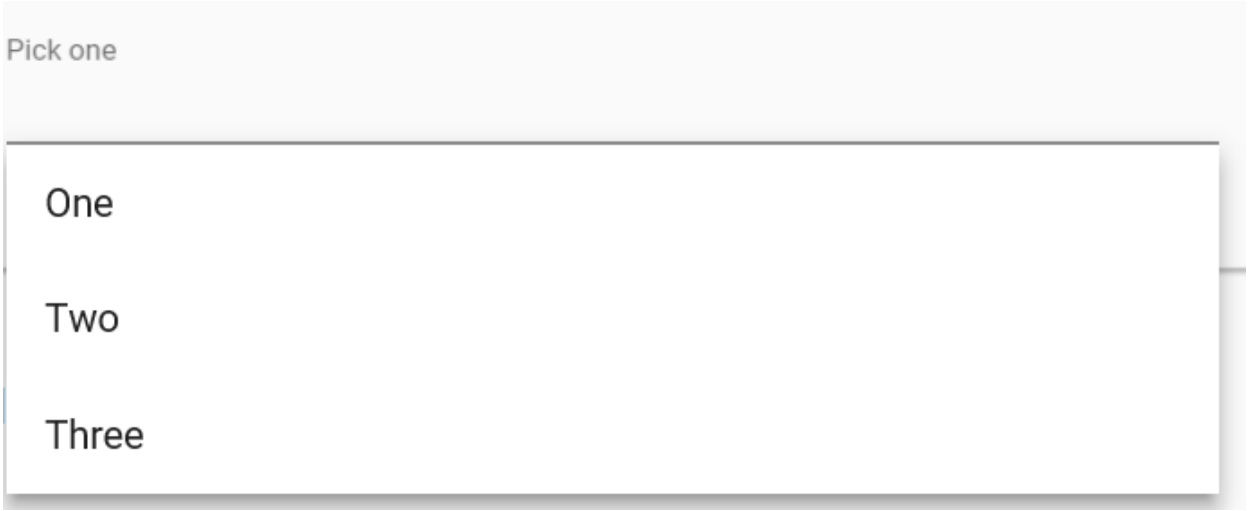
```
<mat-form-field>
  <input type="text" matInput [formControl]="myControl" [matAutocomplete]="auto">
</mat-form-field>

<mat-autocomplete #auto="matAutocomplete">
  <mat-option *ngFor="let option of options" [value]="option">
    {{ option }}
  </mat-option>
```

```
</mat-autocomplete>
```

Ts File Code:-

```
import {Component} from '@angular/core';
import {FormControl} from '@angular/forms';

@Component({
  selector: 'autocomplete-simple-example',
  templateUrl: 'autocomplete-simple-example.html',
  styleUrls: ['autocomplete-simple-example.css']
})
export class AutocompleteSimpleExample {

  myControl: FormControl = new FormControl();

  options = [
    'One',
    'Two',
    'Three'
   ];

}
```

On clicking the text field it will show the suggestion.



There are more autocompletes available with or without filters. For more information:

refer:- https://material.angular.io/components/autocomplete/overview

Form field:

<mat-form-field> is a component used to wrap several Angular Material components and apply common Text field styles such as the underline, floating label, and hint messages.

The following Angular Material components are designed to work inside a <mat-form-field>:

- <input matInput> & <textarea matInput>
- <mat-select>
- <mat-chip-list>

Example:-

```
<mat-form-field>
        <textarea matInput placeholder="Textarea"></textarea
</mat-form-field>
```

For more information:

refer:- https://material.angular.io/components/form-field/overview

more form elements are there refer:-

https://material.angular.io/components/categories/forms

❖ Navigation

there are three main navigation elements provided by material design:

- Menu
- Sidenav
- Toolbar

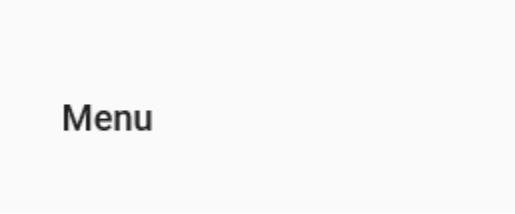
Menu:

<mat-menu> is a floating panel containing list of options.

Example:-

```
<button mat-button [matMenuTriggerFor]="menu">Menu</button>
<mat-menu #menu="matMenu">
    <button mat-menu-item>Item 1</button>
    <button mat-menu-item>Item 2</button>
</mat-menu>
```

By clicking on menu button it will open drop down of iten1 and item 2



on clicking Menu:-

For more detail refer:-

https://material.angular.io/components/menu/overview

Sidenav:

The sidenav components are designed to add side content to a fullscreen app. To set up a sidenav we use three components: <mat-sidenav-container> which acts as a

structural container for our content and sidenav, <mat-sidenav-content> which represents the main content, and <mat-sidenav> which represents the added side content.

```
<mat-sidenav-container>
    <mat-sidenav>Start</mat-sidenav>
    <mat-sidenav-content>Main</mat-sidenav-content>
</mat-sidenav-container>
```

For more information refer:-

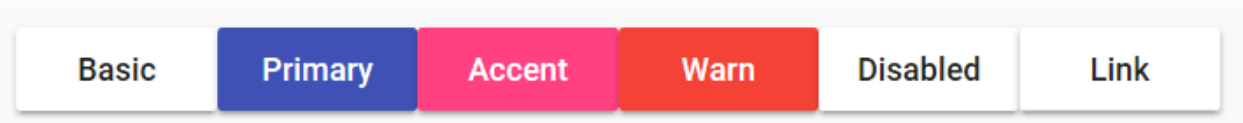
https://material.angular.io/components/categories/nav

❖ Button

Angular Material buttons are native <button> or <a> elements enhanced with Material Design styling and ink ripples.

Example:-

```
<div class="button-row">
    <button mat-raised-button>Basic</button>
    <button mat-raised-button color="primary">Primary</button>
    <button mat-raised-button color="accent">Accent</button>
    <button mat-raised-button color="warn">Warn</button>
    <button mat-raised-button disabled>Disabled</button>
    <a mat-raised-button routerLink=".">Link</a>
</div>
```



For more information refer:-

https://material.angular.io/components/categories/buttons

❖ Layout

Material provides many layout which are:

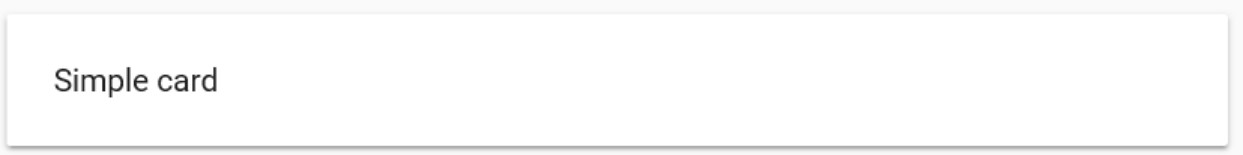
- Card
- Divider
- Grid list
- List
- Stepper
- Tabs

Card:

<mat-card> is a content container for text, photos, and actions in the context of a single subject.

Example:-

```
<mat-card>Simple card</mat-card>
```



Basic card sections:

Element	Description
<mat-card-title>	Card title
<mat-card-subtitle>	Card subtitle
<mat-card-content>	Primary card content. Intended for blocks of text
<img mat-card-image>	Card image. Stretches the image to the container width
<mat-card-actions>	Container for buttons at the bottom of the card
<mat-card-footer>	Section anchored to the bottom of the card

Refer:- https://material.angular.io/components/card/overview for more information on mat cards.

For more information on layouts.

Refer:- https://material.angular.io/components/categories/layout

❖ Table

The mat-table provides a Material Design styled data-table that can be used to display rows of data.

Example:- HTML Code

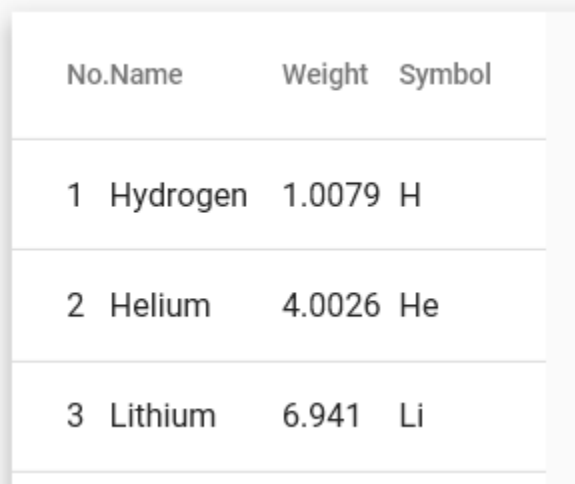
```
<div class="example-container mat-elevation-z8">
  <table mat-table #table [dataSource]="dataSource">

    <!-- Position Column -->
    <ng-container matColumnDef="position">
      <th mat-header-cell *matHeaderCellDef> No. </th>
      <td mat-cell *matCellDef="let element"> {{element.position}}
</td>
    </ng-container>

    <!-- Name Column -->
    <ng-container matColumnDef="name">
      <th mat-header-cell *matHeaderCellDef> Name </th>
      <td mat-cell *matCellDef="let element"> {{element.name}} </td>
    </ng-container>

    <!-- Weight Column -->
    <ng-container matColumnDef="weight">
      <th mat-header-cell *matHeaderCellDef> Weight </th>
      <td mat-cell *matCellDef="let element"> {{element.weight}}
</td>
    </ng-container>
```

```
      <!-- Symbol Column -->
      <ng-container matColumnDef="symbol">
        <th mat-header-cell *matHeaderCellDef> Symbol </th>
        <td mat-cell *matCellDef="let element"> {{element.symbol}}
</td>
      </ng-container>

      <tr mat-header-row *matHeaderRowDef="displayedColumns"></tr>
      <tr mat-row *matRowDef="let row; columns: displayedColumns;"></tr>
    </table>
</div>
```

Ts file code:-

```
import {Component} from '@angular/core';
@Component({
  selector: 'table-basic-example',
  styleUrls: ['table-basic-example.css'],
  templateUrl: 'table-basic-example.html',
})
export class TableBasicExample {
  displayedColumns = ['position', 'name', 'weight', 'symbol'];
  dataSource = ELEMENT_DATA;
}

export interface Element {
  name: string;
  position: number;
  weight: number;
  symbol: string;
}

const ELEMENT_DATA: Element[] = [
  {position: 1, name: 'Hydrogen', weight: 1.0079, symbol:
'H'},
```

```
  {position: 2, name: 'Helium', weight: 4.0026, symbol:
'He'},
  {position: 3, name: 'Lithium', weight: 6.941, symbol: 'Li'}];
```



No.	Name	Weight	Symbol
1	Hydrogen	1.0079	H
2	Helium	4.0026	He
3	Lithium	6.941	Li