

Table of Contents

STEP 1: Software download and installation	2
STEP 2: Create angular project.....	3
STEP 3: Understand angular syntax:.....	7
STEP 4: Add components.....	8
STEP 5: Prepare model and data	11
STEP 6: Structural Directives [*ngFor and *ngIf]	12
STEP 7: @Input Decorator	13

STEP 1: Software download and installation

1. Download and install nodejs for your OS:
<https://nodejs.org/en/download/>
2. Check if node installed by opening a command prompt and type in :

node -v

npm -v

```
(base) Manishs-MacBook-Pro:~ Shalini$ node -v
v17.5.0
(base) Manishs-MacBook-Pro:~ Shalini$ npm -v
8.6.0
```

3. Download and install VSCode for your OS :
<https://code.visualstudio.com/download>
4. Create a folder angularws on your desktop
5. To install latest angular-cli globally on your system execute below command:
npm i -g @angular/cli
6. To install specific angular-cli version locally for your project, execute below command from within the angularws folder created in step 4:

npm i @angular/cli@<version number>

You can get version list from below link

<https://www.npmjs.com/package/@angular/cli?activeTab=versions>

STEP 2: Create angular project

1. Open VSCode and click on File -> Open Folder -> Choose angularws and click Open
2. To create angular project execute below command from within the **angularws** folder

ng new angulardemo

See the screenshot for your reference to select the options on the integrated terminal

```
(base) Manishs-MacBook-Pro:angularws Shalini$ ng new angulardemo
Node.js version v17.5.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For more information, see
https://nodejs.org/en/about/releases/.
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE angulardemo/README.md (1065 bytes)
CREATE angulardemo/.editorconfig (274 bytes)
CREATE angulardemo/.gitignore (548 bytes)
CREATE angulardemo/angular.json (2947 bytes)
CREATE angulardemo/package.json (1042 bytes)
CREATE angulardemo/tsconfig.json (863 bytes)
CREATE angulardemo/.browserslistrc (600 bytes)
CREATE angulardemo/karma.conf.js (1428 bytes)
CREATE angulardemo/tsconfig.app.json (287 bytes)
CREATE angulardemo/tsconfig.spec.json (333 bytes)
CREATE angulardemo/.vscode/extensions.json (130 bytes)
CREATE angulardemo/.vscode/launch.json (474 bytes)
CREATE angulardemo/.vscode/tasks.json (938 bytes)
CREATE angulardemo/src/favicon.ico (948 bytes)
CREATE angulardemo/src/index.html (297 bytes)
CREATE angulardemo/src/main.ts (372 bytes)
CREATE angulardemo/src/polyfills.ts (2338 bytes)
CREATE angulardemo/src/styles.css (80 bytes)
CREATE angulardemo/src/test.ts (749 bytes)
CREATE angulardemo/src/assets/.gitkeep (0 bytes)
CREATE angulardemo/src/environments/environment.prod.ts (51 bytes)
CREATE angulardemo/src/environments/environment.ts (658 bytes)
CREATE angulardemo/src/app/app-routing.module.ts (245 bytes)
CREATE angulardemo/src/app/app.module.ts (393 bytes)
CREATE angulardemo/src/app/app.component.css (0 bytes)
CREATE angulardemo/src/app/app.component.html (23115 bytes)
CREATE angulardemo/src/app/app.component.spec.ts (1088 bytes)
CREATE angulardemo/src/app/app.component.ts (215 bytes)
⌚ Installing packages (npm)...
```

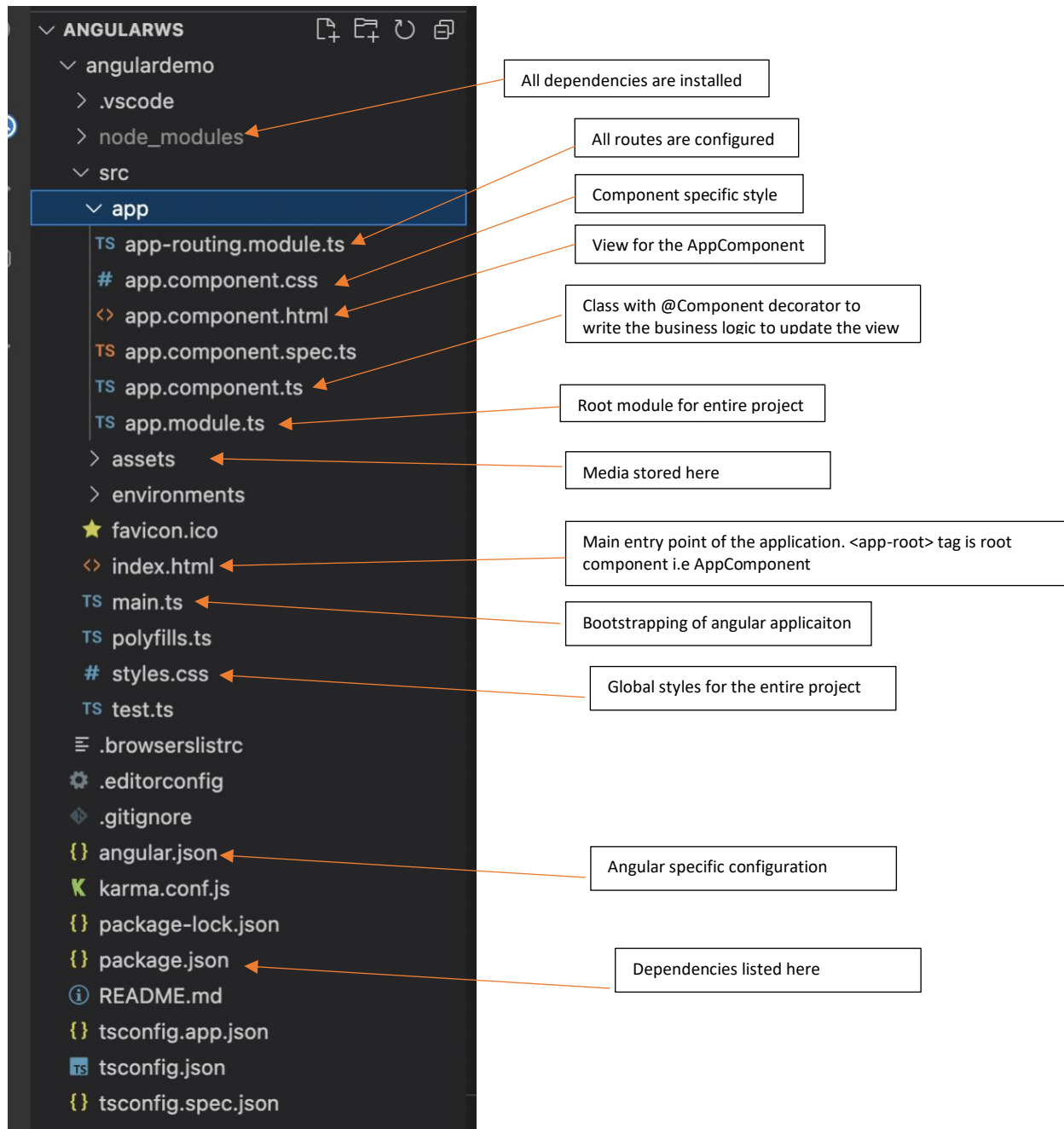
Say Yes for routing

Just hit enter for the default CSS styles

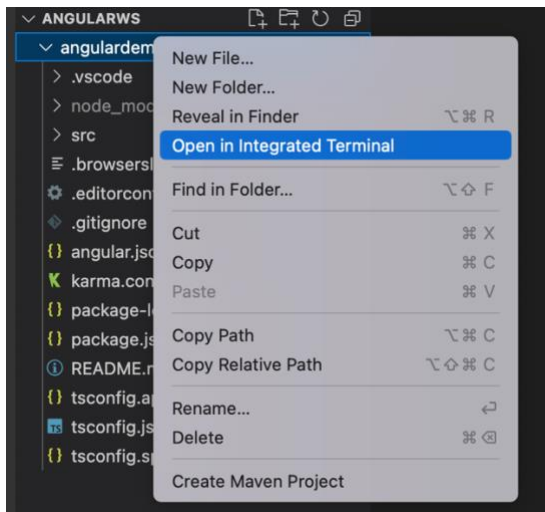
3. After successful installation of packages , terminal should show below output:

```
CREATE angulardemo/src/app/app.component.ts (215 bytes)
✓ Packages installed successfully.
  Directory is already under version control. Skipping initialization of git.
  You have new mail in /var/mail/Shalini
(base) Manishs-MacBook-Pro:angularws Shalini$
```

4. Once the project is created, folder should look as follows :

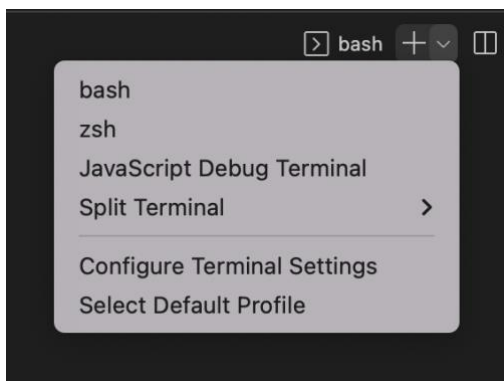


5. To start the server and build the angular application open terminal in VsCode. On the VScode, highlight project, right click and select Open in Integrated Terminal or command prompt as shown below



6. The terminal will be opened in the VsCode.

[WINDOWS USERS PLEASE SELECT CMD FROM THE DROPDOWN OF TERMINAL ON THE RIGHT AS SHOWN BELOW]. INSTEAD OF zsh, windows will have command prompt



7. Once on the terminal and the path should be angulardemo [your project folder], execute below command

```

ng serve

```

(base) Manishs-MacBook-Pro:angulardemo Shalini\$ ng serve

Node.js version v17.5.0 detected.
 Odd numbered Node.js versions will not enter LTS status and should not be used for production. For more information, please see <https://nodejs.org/en/about/releases/>.
 ? Would you like to share pseudonymous usage data about this project with the Angular Team at Google under Google's Privacy Policy at <https://policies.google.com/privacy>. For more details and how to change this setting, see <https://angular.io/analytics>. No
 Global setting: enabled
 Local setting: disabled
 Effective status: disabled
 ✓ Browser application bundle generation complete.

Initial Chunk Files	Names	Raw Size
vendor.js	vendor	2.10 MB
polyfills.js	polyfills	318.01 kB
styles.css, styles.js	styles	210.09 kB
main.js	main	49.85 kB
runtime.js	runtime	6.52 kB
Initial Total		2.67 MB

Build at: 2022-09-11T13:05:26.442Z - Hash: 1d7c0f8ed9b74715 - Time: 4643ms

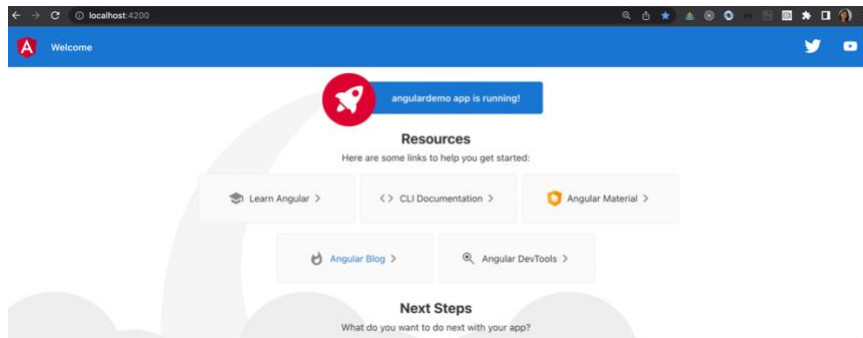
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.

Make sure path is set till angulardemo

Angular application is running on this url

8. Open browser of your choice and type in <http://localhost:4200> in the address bar. Output should be as seen below:



9. Add bootstrap css and js libraries in index.html file

```
<link  
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"  
rel="stylesheet" integrity="sha384-  
EVSTQN3/azprG1Anm3QDgplJlIm9Nao0Yz1ztcQTWfSpd3yD65VohhpUuCOmLASjC"  
crossorigin="anonymous">
```

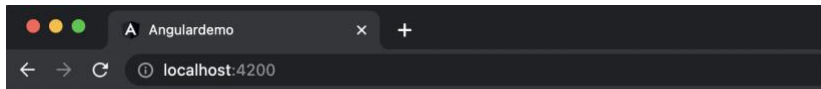
```
<script  
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"  
integrity="sha384-  
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
crossorigin="anonymous"></script>
```

STEP 3: Understand angular syntax:

1. Delete everything in app.component.html file and add the following:

```
<h1>{{title}}</h1>
```

Should see following output:



angulardemo

2. Create a property message in app.component.ts as follows:

```
message:string = 'Some Message'
```

3. Bind this property in app.component.html as follows:

```
<p innerText={{message}}></p>
```

```
<p [innerText]=message></p>
```

Should see output as follows:



angulardemo

Some Message

Some Message

4. 1-way data binding:

- a. String interpolation : {{}} will consider all types as string only

- b. Square bracket syntax : [] retains the type of a variable

```
<button [disabled]="isEnabled"
```

```
(click)="clicked()"
```

```
class="btn btn-primary">Click Me</button>
```

Add following in app.component.ts

```
isEnabled:boolean = false;
```

```
clicked(){ }
```

5. Event binding: ()

6. 2-way data binding

- a. Add FormsModule in imports[] of App.module.ts file. **[DO NOT MISS]**

- b. Use [(ngModel)] i.e banana in a box syntax on the form elements

```
<input type="checkbox" [(ngModel)]=" isEnabled "/>
```

STEP 4: Add components

1. Create Header, Footer and Employee component using the below command from within the angulardemo folder:

ng g c header

ng g c footer

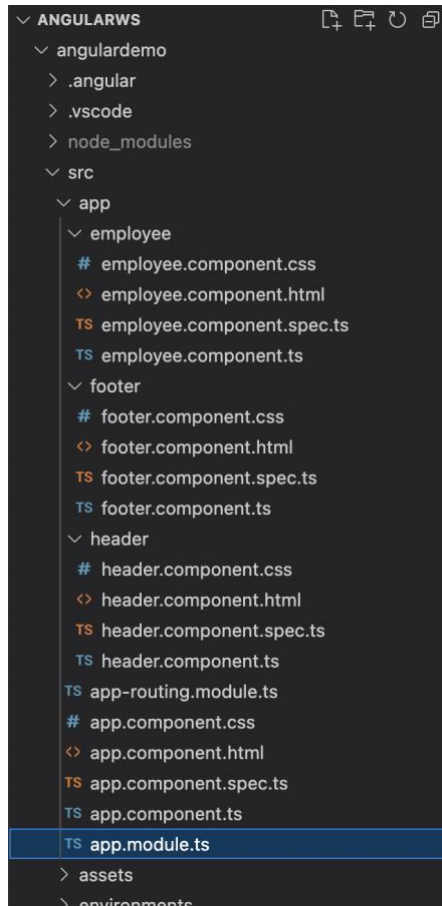
ng g c employee

```
(base) Manishs-MacBook-Pro:angulardemo Shalini$ ng g c header
Node.js version v17.5.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For
more information, please see https://nodejs.org/en/about/releases/.
CREATE src/app/header/header.component.css (0 bytes)
CREATE src/app/header/header.component.html (21 bytes)
CREATE src/app/header/header.component.spec.ts (599 bytes)
CREATE src/app/header/header.component.ts (275 bytes)
UPDATE src/app/app.module.ts (534 bytes)
(base) Manishs-MacBook-Pro:angulardemo Shalini$ ng g c footer
Node.js version v17.5.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For
more information, please see https://nodejs.org/en/about/releases/.
CREATE src/app/footer/footer.component.css (0 bytes)
CREATE src/app/footer/footer.component.html (21 bytes)
CREATE src/app/footer/footer.component.spec.ts (599 bytes)
CREATE src/app/footer/footer.component.ts (275 bytes)
UPDATE src/app/app.module.ts (694 bytes)
(base) Manishs-MacBook-Pro:angulardemo Shalini$ ng g c employee
Node.js version v17.5.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For
more information, please see https://nodejs.org/en/about/releases/.
CREATE src/app/employee/employee.component.css (0 bytes)
CREATE src/app/employee/employee.component.html (23 bytes)
CREATE src/app/employee/employee.component.spec.ts (613 bytes)
CREATE src/app/employee/employee.component.ts (283 bytes)
UPDATE src/app/app.module.ts (706 bytes)
You have new mail in /var/mail/Shalini
```

Apart from the folders created, take a look at declarations[] of app.module.ts file declarations: [

```
    AppComponent,
    HeaderComponent,
    FooterComponent,
    EmployeeComponent
]
```

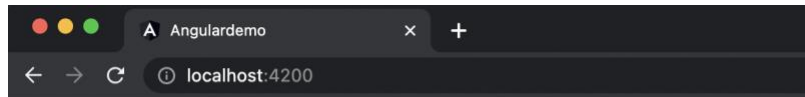

2. The folder structure looks as follows:



3. To add the components in the page and view the corresponding html, copy the selector value from the respective component.ts file and paste it as a tag in the app.component.html file

```
<app-header></app-header>
<app-employee></app-employee>
<app-footer></app-footer>
```

Should get following output with html content of respective component displayed on the page:



angulardemo

Some Message

Some Message

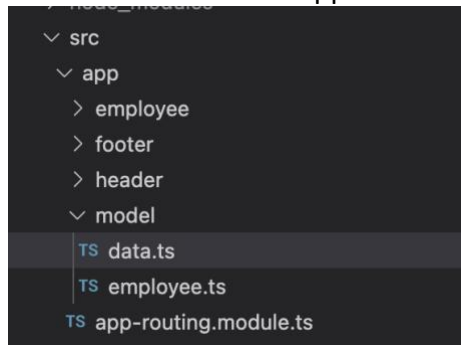
header works!

employee works!

footer works!

STEP 5: Prepare model and data

1. Create a folder within app folder named model

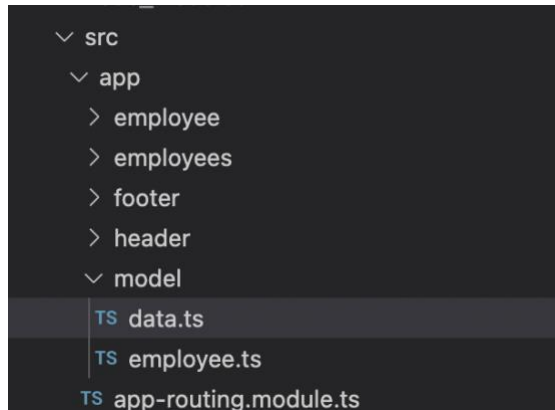


2. Create 2 files within model folder as following:

- a. employee.ts => that exports an interface Employee
export interface Employee{
 eid:number,
 ename:string,
 email:string,
 phone:string,
 password:string,
 address:{
 city?:string,
 country:string,
 zipcode?:number
 }
}
- b. data.ts => that exports array of employees
export let employees = [{eid:1,ename:'shalini',password:'shalini'
 email:'shalini@gmail.com',phone:'1321312312'
 , address:{country:'India'}},
 {eid:2,ename:'Reshma',
 email:'reshma@gmail.com',phone:'987659876',password:'shalini'1
 , address:{country:'India'}},
 {eid:3,ename:'Rahul',
 email:'rahul@gmail.com',phone:'645678678',password:'rahul'
 , address:{country:'India'}}]

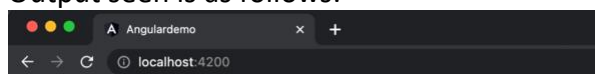
STEP 6: Structural Directives [*ngFor and *ngIf]

1. Create Employees component using the below command from within the angulardemo folder:
ng g c employees
2. Folder structure looks as follows:



3. Import employees[] created before in employees.component.ts and display on employees.component.html using *ngFor.

```
<div *ngIf="employees.length > 0">
  <div *ngFor="let emp of employees">
    <p>Name: {{emp.ename}}</p>
    <p>Name: {{emp.email}}</p>
  </div>
</div>
<div class="container" *ngIf="employees.length <= 0">
  <h1>No employees yet joined your company!!!</h1>
</div>
```
4. Add <app-employees> tag within the app.component.html file
5. Output seen is as follows:



angulardemo

Some Message

Some Message

header works!

Name: shalini

Name: shalini@gmail.com

Name: Reshma

Name: reshma@gmail.com

Name: Rahul

Name: rahul@gmail.com

footer works!

STEP 7: @Input Decorator

1. Instead of displaying the values in EmployeesComponent, pass the employee object from EmployeesComponent to EmployeeComponent using @Input decorator and [] data binding syntax

```
<div *ngIf="employees.length > 0">
  <app-employee [employee]="emp">
  </app-employee>
</div>
</div>
<div class="container" *ngIf="employees.length <= 0">
  <h1>No employees yet joined your company!!!</h1>
</div>
```

2. Add instance variable in employee.component.ts whose name should be same as provided in the [] brackets above as highlighted

```
@Input()
employee:Employee
constructor() {
  this.employee = {eid:1,ename:"",password:"",
  email:"",phone:""
  , address:{country:""}}
}
```

3. Update employee.component.html as follows:

```
<div class="employee">
  <p>Id: {{employee.eid}}</p>
  <p>Name: {{employee.ename}}</p>
  <p>Email: {{employee.email}}</p>
  <p>Phone: {{employee.phone}}</p>
</div>
```

4. Add styles to employee data in employee.component.css file

```
.employee{
  background-color: aliceblue;
  margin-bottom: 10px;
  padding: 10px;
  border: 1px solid;
}
```

5. Output seen is as below:

