

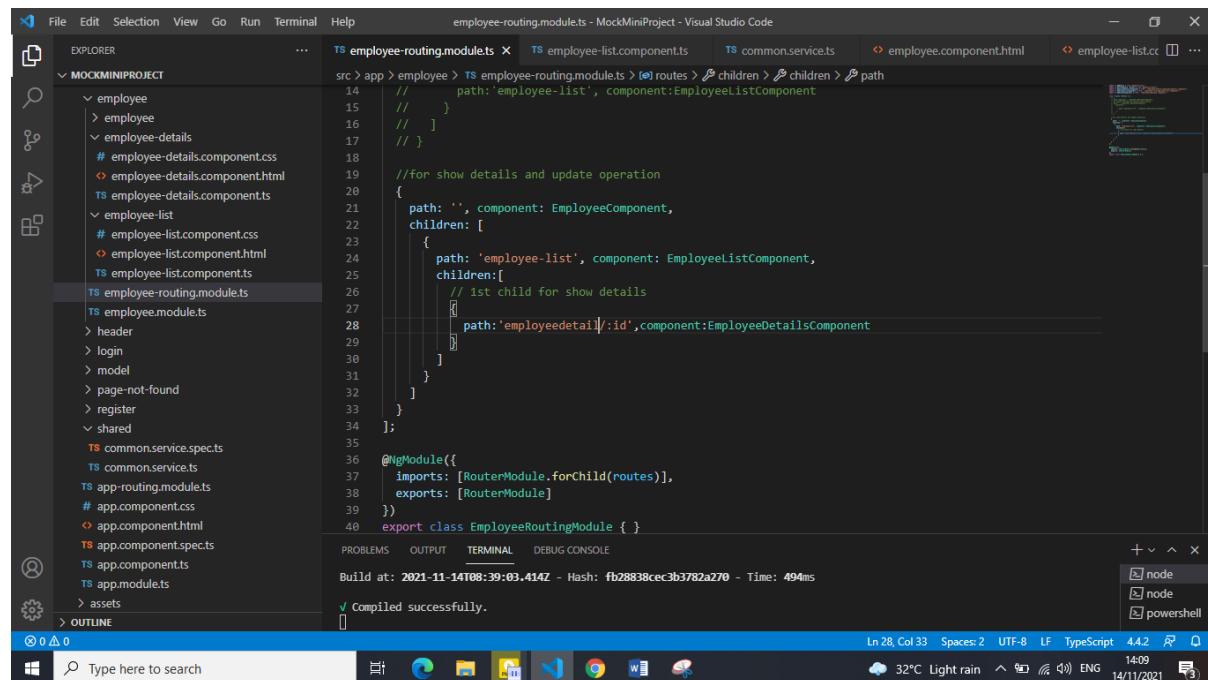
Project Continue....

Show Details:

Step 1: Create one component with name employee-details

```
PS G:\CJC\Angular Learning\AngularWorkspace\MockMiniProject> ng g c employee/employee-details --skip-tests
CREATE src/app/employee/employee-details/employee-details.component.html (31 bytes)
CREATE src/app/employee/employee-details/employee-details.component.ts (314 bytes)
CREATE src/app/employee/employee-details/employee-details.component.css (0 bytes)
UPDATE src/app/employee/employee.module.ts (646 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\MockMiniProject> []
```

Step 2: In employee routing add child routing of employee-list component.



```
TS employee-routing.module.ts - MockMiniProject - Visual Studio Code
src > app > employee > TS employee-routing.module.ts > routes > children > children > path
14 //      path:'employee-list', component:EmployeeListComponent
15 //    }
16 //  ]
17 // }

18 //for show details and update operation
19 {
20   path: '', component: EmployeeComponent,
21   children: [
22     {
23       path: 'employee-list', component: EmployeeListComponent,
24       children:[
25         {
26           path: 'employeeDetail/:id', component:EmployeeDetailsComponent
27         }
28       ]
29     }
30   ]
31 }
32 ]
33 }
34 ];
35
36 @NgModule({
37   imports: [RouterModule.forChild(routes)],
38   exports: [RouterModule]
39 })
40 export class EmployeeRoutingModule { }
```

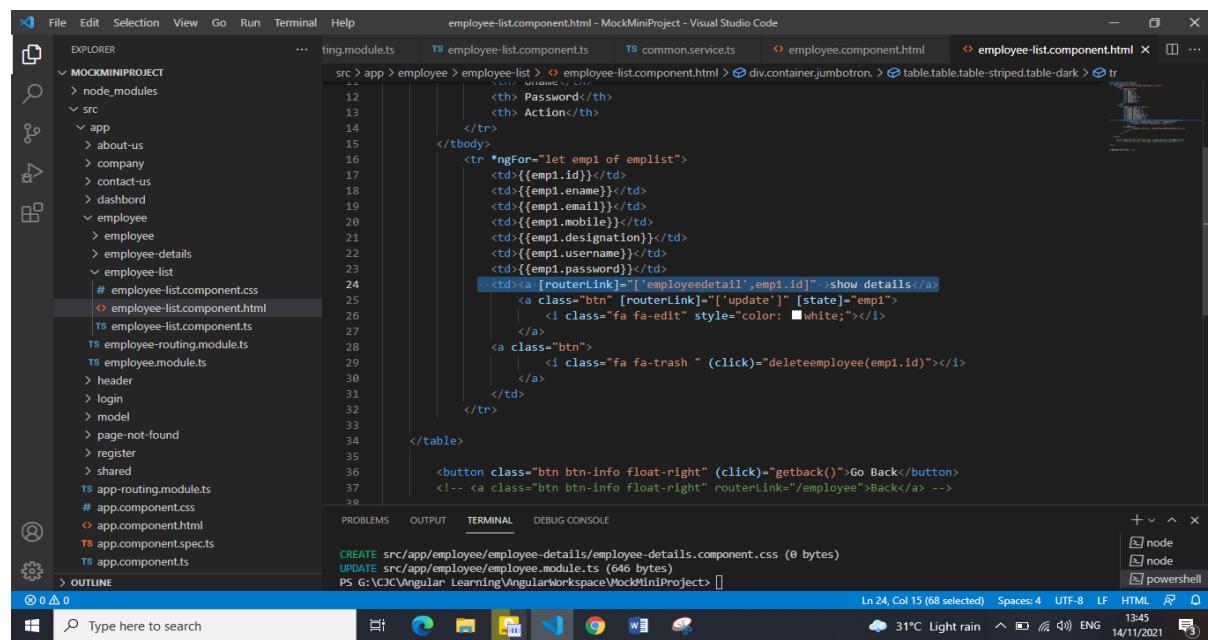
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Build at: 2021-11-14T08:39:03.414Z - Hash: fb28838cec3b3782a270 - Time: 494ms

Compiled successfully.

Ln 28, Col 33 Spaces: 2 UTF-8 LF TypeScript 4.4.2 ⚡ 14:09 32°C Light rain ENG 14/11/2021

Step 3: We have to create a link for show details in employee list html file



```
employee-list.component.html - MockMiniProject - Visual Studio Code
src > app > employee > employee-list > employee-list.component.html > div.containerjumbotron > table.table.table-striped.table-dark > tr
11 <th> Employee ID</th>
12 <th> Name</th>
13 <th> Email</th>
14 <th> Mobile</th>
15 <th> Designation</th>
16 <th> Username</th>
17 <th> Password</th>
18 <th> Action</th>
19 </tr>
20 </thead>
21 <tbody>
22 <tr *ngFor="let emp1 of emplist">
23   <td>{{emp1.id}}</td>
24   <td>{{emp1.ename}}</td>
25   <td>{{emp1.email}}</td>
26   <td>{{emp1.mobile}}</td>
27   <td>{{emp1.designation}}</td>
28   <td>{{emp1.username}}</td>
29   <td>{{emp1.password}}</td>
30   <td><a href="#" [routerLink]="'[employeeDetail]'>show details</a>
31     <a class="btn" [routerLink]="[ 'update' ] [&state]="emp1">
32       <i class="fa fa-edit" style="color: white;"></i>
33     </a>
34     <a class="btn" >
35       <i class="fa fa-trash" (click)="deleteemployee(emp1.id)"></i>
36     </a>
37   </td>
38 </tr>
39 </tbody>
40 <button class="btn btn-info float-right" (click)="getback()">Go Back</button>
41 <!-- <a class="btn btn-info float-right" routerLink="/employee">Back</a> -->
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

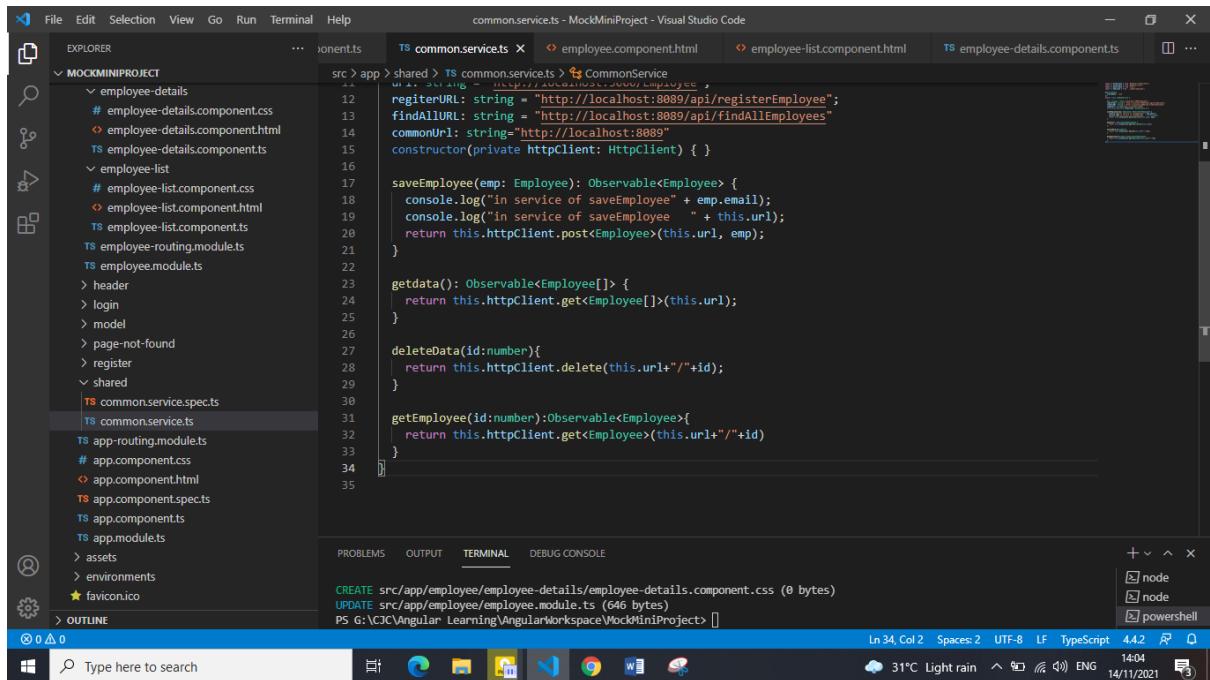
CREATE src/app/employee/employee-details/employee-details.component.css (0 bytes)
UPDATE src/app/employee/employee.module.ts (646 bytes)

PS G:\CJC\Angular Learning\AngularWorkspace\MockMiniProject> []

Ln 24, Col 15 (68 selected) Spaces: 4 UTF-8 LF HTML ⚡ 13:45 31°C Light rain ENG 14/11/2021

Step 4: we have to fetch the data when we click on the link, for that,

- First create one method `getEmployee()` in common service to fetch particular id data from `db.json`



```
src > app > shared > common.service.ts > CommonService
registerURL: string = "http://localhost:8089/api/registerEmployee";
findAllURL: string = "http://localhost:8089/api/findAllEmployees";
commonURL: string = "http://localhost:8089";
constructor(private httpClient: HttpClient) { }

saveEmployee(emp: Employee): Observable<Employee> {
  console.log("in service of saveEmployee" + emp.email);
  console.log("in service of saveEmployee " + this.url);
  return this.httpClient.post<Employee>(this.url, emp);
}

getData(): Observable<Employee[]> {
  return this.httpClient.get<Employee[]>(this.url);
}

deleteData(id:number){
  return this.httpClient.delete(this.url+"/"+id);
}

getEmployee(id:number):Observable<Employee>{
  return this.httpClient.get<Employee>(this.url+"/"+id)
}
```

- Write employee-details html file

```
<p>employee-details works!</p>
<div class="jumbotron" style="text-align:center" >
  <table class="table table-dark center" style="text-align:center" >
    <tr>
      <td> Name</td>
      <td> <input type="text" value="{{employeeObj.ename}}></td>
    </tr>
    <tr>
      <td> Email</td>
      <td> <input type="text" value="{{employeeObj.email}}></td>
    </tr>
    <tr>
      <td> Mobile</td>
      <td> <input type="text" value="{{employeeObj.mobile}}></td>
    </tr>
    <tr>
      <td> Designation</td>
      <td> <input type="text" value="{{employeeObj.designation}}></td>
    </tr>
    <tr>
      <td> Username</td>
      <td> <input type="text" value="{{employeeObj.username}}></td>
    </tr>
    <tr>
      <td> password</td>
      <td> <input type="text" value="{{employeeObj.password}}></td>
    </tr>
  </table>
  <button class="btn btn-info float-right" (click)="getback()">Go Back</button>
</div>
```

```

<tr>
    <td> Name</td>
    <td> <input type="text" value="{{employeeObj.ename}}></td>
</tr>
<tr>
    <td> Email</td>
    <td> <input type="text" value="{{employeeObj.email}}></td>
</tr>
<tr>
    <td> Mobile</td>
    <td> <input type="text" value="{{employeeObj.mobile}}></td>
</tr>
<tr>
    <td> Designation</td>
    <td> <input type="text" value="{{employeeObj.designation}}></td>
</tr>
<tr>
    <td> Username</td>
    <td> <input type="text" value="{{employeeObj.username}}></td>
</tr>
<tr>
    <td> password</td>
    <td> <input type="text" value="{{employeeObj.password}}></td>
</tr>
</table>
<button class="btn btn-info float-right" (click)="getback()">Go Back</button>

```

➤ Write code to fetch the data in employee-details ts file

```

@Directive({
  selector: 'app-employee-details',
  templateUrl: './employee-details.component.html',
  styleUrls: ['./employee-details.component.css']
})
export class EmployeeDetailsComponent implements OnInit {
  employeeObj : Employee;
  constructor(private routes:ActivatedRoute, private common: CommonService,private locations: Location) {}

  ngOnInit(): void {
    // first way is snapshot
    let empId=parseInt(this.routes.snapshot.paramMap.get('id'));
    this.common.getEmployee(empId).subscribe(data=>{
      console.log("Show Employee "+data);
      this.employeeObj=data;
    })
  }
  getback() {
    this.locations.back();
  }
}

```

Another way is

```

@Directive({
  selector: './employee-details',
  styleUrls: ['./employee-details.component.css']
})
export class EmployeeDetailsComponent implements OnInit {
  employeeObj : Employee;
  constructor(private routes:ActivatedRoute, private common: CommonService,private locations: Location) {}

  ngOnInit(): void {
    // Second way is Observable
    this.routes.paramMap.subscribe(param1=>{
      this.common.getEmployee(parseInt(param1.get('id'))).subscribe(data=>{
        this.employeeObj=data;
      })
    })
  }
  getback() {
    this.locations.back();
  }
}

```

Output is:

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232	Engineer	abc	abc	show details
2	ww	ww	12345	ww	ww	ww	show details

[Go Back](#)

Employee-list works..!

Name: abc
Email: abc@gmail.com
Mobile: 23232
Designation: Engineer
Username: abc
password: abc

[Go Back](#)

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232	Engineer	abc	abc	show details
2	ww	ww	12345	ww	ww	ww	show details

[Go Back](#)

Name: ww
Email: ww
Mobile: 12345
Designation: ww
Username: ww
password: ww

[Go Back](#)

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232	Engineer	abc	abc	show details
2	ww	ww	12345	ww	ww	ww	show details

[Go Back](#)

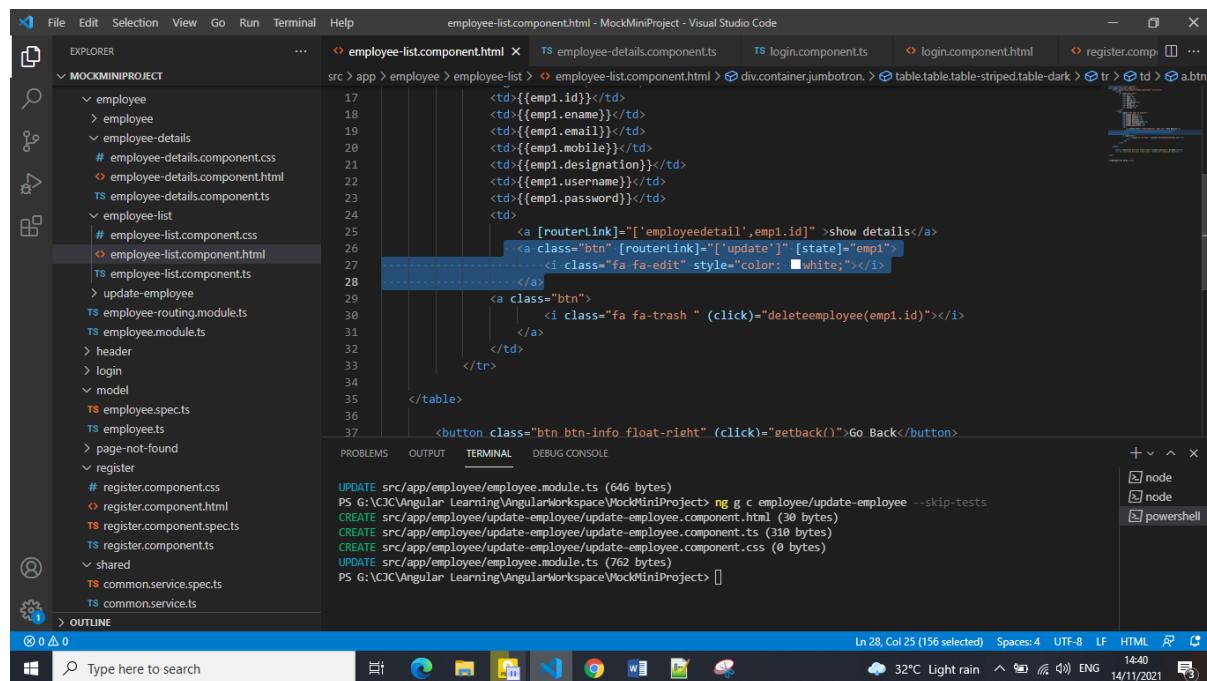


Update Details:

Step 1: create update component

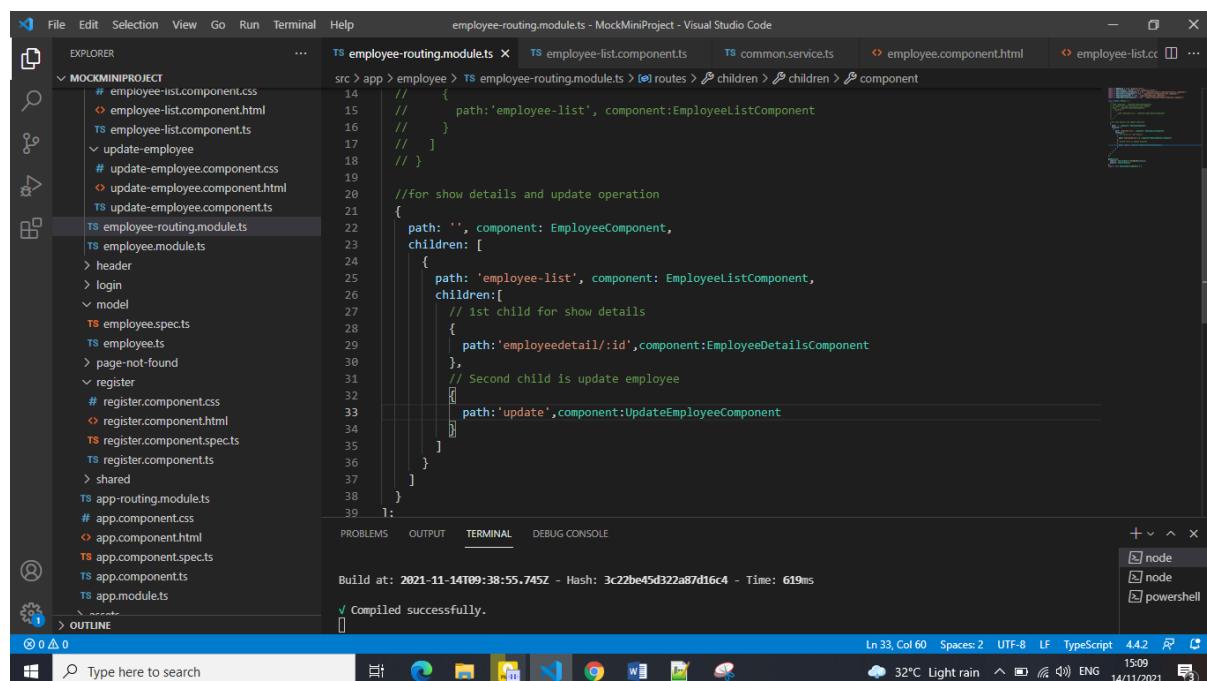
```
PS G:\CJC\Angular Learning\AngularWorkspace\MockMiniProject> ng g c employee/update-employee --skip-tests
CREATE src/app/employee/update-employee/update-employee.component.html (30 bytes)
CREATE src/app/employee/update-employee/update-employee.component.ts (310 bytes)
CREATE src/app/employee/update-employee/update-employee.component.css (0 bytes)
UPDATE src/app/employee/employee.module.ts (762 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\MockMiniProject>
```

Step 2: Go to Employee-list html file and write below routerLink,



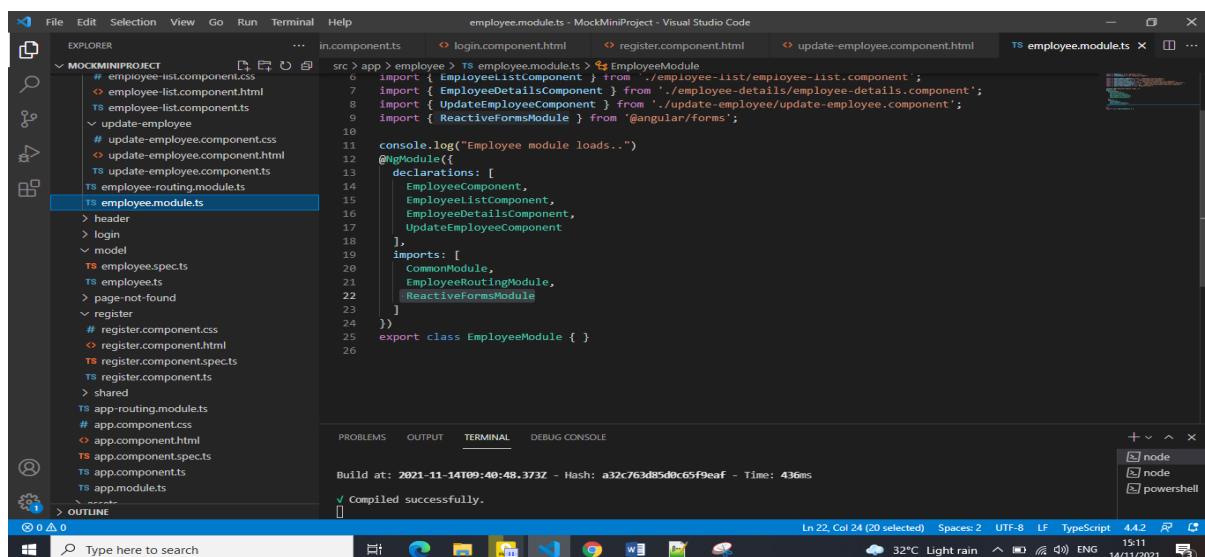
```
<td>{{emp1.id}}</td>
<td>{{emp1.name}}</td>
<td>{{emp1.email}}</td>
<td>{{emp1.mobile}}</td>
<td>{{emp1.designation}}</td>
<td>{{emp1.username}}</td>
<td>{{emp1.password}}</td>
<td>
    <a [routerLink]=['employeeDetail',emp1.id] >show details</a>
    <a class="btn" [routerLink]=["update"] [state]="emp1">
        <i class="fa fa-edit" style="color: white;"></i>
    </a>
    <a class="btn">
        <i class="fa fa-trash" (click)="deleteEmployee(emp1.id)"></i>
    </a>
</td>
</tr>
</table>
<button class="btn btn-info float-right" (click)="getBack()">Go Back</button>
```

Step 3: add child routing in employee-list routing



```
// routes > children > component
path: 'employee-list', component: EmployeeListComponent
// for show details and update operation
path: '', component: EmployeeComponent,
children: [
    {
        path: 'employee-list', component: EmployeeListComponent,
        children:[
            // 1st child for show details
            {
                path:'employeedetail/:id',component:EmployeeDetailsComponent
            },
            // Second child is update employee
            {
                path:'update',component:UpdateEmployeeComponent
            }
        ]
    }
]
```

Step 4: Add ReactiveFormsModule in Employee module



```
File Edit Selection View Go Run Terminal Help
employee.module.ts - MockMiniProject - Visual Studio Code
EXPLORER ... in.component.ts login.component.html register.component.html update-employee.component.html TS employee.module.ts ...
MOCKMINIPROJECT
# employee-list.component.css
employee-list.component.html
TS employee-list.component.ts
update-employee
# update-employee.component.css
update-employee.component.html
TS update-employee.components.ts
TS employee-routing.module.ts
TS employee.module.ts
> header
> login
model
TS employee.specs
TS employees.ts
> page-not-found
register
# register.component.css
register.component.html
TS register.component.specs
TS register.component.ts
shared
TS app-routing.module.ts
# app.component.css
app.component.html
TS app.component.specs
TS app.component.ts
TS app.module.ts
...
OUTLINE
src > app > employee > TS employee.module.ts > EmployeeModule
6 import { EmployeeListComponent } from './employee-list/employee-list.component';
7 import { UpdateEmployeeComponent } from './update-employee/update-employee.component';
8 import { ReactiveFormsModule } from '@angular/forms';
9
10 console.log("Employee module loads...")
11 @NgModule({
12   declarations: [
13     EmployeeComponent,
14     EmployeeListComponent,
15     EmployeeDetailsComponent,
16     UpdateEmployeeComponent
17   ],
18   imports: [
19     CommonModule,
20     EmployeeRoutingModule,
21     ReactiveFormsModule
22   ]
23 })
24 export class EmployeeModule { }
25
26
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Build at: 2021-11-14T09:40:48.373Z - Hash: a32c763d85d0c65f9eaF - Time: 436ms
Compiled successfully.
Ln 22, Col 24 (20 selected) Spaces: 2 UTF-8 LF TypeScript 4.4.2
node node powershell
32°C Light rain 15:11 14/11/2021
```

Step 5: Add code in update HTML file

```
<p>update-employee works!</p>
<form [formGroup]="updateForm" (ngSubmit)="onSubmit()">

<div class="jumbotron" style="text-align:center" >
  <h5 style="text-align: center; color: darkviolet;">! Update Here !</h5>
  <table class="table table-dark center" style="text-align:center" >
    <tr>
      <td> Name</td>
      <td> <input type="text" formControlName="ename" placeholder="enter name"></td>
    </tr>
    <tr>
      <td> Email</td>
      <td> <input type="text" formControlName="email" placeholder="enter email"></td>
    </tr>
    <tr>
      <td> Mobile</td>
      <td> <input type="text" formControlName="mobile" placeholder="enter mbole no"></td>
    </tr>
    <tr>
      <td> Designation</td>
      <td> <input type="text" formControlName="designation" placeholder="enter Designation"></td>
    </tr>
    <tr>
      <td> Username</td>
      <td> <input type="text" formControlName="username" placeholder="enter username"></td>
    </tr>
    <tr>
      <td> password</td>
      <td> <input type="password" formControlName="password" placeholder="enter password"></td>
    </tr>
  </table>
  <button class="btn btn-primary" type="submit">Update</button>&nbsp;&nbsp;
  <button class="btn btn-primary" type="submit" (click)="goBack()"> back </button>
</div>
</form>
```

```
File Edit Selection View Go Run Terminal Help update-employee.component.html - MockMiniProject - Visual Studio Code

EXPLORER ... iis.component.ts T5 login.components.ts login.component.html register.component.html update-employee.component.html

MOCKMINIPROJECT
# employee-list.component.css
employee-list.component.html
TS employee-list.component.ts
update-employee
# update-employee.component.css
update-employee.component.html
TS update-employee.component.ts
TS employee-routing.module.ts
TS employee.module.ts
header
login
model
# employee.spec.ts
TS employee.ts
page-not-found
register
# register.component.css
register.component.html
TS register.component.ts
register.component.ts
shared
# common.service.spec.ts
TS common.service.ts
TS app-routing.module.ts
# app.component.css
app.component.html
TS app.component.ts

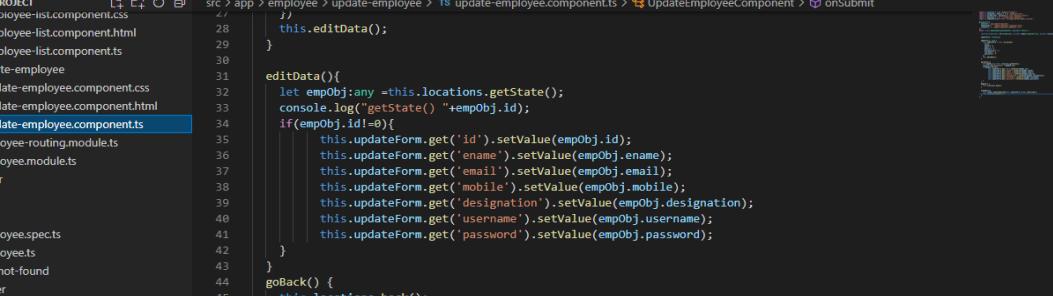
Build at: 2021-11-14T09:51:26.208Z - Hash: 4a537b9d1f04c476e2e - Time: 1054ms
Compiled successfully.

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ^ x

Build at: 2021-11-14T09:51:26.208Z - Hash: 4a537b9d1f04c476e2e - Time: 1054ms
Compiled successfully.

Ln 35, Col 8 Spaces:4 UTF-8 LF HTML RP L
31°C Light rain ENG 14/11/2021
```

Step 6: Write update and edit code in update.ts file



File Edit View Go Run Terminal Help update-employee.component.ts - MockMiniProject - Visual Studio Code

EXPLORER

MOCKMINIPROJECT

- # employee-list.component.css
- employee-list.component.html
- TS employee-list.component.ts
- update-employee

 - # update-employee.component.css
 - update-employee.component.html
 - TS update-employee.component.ts

- TS employee-routing.module.ts
- TS employee.module.ts
- header
- login
- model

 - TS employee.specs.ts
 - TS employee.ts

- page-not-found
- register

 - # register.component.css
 - register.component.html
 - TS register.component.spec.ts
 - TS register.component.ts

- shared

 - TS common.service.spec.ts
 - TS common.services.ts

- app-routing.module.ts
- # app.component.css
- app.component.html
- TS app.component.spec.ts

update-employee.component.ts # update-employee.component.css TS employee-details.component.ts TS login.component.ts

```
src/app/employee/update-employee/TS update-employee.component.ts > UpdateEmployeeComponent > onSubmit
```

```
27     this.editData();
28 }
29
30
31 editData(){
32     let empObj:any =this.locations.getState();
33     console.log("getState() "+empObj.id);
34     if(empObj.id!=0){
35         this.updateForm.get('id').setValue(empObj.id);
36         this.updateForm.get( 'ename').setValue(empObj.ename);
37         this.updateForm.get('email').setValue(empObj.email);
38         this.updateForm.get('mobile').setValue(empObj.mobile);
39         this.updateForm.get('designation').setValue(empObj.designation);
40         this.updateForm.get('username').setValue(empObj.username);
41         this.updateForm.get('password').setValue(empObj.password);
42     }
43 }
44 goBack() {
45     this.locations.back();
46 }
47
48 onSubmit(){}
49     this.common.updateEmployee(this.updateForm.value).subscribe();
50
51 }
52 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Compiled successfully.

Ln 50, Col 27 Spaces: 2 UTF-8 TypeScript 4.4.2 node node

31°C Light rain ENG 14/11/2021

Step 7: Write update method in service

```

src/app/shared/TS_common.service.ts > CommonService > updateEmployee
registerURL: string = "http://localhost:8089/api/registerEmployee";
findAllURL: string = "http://localhost:8089/api/findAllEmployees";
commonUrl: string = "http://localhost:8089";
constructor(private httpClient: HttpClient) { }

saveEmployee(emp: Employee): Observable<Employee> {
  console.log("in service of saveEmployee" + emp.email);
  console.log("in service of saveEmployee " + this.url);
  return this.httpClient.post<Employee>(this.url, emp);
}

getData(): Observable<Employee[]> {
  return this.httpClient.get<Employee[]>(this.url);
}

deleteData(id:number){
  return this.httpClient.delete(this.url+"/"+id);
}

getEmployee(id:number):Observable<Employee>{
  return this.httpClient.get<Employee>(this.url+"/"+id)
}

updateEmployee(emp:Employee):Observable<Employee>{
  return this.httpClient.put<Employee>(this.url+'/'+emp.id,emp)
}
  
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Compiled successfully.

Ln 36, Col 61 Spaces 2 UTF-8 LF TypeScript 4.4.2 15:23 31°C Light rain ENG 14/11/2021

Output:

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232222	Engineer	abc	abc	show details edit delete
2	ww	ww	9899314536	Tester	ww	ww	show details edit delete

Go Back

Employee-list works..!

I Update Here !

Name	<input type="text" value="ww"/>
Email	<input type="text" value="ww"/>
Mobile	<input type="text" value="9899314536"/>
Designation	<input type="text" value="Tester"/>
Username	<input type="text" value="ww"/>
password	<input type="text" value=".."/>

Update back

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232222	Engineer	abc	abc	show details edit delete
2	ww	ww	9899314536	Tester	ww	ww	show details edit delete

Go Back

Changing Mobile no.

The screenshot shows a web browser window with the URL `localhost:3000/Employee`. The page title is "Mini Mock Project". A form titled "I Update Here !" is displayed, containing fields for Name, Email, Mobile, Designation, Username, and Password. The "Mobile" field has the value "7777777777". Below the form is a table with the following data:

ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232222	Engineer	abc	abc	show details edit delete
2	ww	ww	98999314536	Tester	ww	ww	show details edit delete

At the bottom right of the table is a "Go Back" button. The browser's taskbar shows various pinned icons, and the system tray indicates the date as 14/11/2021 and time as 15:25.

Mobile no gets changed..

The screenshot shows a web browser window with the URL `localhost:3000/Employee`. The page title is "Mock Project". The navigation bar includes links for Home, Company, Employee, AboutUs, Contact, and Logout. The main content area displays a table with the following data:

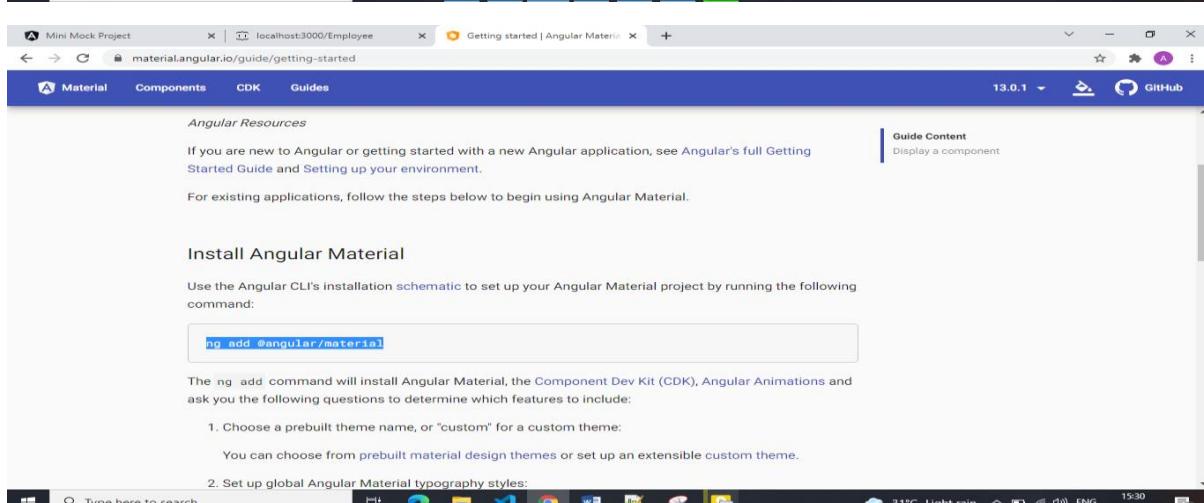
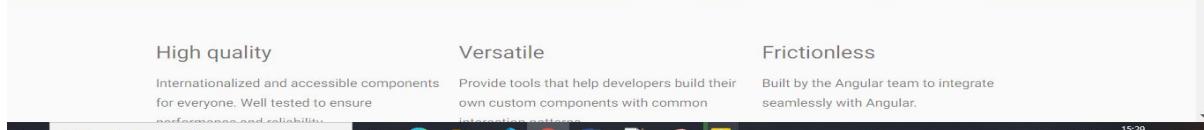
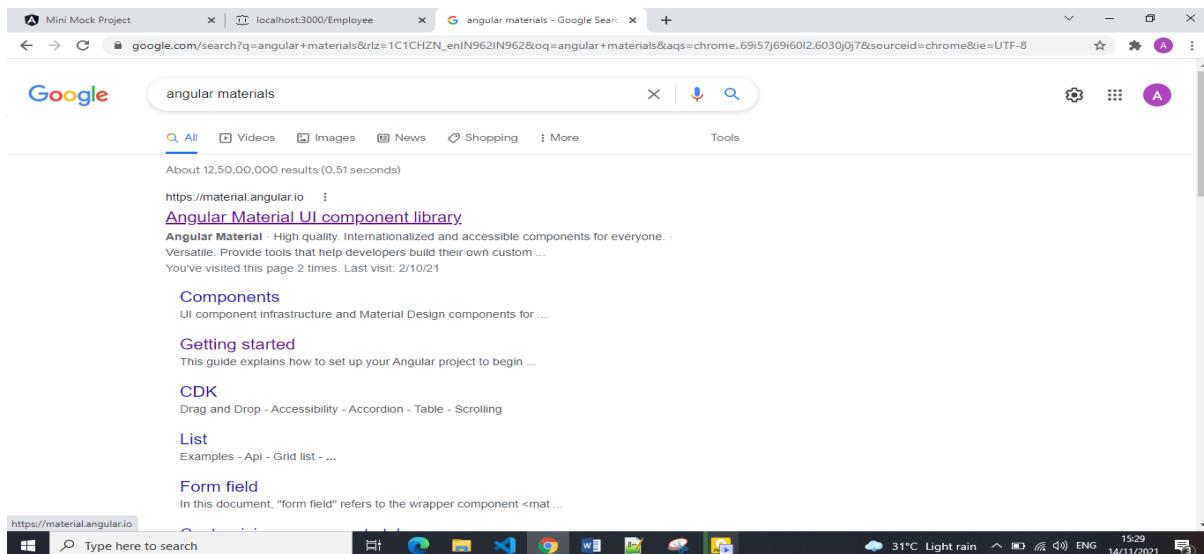
ID	Name	Email	Mobile	Designation	Uname	Password	Action
1	abc	abc@gmail.com	23232222	Engineer	abc	abc	show details edit delete
2	ww	ww	7777777777	Tester	ww	ww	show details edit delete

At the bottom right of the table is a "Go Back" button. The browser's taskbar shows various pinned icons, and the system tray indicates the date as 14/11/2021 and time as 15:35.

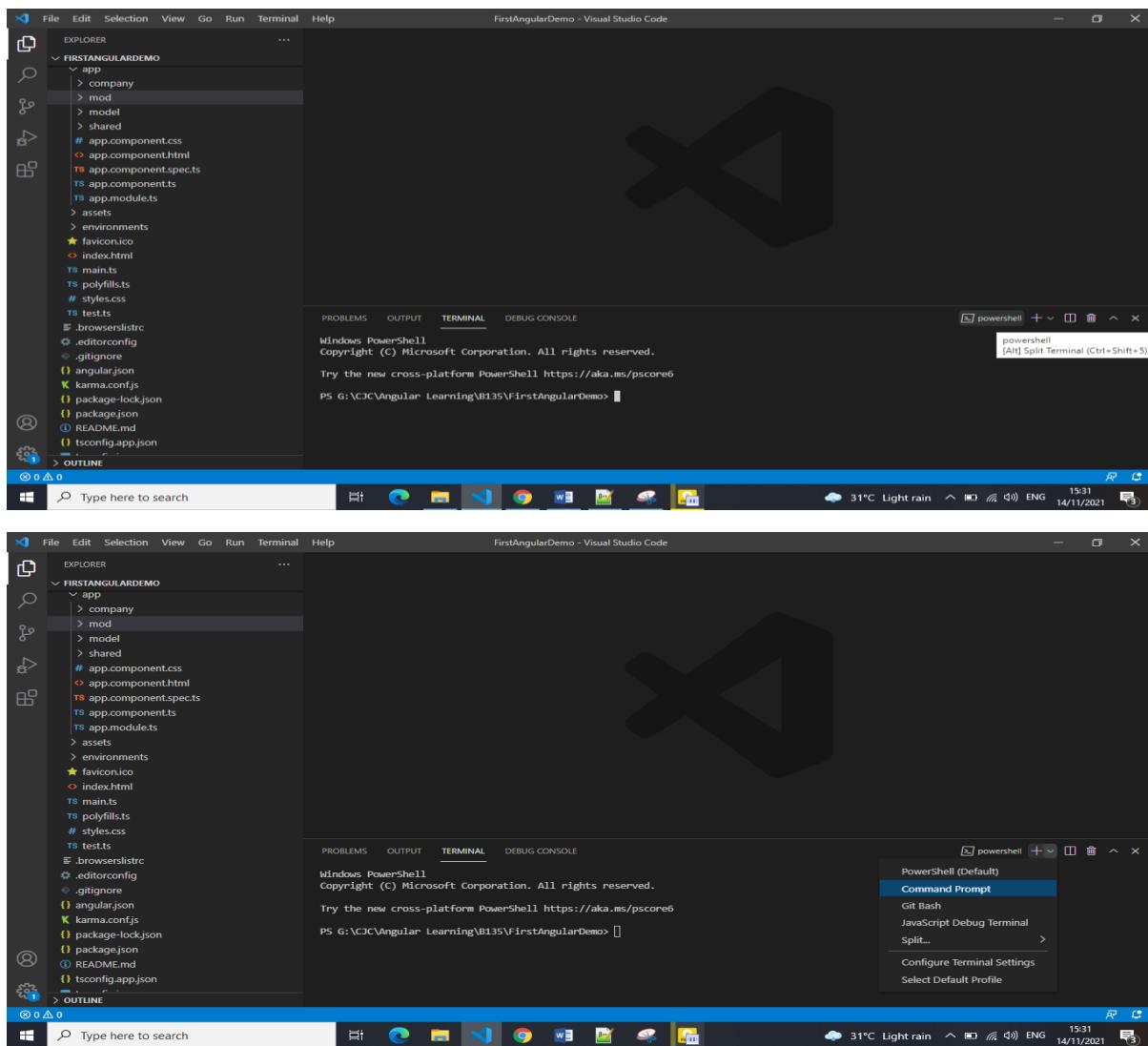
Angular Material

Rather than using third party bootstrap styling, we can also use the angular styling library. This is angular libraries.

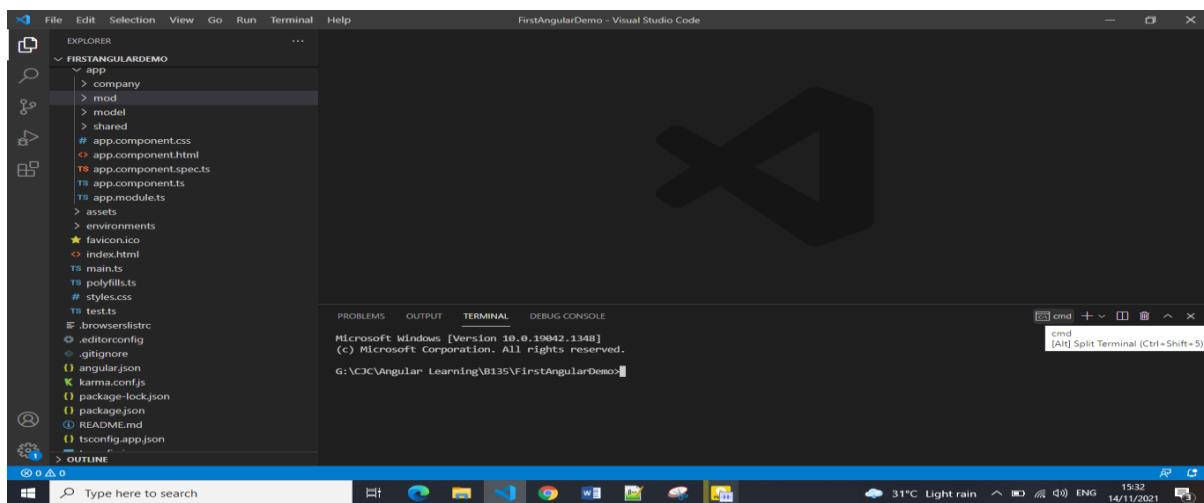
Step 1: Go to angular material official site



Step 2: Go to the terminal, open cmd prompt not the powershell like below



Now its cmd, not powershell



Step 4: Paste the copied command

The screenshot shows the Visual Studio Code interface with the title bar "FirstAngularDemo - Visual Studio Code". The Explorer sidebar on the left shows a project structure with files like app.component.css, app.component.html, and app.module.ts. The Terminal tab is active, displaying the command "G:\CJC\Angular Learning\B135\FirstAngularDemo>ng add @angular/material". The output pane shows the process of installing the package, including finding a compatible version and package information loaded.

This screenshot is similar to the previous one, but the terminal output includes a message: "The package @angular/material@12.2.13 will be installed and executed. Would you like to proceed? Yes".

In this final screenshot, the terminal output shows a theme selection menu from Material Design. Options include "Indigo/Pink", "Deep Purple/Amber", "Pink/Blue Grey", "Purple/Green", and "Custom". The "Indigo/Pink" theme is highlighted with a preview link: "Preview: https://material.angular.io/theme-indigo-pink".

File Edit Selection View Go Run Terminal Help

FIRSTANGULARDEMO

EXPLORER

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

node + ×

Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

G:\CJC\Angular Learning\B135\FirstAngularDemo>ng add @angular/material
i Using package manager: npm
✓ Found compatible package version: @angular/material@12.2.13.
✓ Package information loaded.

The package @angular/material@12.2.13 will be installed and executed.
Would you like to proceed? Yes
✓ Package successfully installed.
? Choose a prebuilt theme name, or "custom" for a custom theme: Indigo/Pink
[Preview: https://material.angular.io?theme=indigo-pink]
? Set up global Angular Material typography styles? Yes
? Set up browser animations for Angular Material? (Y/n)

OUTLINE

Type here to search

31°C Light rain 15:38 14/11/2021

File Edit Selection View Go Run Terminal Help

FIRSTANGULARDEMO

EXPLORER

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

node + ×

Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

G:\CJC\Angular Learning\B135\FirstAngularDemo>ng add @angular/material
i Using package manager: npm
✓ Found compatible package version: @angular/material@12.2.13.
✓ Package information loaded.

The package @angular/material@12.2.13 will be installed and executed.
Would you like to proceed? Yes
✓ Package successfully installed.
? Choose a prebuilt theme name, or "custom" for a custom theme: Indigo/Pink
[Preview: https://material.angular.io?theme=indigo-pink]
? Set up global Angular Material typography styles? Yes
? Set up browser animations for Angular Material? Yes
UPDATE package.json (1148 bytes)
Installing packages (npm)...

OUTLINE

Type here to search

31°C Light rain 15:38 14/11/2021

File Edit Selection View Go Run Terminal Help

FIRSTANGULARDEMO

EXPLORER

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

cmd + ×

Microsoft Windows [Version 10.0.19042.1348]
(c) Microsoft Corporation. All rights reserved.

G:\CJC\Angular Learning\B135\FirstAngularDemo>ng add @angular/material
i Using package manager: npm
✓ Found compatible package version: @angular/material@12.2.13.
✓ Package information loaded.

The package @angular/material@12.2.13 will be installed and executed.
Would you like to proceed? Yes
✓ Package successfully installed.
? Choose a prebuilt theme name, or "custom" for a custom theme: Indigo/Pink
[Preview: https://material.angular.io?theme=indigo-pink]
? Set up global Angular Material typography styles? Yes
? Set up browser animations for Angular Material? Yes
UPDATE package.json (1148 bytes)
✓ Packages installed successfully.
UPDATE src/app/app.module.ts (582 bytes)
UPDATE angular.json (3263 bytes)
UPDATE src/index.html (584 bytes)
UPDATE src/styles.css (181 bytes)

OUTLINE

Type here to search

31°C Light rain 15:40 14/11/2021

Installation is completed..

Step 5:

The screenshots illustrate the steps to integrate Angular Material into a project:

- Screenshot 1: Installation Guide**

The first screenshot shows the "Install Angular Material" guide. It provides instructions for running the command `ng add @angular/material`. A note states that this command will install Angular Material, the Component Dev Kit (CDK), Angular Animations and ask you the following questions to determine which features to include. The sidebar includes links for "Guide Content" and "Display a component".
- Screenshot 2: Components Category**

The second screenshot shows the "Components" category page. It lists various UI components: Autocomplete, Badge, Bottom Sheet, Button, Button toggle, Card, Checkbox, Chips, Core, Datepicker, and Dialog. To the right, there are three examples: "Autocomplete" (a dropdown menu), "Badge" (a small value indicator), and "Bottom Sheet" (a large interactive panel). A note says "Angular Material offers a wide variety of UI components based on the [Material Design specification](#)".
- Screenshot 3: Button Examples**

The third screenshot shows the "Button" examples page. It displays various types of buttons: Basic, Raised, Stroked, Flat, Icon, FAB, and Mini FAB. Each type is shown in different color variations: Basic, Primary, Accent, Warn, Disabled, and Link.

We want that Warn button in our project then, click on view code option, copy the warn button code

The screenshot shows the Angular Material documentation page for the 'Button' component. The 'Components' tab is selected. The 'HTML' tab is active, displaying the source code for a button example. The code includes sections for basic buttons, raised buttons (Primary, Accent, Warn), disabled buttons, and links. A right-click context menu is open over the 'Warn' colored button, with options like 'Copy', 'Copy link to highlight', 'Search Google for <button mat-raised-button color="warn">Warn</...>', 'Print...', and 'Inspect'.

```

<section>
  <div class="example-label">Basic</div>
  <div class="example-button-row">
    <button mat-button>Basic</button>
    <button mat-button color="primary">Primary</button>
    <button mat-button color="accent">Accent</button>
    <button mat-button color="warn">Warn</button>
    <button mat-button disabled>Disabled</button>
    <a mat-button href="https://www.google.com/" target="_blank">Link</a>
  </div>
</section>
<mat-divider></mat-divider>
<section>
  <div class="example-label">Raised</div>
  <div class="example-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 href="https://www.google.com/" target="_blank">Link</a>
  </div>
</section>
<mat-divider></mat-divider>
<section>
  <div class="example-label">Stroked</div>
  <div class="example-button-row">
    <button mat-stroked-button>Basic</button>

```

This screenshot is identical to the one above, showing the Angular Material documentation for the 'Button' component. The 'Components' tab is selected, and the 'HTML' tab shows the button example code. A right-click context menu is open over the 'Warn' colored button, providing options like 'Copy', 'Copy link to highlight', 'Search Google for <button mat-raised-button color="warn">Warn</...>', 'Print...', and 'Inspect'.

Paste it in component which you want to insert this button

Lets, paste it in app.html file

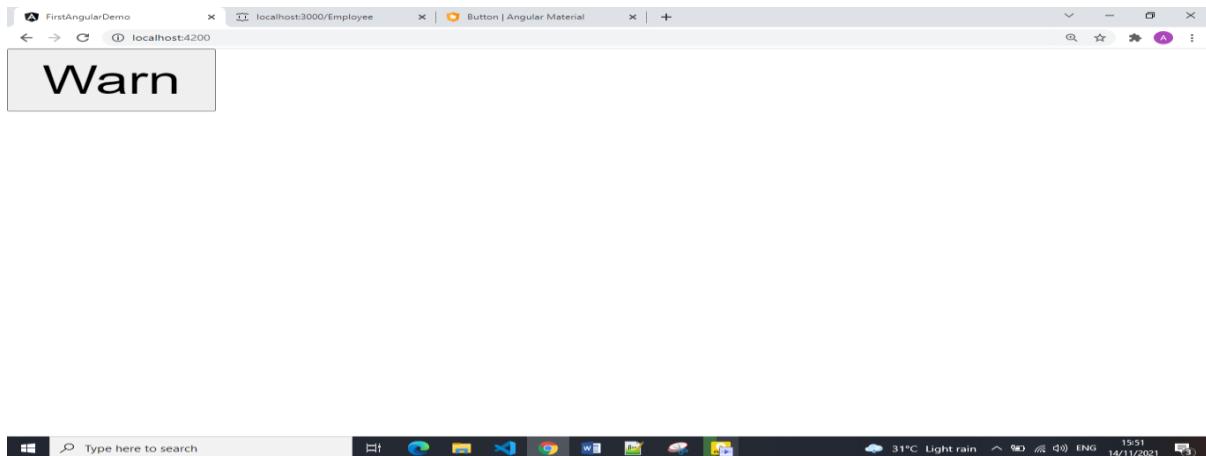
The screenshot shows the Visual Studio Code interface with the 'app.component.html' file open. The code editor contains the following HTML:

```

<button mat-raised-button color="warn">Warn</button>

```

The status bar at the bottom indicates the file is saved and shows the path 'G:\CJAngular Learning\B135\FirstAngularDemo>ng s'. A terminal window is also visible in the background.



But, still we can't use this button, so we have to register this button module in the parent module

Then, go to the browser, open API section on same button example you use.

API reference for Angular Material button

```
import {MatButtonModule} from '@angular/material/button';
```

API reference for Angular Material button

```
import {MatButtonModule} from '@angular/material/button';
```

Paste it in app module

```

File Edit Selection View Go Run Terminal Help
app.module.ts - FirstAngularDemo - Visual Studio Code
EXPLORER ... app.component.html TS app.module.ts ...
src > app > TS app.module.ts ...
src
  > node_modules
  > app
    > company
    > mod
    > model
    > shared
      # app.component.css
      < app.component.html
      TS app.component.spec.ts
      TS app.component.ts
      TS app.module.ts
    > assets
    > environments
    ★ favicon.ico
    < index.html
    TS main.ts
    TS polyfills.ts
    # styles.css
    TS test.ts
    # browserslistrc
    ⚙ .editorconfig
    ⚙ .gitignore
    ⚙ angular.json
    K karma.conf.js
    ⚙ package-lock.json
    ⚙ package.json
  > OUTLINE
  ⚙ 0 0 Initializing JS/TS language features
  Type here to search
  PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
  ✓ Compiled successfully.
  ✓ Browser application bundle generation complete.
  Initial Chunk Files | Names | Size
  main.js | main | 57.35 kB
  4 unchanged chunks
  Build at: 2021-11-14T10:19:34.363Z - Hash: dcc272c32749ffe83041 - Time: 1901ms
  ✓ Compiled successfully.
  Generating browser application bundles...
Ln 8, Col 1 (57 selected) Spaces: 2 UTF-8 LF TypeScript 4.4.2 ⚙ 31°C Light rain ⚙ 90 ⚙ ENG 15:49 14/11/2021

```

And also, Register the module in import section:

```

File Edit Selection View Go Run Terminal Help
app.module.ts - FirstAngularDemo - Visual Studio Code
EXPLORER ... app.component.html TS app.module.ts ...
src > app > TS app.module.ts > AppModule
src
  > node_modules
  > app
    > company
    > mod
    > model
    > shared
      # app.component.css
      < app.component.html
      TS app.component.spec.ts
      TS app.component.ts
      TS app.module.ts
    > assets
    > environments
    ★ favicon.ico
    < index.html
    TS main.ts
    TS polyfills.ts
    # styles.css
    TS test.ts
    # browserslistrc
    ⚙ .editorconfig
    ⚙ .gitignore
    ⚙ angular.json
    K karma.conf.js
    ⚙ package-lock.json
    ⚙ package.json
  > OUTLINE
  ⚙ 0 0
  Type here to search
  PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
  ✓ Compiled successfully.
  ✓ Browser application bundle generation complete.
  Initial Chunk Files | Names | Size
  vendor.js | vendor | 3.00 MB
  main.js | main | 57.86 kB
  3 unchanged chunks
  Build at: 2021-11-14T10:20:24.437Z - Hash: db166d044696ffdb0d7a - Time: 1556ms
  ✓ Compiled successfully.
Ln 18, Col 5 (15 selected) Spaces: 2 UTF-8 LF TypeScript 4.4.2 ⚙ 31°C Light rain ⚙ 90 ⚙ ENG 15:50 14/11/2021

```



To Add Icons:

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure for "FIRSTANGULARDEMO". The "src" folder contains "app", "mod", "model", "shared", and files like "app.component.css", "app.component.html", "app.component.spec.ts", "app.component.ts", "app.module.ts", "index.html", "main.ts", "polyfills.ts", "styles.css", "tests.ts", ".browserslistrc", ".editorconfig", ".gitignore", "angular.json", "karma.conf.js", "package-lock.json", and "package.json".
- Editor (Center):** Displays the content of "app.component.html". The code includes various Angular components like `<button mat-raised-button color="warn">Warn</button>`, `<button mat-icon-button aria-label="Example icon button with a vertical three dot icon"><mat-icon>more_vert</mat-icon></button>`, and `<button mat-icon-button color="primary" aria-label="Example icon button with a home icon"><mat-icon>home</mat-icon></button>`.
- Bottom Status Bar:** Shows build information: "Build at: 2021-11-14T10:26:15.545Z - Hash: 3bac187791d95c7e8663 - Time: 3470ms", "Compiled successfully.", and file statistics: "Ln 7, Col 1 (835 selected) Spaces:4 UTF-8 LF HTML".

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "FIRSTANGULARDEMO". The "app.module.ts" file is currently selected.
- Code Editor:** Displays the content of "app.module.ts". The code defines the AppModule with declarations for AppComponent and CompanyComponent, imports from BrowserModule, Module1Module, BrowserAnimationsModule, MatButtonModule, and MatIconModule, and exports the AppModule.
- Bottom Status Bar:** Shows the build status: "Build at: 2021-11-14T10:26:15.545Z - Hash: 3bac187791d95c7e8663 - Time: 3476ms" and "Compiled successfully."
- Bottom Icons:** Includes icons for node.js, terminal, outline, search, and file operations.

