Import Bootstrap in your Application

```
!DOCTYPE html>
<html lang="en">
<head>

  <title>Web Development Blog</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

</head>
<body>

    <h1>Web Development Blog</h1>
    <p>This blog discusses web development tools and libraries.</p>

</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Web Development Blog</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css
" integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">
</head>
<body>
    <h1>Web Development Blog</h1>
    <p>This blog discusses web development tools and libraries.</p>
</body>
</html>
```

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

Develop a static web page

```
<!DOCTYPE html>
<html lang="en">
<head>

  <title>Web Development Blog</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.3/css/bootstrap.min.css
" integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">
</head>

<body>
    <div class="page-header">
        <h1>Introduction to Bootstrap</h1>
        <img src="https://images.pexels.com/photos/326424/pexels-photo-
326424.jpeg" class="img-fluid w-100 p-3" alt="Responsive image">
    </div>
    <div class=“container-fluid”>
        <p> Get started with <mark>Bootstrap</mark>, the world’s most popular
framework for building responsive, mobile-first sites, with Bootstrap <abbr
title="Content Delivery Network">CDN</abbr> and a template starter page. </p>

        <p>According to the official website, Bootstrap is the most popular
HTML, CSS, and JS framework for developing responsive, mobile first projects
on the web. Sounds great! Now how do I use it? It would be easy to send you
over to their Getting Started page and call it a day. Their setup guide is
indeed a host of useful information — links to CDNs, explanations on how to
install with Bower, npm, and Composer, information on integration with
Autoprefixer and LESS, a bunch of templates, licenses, and translations — but
you could read this blog here instead for a better and easier
understanding.</p>
        <span class="glyphicon glyphicon-pencil"></span>

        <p class="text-center">The list below shows the different version of
Bootstrap so far. <span class="glyphicon glyphicon-list-alt"></span></p>

        <ul class="list-group">
          <li class="list-group-item">v1.x</li>
          <li class="list-group-item">v2.x</li>
          <li class="list-group-item">v3.x</li>
          <li class="list-group-item">v4.x</li>
```

```
        </ul>

        <p class="text-center bg-info"> The main features of bootstrap is, it is very simple and easy to use, huge JavaScript plugins are available, easily design mobile friendly website.</p>

    </div>


    <div class="container-fluid">
      <h2></h2>
      <table class="table table-bordered table-striped">
        <thead>
          <tr >
            <th>Features of Bootstrap</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>Easy to Use</td>
          </tr>
          <tr>
            <td>Mobile Friendly</td>
          </tr>
          <tr>
            <td>Prestyled Components</td>
          </tr>
        </tbody>
      </table>
    </div>
    <div class="container-fluid"
        <p class="text-left">Lets take a look at the components of bootstrap:</p>
        <h1>CSS<span class="badge badge-secondary">Quoted</span></h1>
        <p class="text-justify">Copy-paste the stylesheet link into your head before all other stylesheets to load our CSS.</p>
        <h2>JS <span class="badge badge-secondary">Quoted</span></h2>
        <p class="text-justify">Many of our components require the use of JavaScript to function. Specifically, they require jQuery, Popper.js, and our own JavaScript plugins. Place the following scripts near the end of your pages, right before the closing bodytag, to enable them. jQuery must come first, then Popper.js, and then our JavaScript plugins.We use jQuery’s slim build, but the full version is also supported.</p>
    </div>
    <div class="card">
          <div class="card-header">Author</div>
          <div class="card-body">The article was contributed by katesmith12</div>
```

```
          <a href="https://www.sitesbay.com/bootstrap/bootstrap-features-of-
bootstrap" class="card-footer btn btn-outline-primary stretched-link">View
References</a>
    </div>
    <!-- Footer -->
    <footer class="page-footer font-small indigo">

      <!-- Copyright -->
      <div class="footer-copyright text-center py-3">This is the footer
section where you can put the address and copyright details. © 2018 Copyright:
        <a href="https://simplilearn.net/"> Simplilearn</a>
      </div>
      <!-- Copyright -->

    </footer>
    <!-- Footer -->

<script src="https://code.jquery.com/jquery-3.3.2.slim.min.js"
integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.3/js/bootstrap.min.js"
integrity="sha384-
ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"
crossorigin="anonymous"></script>
</body>
</html>
```

Forms

```
!DOCTYPE html>
<html lang="en">
<head>

  <title>Web Development Blog</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.3/css/bootstrap.min.css
" integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">
```

```
</head>

<body>
    <div class="page-header">
        <h1>Introduction to Bootstrap</h1>
        <img src="https://images.pexels.com/photos/326424/pexels-photo-
326424.jpeg" class="img-fluid w-100 p-3" alt="Responsive image">
    </div>
    <div class=“container-fluid”>
        <p> Get started with <mark>Bootstrap</mark>, the world’s most popular
framework for building responsive, mobile-first sites, with Bootstrap <abbr
title="Content Delivery Network">CDN</abbr> and a template starter page. </p>

        <p>According to the official website, Bootstrap is the most popular
HTML, CSS, and JS framework for developing responsive, mobile first projects
on the web. Sounds great! Now how do I use it? It would be easy to send you
over to their Getting Started page and call it a day. Their setup guide is
indeed a host of useful information — links to CDNs, explanations on how to
install with Bower, npm, and Composer, information on integration with
Autoprefixer and LESS, a bunch of templates, licenses, and translations — but
you could read this blog here instead for a better and easier
understanding.</p>
        <span class="glyphicon glyphicon-pencil"></span>

        <p class="text-center">The list below shows the different version of
Bootstrap so far. <span class="glyphicon glyphicon-list-alt"></span></p>

        <ul class="list-group">
          <li class="list-group-item">v1.x</li>
          <li class="list-group-item">v2.x</li>
          <li class="list-group-item">v3.x</li>
          <li class="list-group-item">v4.x</li>
        </ul>

        <p class="text-center bg-info"> The main features of bootstrap is, it
is very simple and easy to use, huge JavaScript plugins are available, easily
design mobile friendly website.</p>

    </div>


    <div class="container-fluid">
      <h2></h2>
      <table class="table table-bordered table-striped">
        <thead>
          <tr >
            <th>Features of Bootstrap</th>
          </tr>
```

```
        </thead>
        <tbody>
          <tr>
            <td>Easy to Use</td>
          </tr>
          <tr>
            <td>Mobile Friendly</td>
          </tr>
          <tr>
            <td>Prestyled Components</td>
          </tr>
        </tbody>
      </table>
    </div>
    <div class="container-fluid"
        <p class="text-left">Lets take a look at the components of
bootstrap:</p>
        <h1>CSS<span class="badge badge-secondary">Quoted</span></h1>
        <p class="text-justify">Copy-paste the stylesheet link into your head
before all other stylesheets to load our CSS.</p>
        <h2>JS <span class="badge badge-secondary">Quoted</span></h2>
        <p class="text-justify">Many of our components require the use of
JavaScript to function. Specifically, they require jQuery, Popper.js, and our
own JavaScript plugins. Place the following scripts near the end of your
pages, right before the closing bodytag, to enable them. jQuery must come
first, then Popper.js, and then our JavaScript plugins.We use jQuery's slim
build, but the full version is also supported.</p>
    </div>
    <div class="card">
          <div class="card-header">Author</div>
          <div class="card-body">The article was contributed by
katesmith12</div>
          <a href="https://www.sitesbay.com/bootstrap/bootstrap-features-of-
bootstrap" class="card-footer btn btn-outline-primary stretched-link">View
References</a>
    </div>
    <!-- Footer -->
    <footer class="page-footer font-small indigo">

      <!-- Copyright -->
      <div class="footer-copyright text-center py-3">This is the footer
section where you can put the address and copyright details. © 2018 Copyright:
        <a href="https://simplilearn.net/"> Simplilearn</a>
      </div>
      <!-- Copyright -->

    </footer>
    <!-- Footer -->
```

```
<script src="https://code.jquery.com/jquery-3.3.2.slim.min.js"
integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.3/js/bootstrap.min.js"
integrity="sha384-
ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"
crossorigin="anonymous"></script>
</body>
</html>
```

Navigation and Pagination

```
!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css
" integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></scrip
t>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></sc
ript>
  </head>
  <body>
    <nav class="navbar navbar-expand-md">
      <a class="navbar-brand" href="#"><img src="images/noodle.png"
alt=""></a>
      <button class="navbar-toggler navbar-dark" type="button" data-
toggle="collapse" data-target="#main-navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="main-navigation">
        <ul class="navbar-nav">
          <li class="nav-item">
            <a class="nav-link" href="index.html">Home</a>
```

```
          </li>
          <li class="nav-item">
            <a class="nav-link" href="form.html">Contact</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="page.html">Pagination</a>
          </li>
        </ul>
      </div>
    </nav>
    <p>The 'nav' tag is used to define a block of links for navigation, for the current page, or for other pages.</p>
  </body>
</html>


<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO" crossorigin="anonymous">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  </head>
  <body>
    <p>Pagination is used to divide a document into different pages with numbers.</p>
    <nav aria-label="Pagination Demo">
      <ul class="pagination">
        <li class="page-item"><a class="page-link" href="#">Previous</a></li>
        <li class="page-item"><a class="page-link" href="index.html">1</a></li>
        <li class="page-item"><a class="page-link" href="form.html">2</a></li>
        <li class="page-item"><a class="page-link" href="nav.html">3</a></li>
        <li class="page-item"><a class="page-link" href="#">Next</a></li>
      </ul>
    </nav>
  </body>
</html>
```

Function and Prototyping

```
<html>
   <body>
      <h1>MEAN Stack</h1>
      <p> Lesson 3 Demos </p>


      <script src="functions_and_prototypes.js"></script>



   </body>
</html>
```

```
// function constructor
function Employee(name, designation, yearOfBirth){
   this.name= name;
   this.designation= designation;
   this.yearOfBirth= yearOfBirth;
}


// creating calculateAge() method to the Prototype property
Employee.prototype.calculateAge= function(){
   console.log('The current age is: '+(2019- this.yearOfBirth));
}
console.log(Employee.prototype);
```

```
// creating Objects
let Emp1= new Employee('Alex', 'Junior Tester', 1995);
console.log(Emp1) ;
Emp1.calculateAge();

let Emp2= new Employee('Dexter', 'Senior Software Developer', 1990);
console.log(Emp2)
Emp2.calculateAge();

let Emp3= new Employee('Annie', 'Junior HR', 1998);
console.log(Emp3)
Emp3.calculateAge();
```

Working and functions

```
<html>
   <body>
      <h1>MEAN Stack</h1>
      <p> Lesson 3 Demos </p>

      <script src="function.js"></script>


   </body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>
<h1>Javascript – Function<h1>

<script>

	var x = (2 * 3) + 5;
	var y = 3 * 4;

	var result = myFunction(2,3);
	console.log(result);

	function myFunction(num1, num2) {
		var a = num1 * num2;
		var b = num1 + num2;
		return(a + b);
	}

	console.log( myFunction(3, 4));

	function toCelcius(f){
			return (5/9) * (f-32);
	}
```

```
        console.log("The temperature is "+ toCelcius(60));


</script>


</body>
</html>
```

IIFEs,Callbacks,and Closurs

```
<html>
   <body>
      <h1>MEAN Stack</h1>
      <p> Lesson 3 Demos </p>

      <script src="IIFEs_Callbacks_Closures.js"></script>



   </body>
</html>
```

```
//IIFE and Closure
const empId = (function() {
   let count = 0;
   return function() {
     ++count;
```

```
    return `emp${count}`;
  };
})();

console.log("New Emplyee IDs are listed here");
console.log("Alex: "+empId());
console.log("Dexter: "+empId());
console.log("Annie: "+empId());


//Callbacks
console.log("\n"); //to start the output from the neext line
function fullName(firstName, lastName, callback){
  console.log("My name is " + firstName + " " + lastName);
  callback(lastName);
}

var greeting = function(ln){
  console.log('Welcome ' + ln);
};

fullName("Alex", "Wilson", greeting);
console.log("\n");
fullName("Dexter", "Johnson", greeting);
console.log("\n");
fullName("Annie", "Butler", greeting);
```

Maps and Classes

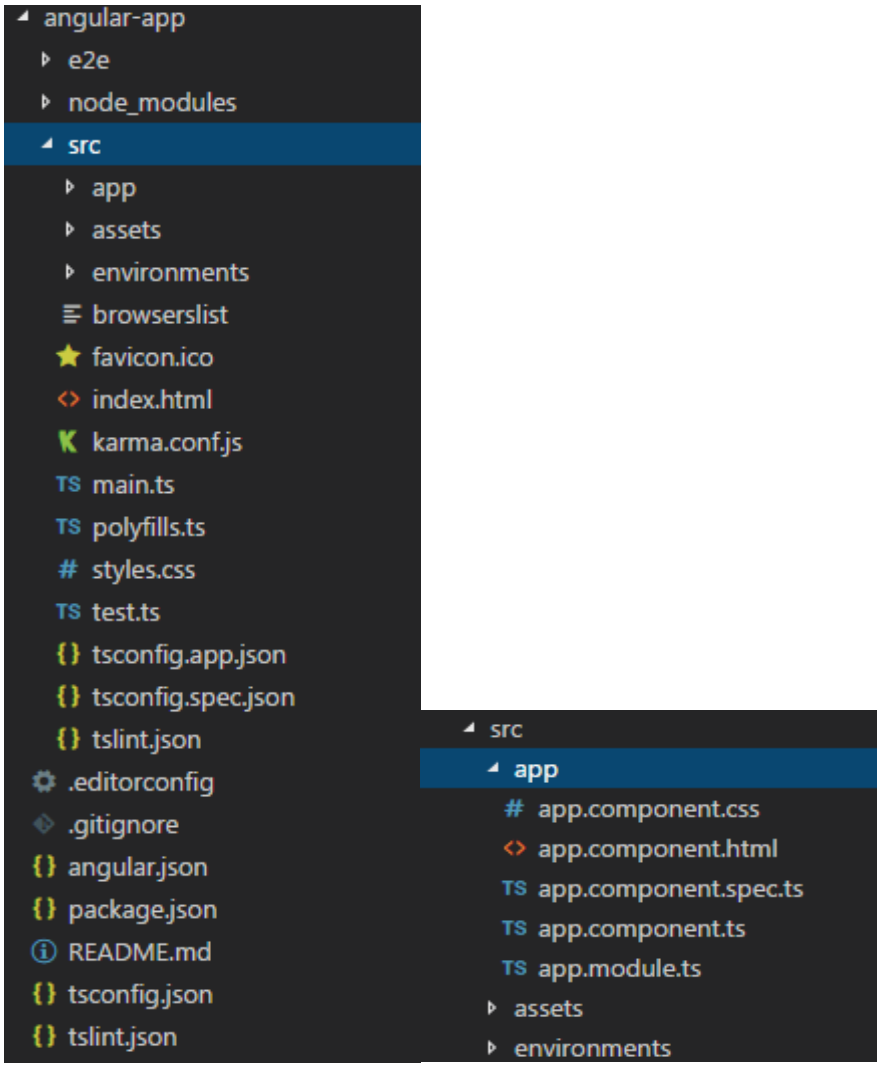
```
<html>
   <body>
      <h1>MEAN Stack</h1>
      <p> Lesson 3 Demos </p>

      <script src="maps_and_classes.js"></script>


   </body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>
<h1>Javascript – Maps and Classes<h1>

<script>

var map1 = new Map();
map1.set("first name", "Robb");
map1.set("last name", "Stark");
map1.set("friend 1","Bran")
   .set("friend 2","Arya");
console.log(map1);
console.log("map1 has friend 3 ? " + map1.has("friend 3"));
console.log("get value for key = friend 3 - "+ map1.get("friend 3"));
```

```
console.log("delete element with key = friend 2 - " + map1.delete("friend 2"));
map1.clear();
console.log(map1);
class Employee
{
    constructor(id,name)
    {
      this.id=id;
      this.name=name;
    }
    detail()
    {
  document.writeln(this.id+" "+this.name+"<br>")
    }
  }
//passing object to a variable
var e1=new Employee(101,"Michael");
var e2=new Employee(102,"Bob");
e1.detail();
e2.detail();
```

Components

```
$ ng new angular-app
CREATE angular-app/angular.json (3813 bytes)
CREATE angular-app/package.json (1310 bytes)
CREATE angular-app/README.md (1027 bytes)
CREATE angular-app/tsconfig.json (435 bytes)
CREATE angular-app/tslint.json (2824 bytes)
CREATE angular-app/.editorconfig (246 bytes)
CREATE angular-app/.gitignore (587 bytes)
CREATE angular-app/src/favicon.ico (5430 bytes)
CREATE angular-app/src/index.html (297 bytes)
CREATE angular-app/src/main.ts (372 bytes)
CREATE angular-app/src/polyfills.ts (3571 bytes)
CREATE angular-app/src/test.ts (642 bytes)
CREATE angular-app/src/styles.css (80 bytes)
CREATE angular-app/src/browserslist (388 bytes)
CREATE angular-app/src/karma.conf.js (980 bytes)
CREATE angular-app/src/tsconfig.app.json (166 bytes)
CREATE angular-app/src/tsconfig.spec.json (256 bytes)
CREATE angular-app/src/tslint.json (314 bytes)
CREATE angular-app/src/assets/.gitkeep (0 bytes)
CREATE angular-app/src/environments/environment.prod.ts (51 bytes
CREATE angular-app/src/environments/environment.ts (662 bytes)
CREATE angular-app/src/app/app.module.ts (314 bytes)
CREATE angular-app/src/app/app.component.html (1120 bytes)
CREATE angular-app/src/app/app.component.spec.ts (993 bytes)
CREATE angular-app/src/app/app.component.ts (215 bytes)
CREATE angular-app/src/app/app.component.css (0 bytes)
CREATE angular-app/e2e/protractor.conf.js (752 bytes)
CREATE angular-app/e2e/tsconfig.e2e.json (213 bytes)
CREATE angular-app/e2e/src/app.e2e-spec.ts (303 bytes)
CREATE angular-app/e2e/src/app.po.ts (204 bytes)
```



Welcome | app.module.ts

```ts
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

product-list.component.html | product-list.component.ts

```html
<h1>Product List Page</h1>
```

product-list.component.ts

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


@Component({
    selector:'product-list',
    templateUrl:'./product-list.component.html'
})

export class ProductListComponent{
    pageTitle: string = "Product List Page"
}
```

Welcome | app.module.ts | app.component.ts | product-list.component.html

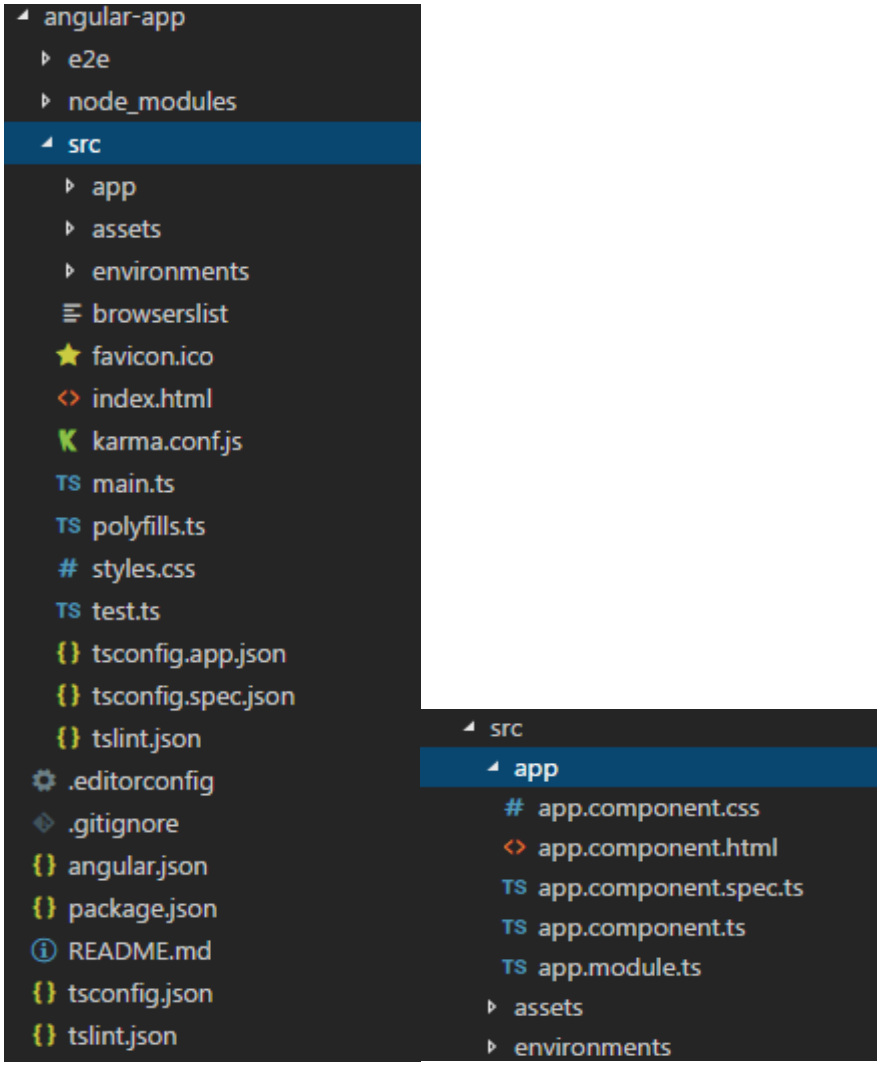
```ts
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';
import { ProductListComponent } from './products/product-list.component';

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Welcome | app.module.ts | app.component.ts | product-list.component.html

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

@Component({
  selector: 'app-root',
  template: `
    <product-list></product-list>
  `,
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angular-app';
}
```

Property Binding

```
$ ng new angular-app
CREATE angular-app/angular.json (3813 bytes)
CREATE angular-app/package.json (1310 bytes)
CREATE angular-app/README.md (1027 bytes)
CREATE angular-app/tsconfig.json (435 bytes)
CREATE angular-app/tslint.json (2824 bytes)
CREATE angular-app/.editorconfig (246 bytes)
CREATE angular-app/.gitignore (587 bytes)
CREATE angular-app/src/favicon.ico (5430 bytes)
CREATE angular-app/src/index.html (297 bytes)
CREATE angular-app/src/main.ts (372 bytes)
CREATE angular-app/src/polyfills.ts (3571 bytes)
CREATE angular-app/src/test.ts (642 bytes)
CREATE angular-app/src/styles.css (80 bytes)
CREATE angular-app/src/browserslist (388 bytes)
CREATE angular-app/src/karma.conf.js (980 bytes)
CREATE angular-app/src/tsconfig.app.json (166 bytes)
CREATE angular-app/src/tsconfig.spec.json (256 bytes)
CREATE angular-app/src/tslint.json (314 bytes)
CREATE angular-app/src/assets/.gitkeep (0 bytes)
CREATE angular-app/src/environments/environment.prod.ts (51 bytes
CREATE angular-app/src/environments/environment.ts (662 bytes)
CREATE angular-app/src/app/app.module.ts (314 bytes)
CREATE angular-app/src/app/app.component.html (1120 bytes)
CREATE angular-app/src/app/app.component.spec.ts (993 bytes)
CREATE angular-app/src/app/app.component.ts (215 bytes)
CREATE angular-app/src/app/app.component.css (0 bytes)
CREATE angular-app/e2e/protractor.conf.js (752 bytes)
CREATE angular-app/e2e/tsconfig.e2e.json (213 bytes)
CREATE angular-app/e2e/src/app.e2e-spec.ts (303 bytes)
CREATE angular-app/e2e/src/app.po.ts (204 bytes)
```



Welcome | app.module.ts

```ts
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

product-list.component.html

```
<h1>Product List Page</h1>
```

product-list.component.ts

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


@Component({
    selector:'product-list',
    templateUrl:'./product-list.component.html'
})

export class ProductListComponent{
    pageTitle: string = "Product List Page"
}
```

Welcome | app.module.ts | app.component.ts | product-list.component.html

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';
import { ProductListComponent } from './products/product-list.component';

@NgModule({
  declarations: [
    AppComponent,
    ProductListComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

Welcome | app.module.ts | app.component.ts | product-list.component.html

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

@Component({
  selector: 'app-root',
  template: `
    <product-list></product-list>
  `,
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angular-app';
}
```

Class and Style Binding

```
  <button [attr.disabled]=”true” class="btn btn-primary"
(click)='toggleImage()'>

      Show Image

  </button>

.inStock{

   background: #096d09;

   color:#ffff;

   Font-size:15px;

}

 <td [class.inStock]="true">

    {{ product.price}}
```

```
</td>
  <td [style.color]="'#306A9D'" [style.font-weight]="700">
      {{ product.productName }}
  </td>

 <td [style.color]="product.price > 20 ? 'red': 'green'"
[style.font-size.px]="16"  [style.font-weight]="700">
       {{ product.price}}
 </td>
```

Event binding

```
export class ProductListComponent{
  pageTitle: string = "Product List Page";
  imageWidth:number = 80;
  imageMargin:number = 10;

  showImage:boolean = false;

  toggleImage() : void {
     this.showImage = !this.showImage;
     // (!false = true) // (!true == false)
         console.log('Value of ShowImage inside function ::',
this.showImage);
  }
```

```
}
 <button class="btn btn-primary" (click)='toggleImage()'>
    Show Image
 </button>
```

Two-way binding

```
C:\Users\rakesh.deshpande\Desktop\Master LVCs\Phase - 3\AssistedPractice>ng new angCare
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use?
> CSS
  SCSS   [ http://sass-lang.com/documentation/file.SASS_REFERENCE.html#syntax ]
  Sass   [ http://sass-lang.com/documentation/file.INDENTED_SYNTAX.html       ]
  Less   [ http://lesscss.org                                                 ]
  Stylus [ http://stylus-lang.com                                             ]
```

```
Hash: e90882c647d367160324
Time: 19229ms
chunk {es2015-polyfills} es2015-polyfills.js, es2015-polyfills.js.map (es2015-polyfills) 285 kB [initial] [rendered]
chunk {main} main.js, main.js.map (main) 11.6 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 236 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 16.3 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 3.77 MB [initial] [rendered]
i ｢wdm｣: Compiled successfully.
```

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ChildComponent } from './child/child.component';

@NgModule({
```

```
  declarations: [
    AppComponent,
    ChildComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

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

```
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
    public cdata: string;
}
```

```
<h2>Parent Component</h2>
This is Parent Component<br>
Enter Text:
<input type="text" #ptext (keyup)="0"/><br>
```

```
The value of Child component is: {{cdata}}


<app-child (cevent)="cdata=$event" [pdata]="ptext.value"></app-child>
```

```
<h2>Child Component</h2>
This is Child Component<br>
Enter Text:
<input type="text" #cdata (keyup)="onChange(cdata.value)"/><br>
The value od Parent component is: {{pdata}}
```

```
import { Component, OnInit, Input, EventEmitter } from '@angular/core';


@Component({
  selector: 'app-child',
  templateUrl: './child.component.html',
  styleUrls: ['./child.component.css'],
  inputs: [`pdata`],
  outputs: [`cevent`]
})
export class ChildComponent implements OnInit {


  constructor() { }


  ngOnInit() {
  }
```

```
  public pdata: string;
  cevent= new EventEmitter<string>();

  onChange(value:string){
    this.cevent.emit(value);
  }
}
```

Form Validation

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Page Title</title>
</head>
<body style="padding:40px ">
<p style="text-align: center;font-size: 30px">  Hello <input type="text" placeholder="Your name" (input)="ontyping($event)"/>, Welcome to the <b>angCare!</b></p>
<p style="text-align: center;font-size: 20px">Hello {{name}}<br>I am Joe, your personal assistant! I will guide you further...</p>
<div style="text-align:left;padding: 20px">
<p> Click on Sign up to create your account with angCare: <button (click)="signup()" class="btn btn-primary">Sign up with {{title}}</button>   : {{status}}</p>
```

```
<p> Click on Sign up to create your account with angCare: <button>Sign In
</button></p>

</div>

<div >

<div class="jumbotron">

 <div class="container">

    <div class="row">

       <div class="col-md-6 offset-md-3">

          <h2>Angular 6 Reactive Form Validation</h2>

          <form [formGroup]="registerForm" (ngSubmit)="onSubmit()">

             <div class="form-group">

                <label>First Name</label>

                <input type="text" formControlName="firstName" class="form-
control" [ngClass]="{ 'is-invalid': submitted && f.firstName.errors }" />

                <div *ngIf="submitted && f.firstName.errors" class="invalid-
feedback">

                   <div *ngIf="f.firstName.errors.required">First Name is
required</div>

                </div>

             </div>


             <div class="form-group">

                <label>Last Name</label>

                <input type="text" formControlName="lastName" class="form-
control" [ngClass]="{ 'is-invalid': submitted && f.lastName.errors }" />

                <div *ngIf="submitted && f.lastName.errors" class="invalid-
feedback">

                   <div *ngIf="f.lastName.errors.required">Last Name is
required</div>

                </div>
```

```
            </div>
            <div class="form-group">
                <label>Email</label>
                <input type="text" formControlName="email" class="form-control"
[ngClass]="{ 'is-invalid': submitted && f.email.errors }" />
                <div *ngIf="submitted && f.email.errors" class="invalid-feedback">
                    <div *ngIf="f.email.errors.required">Email is required</div>
                    <div *ngIf="f.email.errors.email">Email must be a valid email
address</div>
                </div>
            </div>
            <div class="form-group">
                <label>Password</label>
                <input type="password" formControlName="password"
class="form-control" [ngClass]="{ 'is-invalid': submitted && f.password.errors }"
/>
                <div *ngIf="submitted && f.password.errors" class="invalid-
feedback">
                    <div *ngIf="f.password.errors.required">Password is
required</div>
                    <div *ngIf="f.password.errors.minlength">Password must be at
least 6 characters</div>
                </div>
            </div>
            <div class="form-group">
                <button [disabled]="loading" class="btn btn-
primary">Register</button>
            </div>
        </form>
    </div>
```

```
    </div>
  </div>
</div>
</div>
</body>
</html>
<router-outlet></router-outlet>
```

```
import { Component,OnInit } from '@angular/core';
import {FormBuilder, FormGroup, Validators } from '@angular/forms';
import { Title } from '@angular/platform-browser';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

export class AppComponent {

  constructor(private formBuilder: FormBuilder) { }
  title = 'angCare';
  status = 'You haven\'t signed up yet';
  name = '';
  submitted = false;
  registerForm: FormGroup;
```

```
 ontyping(event:Event) {

  this.name = (<HTMLInputElement>event.target).value;
 }
 signup() {

   this.status = 'Oops! We are working on it!';

 }

 ngOnInit() {
  this.registerForm = this.formBuilder.group({
      firstName: ['', Validators.required],
      lastName: ['', Validators.required],
      email: ['', [Validators.required, Validators.email]],
      password: ['', [Validators.required, Validators.minLength(6)]]
  });
}

get f() { return this.registerForm.controls; }

onSubmit() {
  this.submitted = true;

  // stop here if form is invalid
  if (this.registerForm.invalid) {
      return;
```

```
  }

  alert('Your request has been submitted for approval')
}
}
```

```
import { ReactiveFormsModule} from '@angular/forms'
```

```
ReactiveFormsModule
```

Directives

```
// Required services for custom directives
import { Directive, ElementRef, Renderer2 } from '@angular/core';

@Directive({
  selector: '[appChangeColor]' // Directive selector
})

export class ChangeColorDirective {

  constructor(elem: ElementRef, renderer: Renderer2) {
    renderer.setStyle(elem.nativeElement, 'color', 'olive');
  }

}
```

```
import { ChangeColorDirective } from './ChangeColor.directive';

@NgModule({
   imports: [
      SharedModule,
      AppRoutingModule
   ],
   declarations: [
      ChangeColorDirective,
      ProductComponent,
      MyUpperPipe,
      DiscountPipe,
      ProductSearch,
      ProductDetailComponent
   ],
 <h4 appChangeColor>Number of Product Serach on Basis of {{userInput}}:</h4
>
```

Pipes

```
<td [style.color]="'#306A9D'" [style.font-weight]="700" [style.font-size.px]="20">

  {{ product.productName | uppercase }}

</td>
```

```
import { Pipe, PipeTransform } from "@angular/core";

@Pipe({
  name: 'convertToSpaces'
})
export class ConvertToSpacesPipe implements PipeTransform{
  transform(value:string, character:string, ) {
     return value.replace(character, '@');
  }
}
import { ConvertToSpacesPipe } from 'src/app/products/convert-to-spaces.pipe';


declarations: [
  AppComponent,
  ProductListComponent,
  ConvertToSpacesPipe
 ]


<td>{{ product.productCode | convertToSpaces:'-'}}</td>
```

Routing Mechanism

```
<!DOCTYPE html>
<html>
```

```
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Page Title</title>
</head>
<body style="padding:40px ">
<p style="text-align: center;font-size: 30px">  Hello <input type="text" placeholder="Your name" (input)="ontyping($event)"/>, Welcome to the <b>angCare!</b></p>
<p style="text-align: center;font-size: 20px">Hello {{name}}<br>I am Joe, your personal assistant! I will guide you further...</p>
<div style="text-align:left;padding: 20px">
<p> Click on Sign up to create your account with angCare: <button (click)="signup()" class="btn btn-primary" [routerLink]="'/signup'">Sign up with {{title}}</button>   : {{status}}</p>
<p> Click on Sign up to create your account with angCare: <button>Sign In </button></p>
</div>

</body>
</html>
<router-outlet></router-outlet>

import { Component,OnInit } from '@angular/core';
import {FormBuilder, FormGroup, Validators } from '@angular/forms';
import { Title } from '@angular/platform-browser';

@Component({
  selector: 'app-root',
```

```
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

export class AppComponent {

  constructor(private formBuilder: FormBuilder) { }
  title = 'angCare';
  status = 'You haven\'t signed up yet';
  name = '';

  ontyping(event:Event) {

    this.name = (<HTMLInputElement>event.target).value;
  }
  signup() {

     this.status = 'Oops! We are working on it!';

  }




}
```

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import {RouterModule, Routes} from '@angular/router'
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ReactiveFormsModule} from '@angular/forms'
import { SignupComponent } from './signup/signup.component';
import { SigninComponent } from './signin/signin.component';


const routes: Routes = [{

  path:'',
  component:AppComponent
},

{
  path:'signup',
  component:SignupComponent
},
{
  path:'sigin',
  component:SigninComponent
}

]

@NgModule({
```

```
  declarations: [
    AppComponent,
    SignupComponent,
    SigninComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    ReactiveFormsModule,
    RouterModule.forRoot(routes)
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

```
<p>
  signup works!
</p>

<div class="jumbotron">
  <div class="container">
      <div class="row">
          <div class="col-md-6 offset-md-3">
              <h2>Angular 6 Reactive Form Validation</h2>
              <form [formGroup]="registerForm" (ngSubmit)="onSubmit()">
```

```
            <div class="form-group">
                <label>First Name</label>
                <input type="text" formControlName="firstName" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.firstName.errors }" />
                <div *ngIf="submitted && f.firstName.errors" class="invalid-feedback">
                    <div *ngIf="f.firstName.errors.required">First Name is required</div>
                </div>
            </div>

            <div class="form-group">
                <label>Last Name</label>
                <input type="text" formControlName="lastName" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.lastName.errors }" />
                <div *ngIf="submitted && f.lastName.errors" class="invalid-feedback">
                    <div *ngIf="f.lastName.errors.required">Last Name is required</div>
                </div>
            </div>
            <div class="form-group">
                <label>Email</label>
                <input type="text" formControlName="email" class="form-control" [ngClass]="{ 'is-invalid': submitted && f.email.errors }" />
                <div *ngIf="submitted && f.email.errors" class="invalid-feedback">
                    <div *ngIf="f.email.errors.required">Email is required</div>
                    <div *ngIf="f.email.errors.email">Email must be a valid email address</div>
                </div>
```

```
            </div>
            <div class="form-group">
                <label>Password</label>
                <input type="password" formControlName="password"
class="form-control" [ngClass]="{ 'is-invalid': submitted && f.password.errors }"
/>
                <div *ngIf="submitted && f.password.errors" class="invalid-
feedback">
                    <div *ngIf="f.password.errors.required">Password is
required</div>
                    <div *ngIf="f.password.errors.minlength">Password must be at
least 6 characters</div>
                </div>
            </div>
            <div class="form-group">
                <button [disabled]="loading" class="btn btn-
primary">Register</button>
            </div>
        </form>
    </div>
  </div>
</div>

<router-outlet></router-outlet>
```

```
import { Component, OnInit } from '@angular/core';
import {FormBuilder, FormGroup, Validators } from '@angular/forms';
@Component({
```

```
  selector: 'app-signup',
  templateUrl: './signup.component.html',
  styleUrls: ['./signup.component.css']
})
export class SignupComponent implements OnInit {
  submitted = false;
  registerForm: FormGroup;
  constructor(private formBuilder: FormBuilder) { }
  ngOnInit() {
    this.registerForm = this.formBuilder.group({
        firstName: ['', Validators.required],
        lastName: ['', Validators.required],
        email: ['', [Validators.required, Validators.email]],
        password: ['', [Validators.required, Validators.minLength(6)]]
    });
}
get f() { return this.registerForm.controls; }
onSubmit() {
  this.submitted = true;

  // stop here if form is invalid
  if (this.registerForm.invalid) {
      return;
  }

  alert('Your request has been submitted for approval')
}
```

```
}
```