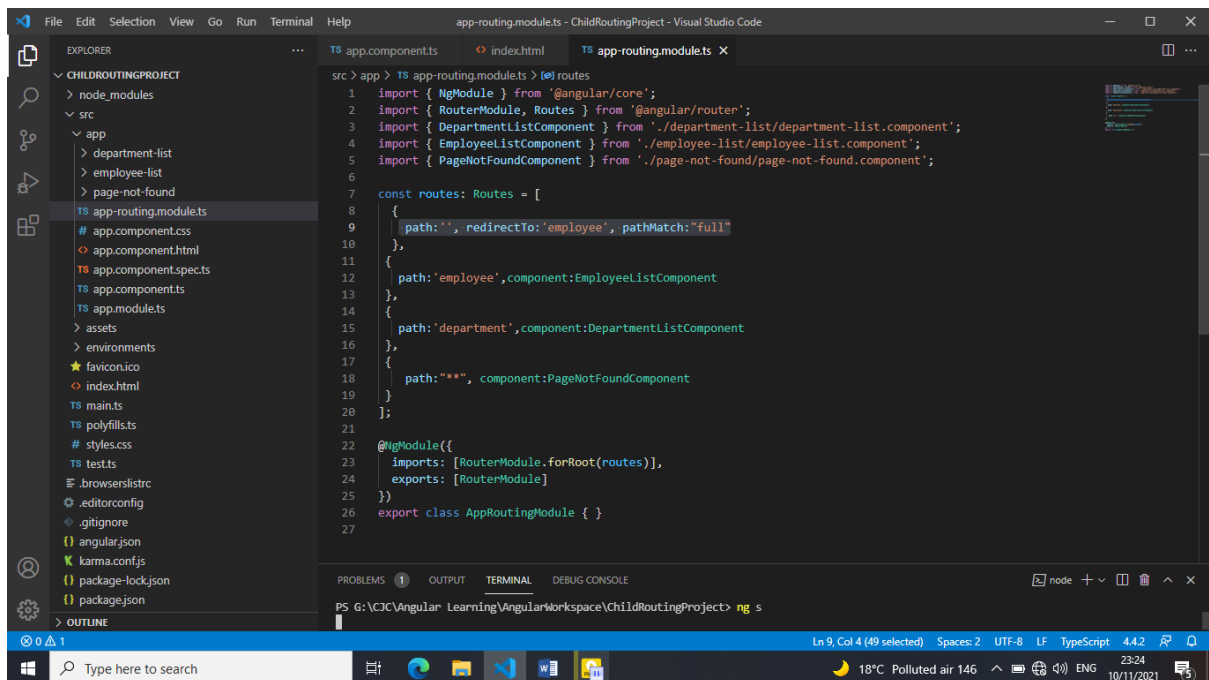


Child Routing

redirectTo: attribute in the app-routing module

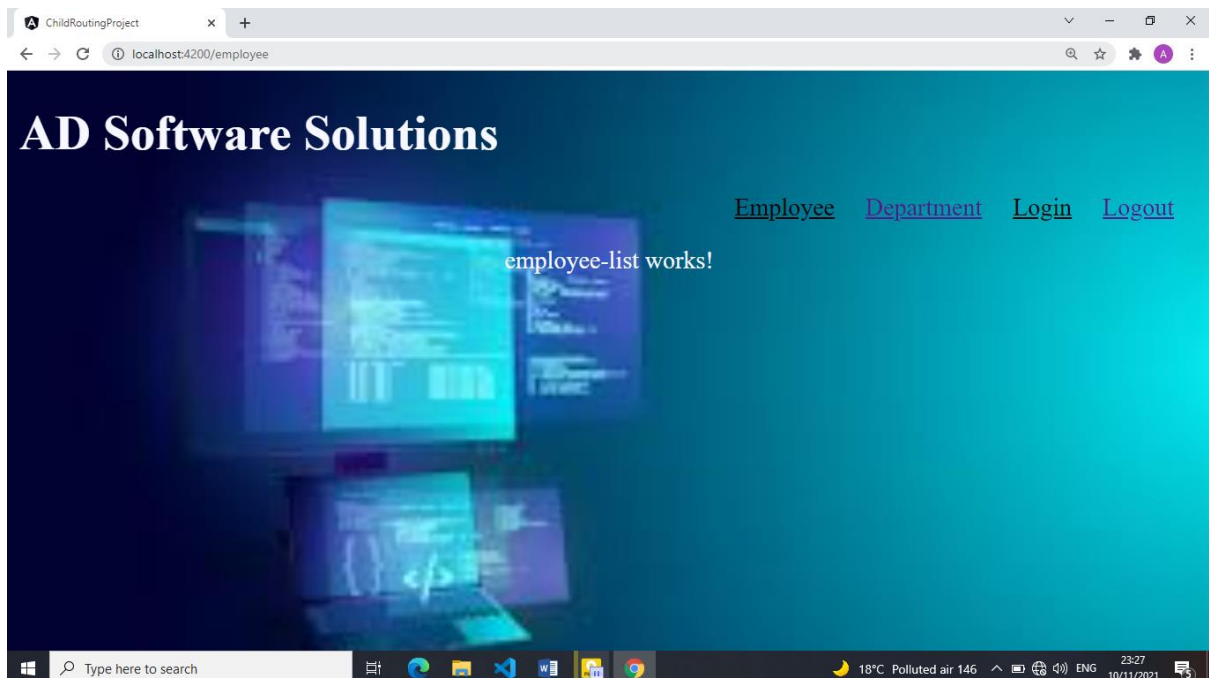
This attribute is used to redirect to the specific action. Below we are redirecting to the “employee” action.



```
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { DepartmentListComponent } from '../department-list/department-list.component';
4 import { EmployeeListComponent } from '../employee-list/employee-list.component';
5 import { PageNotFoundComponent } from '../page-not-found/page-not-found.component';
6
7 const routes: Routes = [
8   {
9     path: '', redirectTo: 'employee', pathMatch: 'full'
10  },
11  {
12    path: 'employee', component: EmployeeListComponent
13  },
14  {
15    path: 'department', component: DepartmentListComponent
16  },
17  {
18    path: '***', component: PageNotFoundComponent
19  }
20 ];
21
22 @NgModule({
23   imports: [RouterModule.forRoot(routes)],
24   exports: [RouterModule]
25 })
26 export class AppRoutingModule { }
27
```

Output:

Every time we hit the localhost: 4200/ it will redirect to localhost:4200/employee.

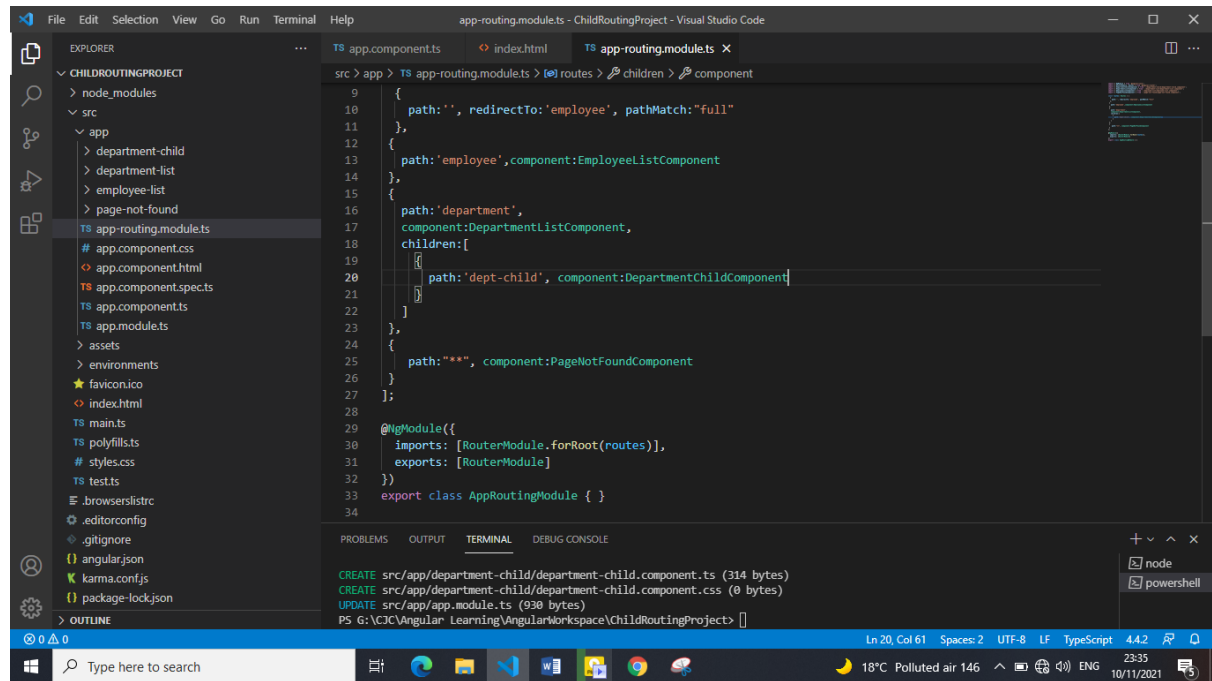


Step 1: create one component which we want to make a child component of already present component.

Now we have create child component as department-child of department-list component

```
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject> ng g c department-child
CREATE src/app/department-child/department-child.component.html (31 bytes)
CREATE src/app/department-child/department-child.component.spec.ts (690 bytes)
CREATE src/app/department-child/department-child.component.ts (314 bytes)
CREATE src/app/department-child/department-child.component.css (0 bytes)
UPDATE src/app/app.module.ts (930 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject>
```

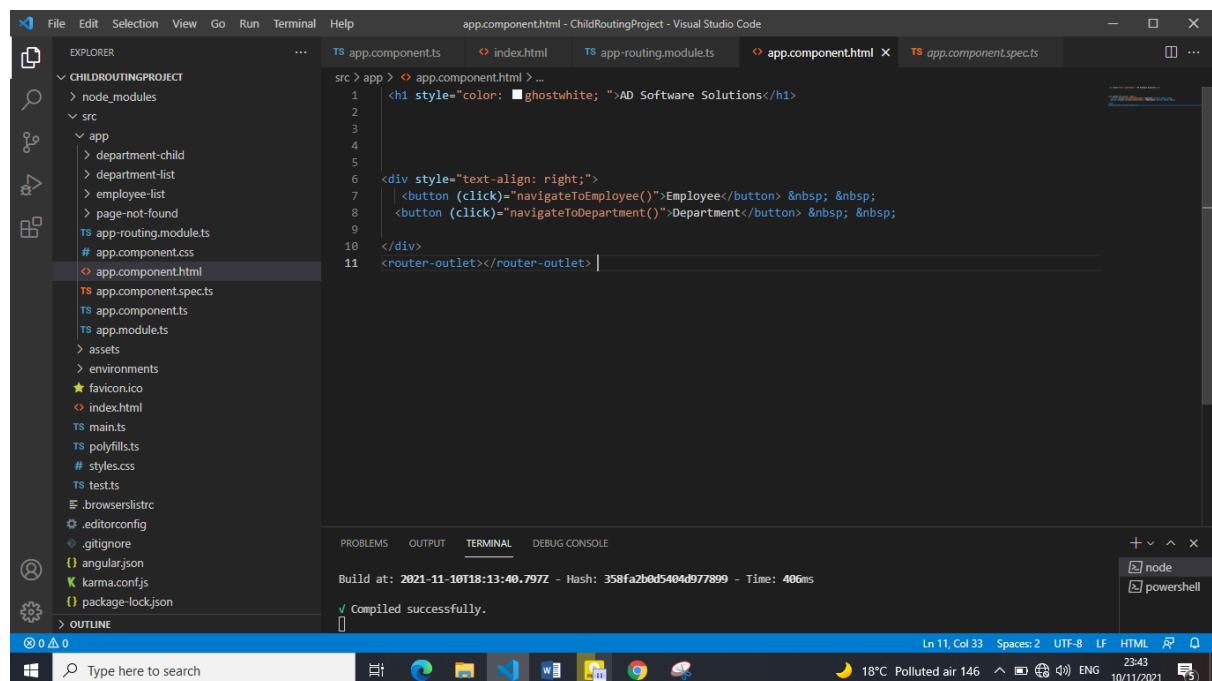
Step 2: create a child 'dept-child' routing in department-list routing



```
src > app > TS app-routing.module.ts > routes > children > component
9
10 {
11   path: '', redirectTo: 'employee', pathMatch: 'full'
12 },
13 {
14   path: 'employee', component: EmployeeListComponent
15 },
16 {
17   path: 'department',
18   component: DepartmentListComponent,
19   children: [
20     {
21       path: 'dept-child', component: DepartmentChildComponent
22     }
23 ],
24 },
25 {
26   path: '***', component: PageNotFoundComponent
27 }
28 ];
29 @NgModule({
30   imports: [RouterModule.forRoot(routes)],
31   exports: [RouterModule]
32 })
33 export class AppRoutingModule { }
34
```

```
CREATE src/app/department-child/department-child.component.ts (314 bytes)
CREATE src/app/department-child/department-child.component.css (0 bytes)
UPDATE src/app/app.module.ts (930 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject>
```

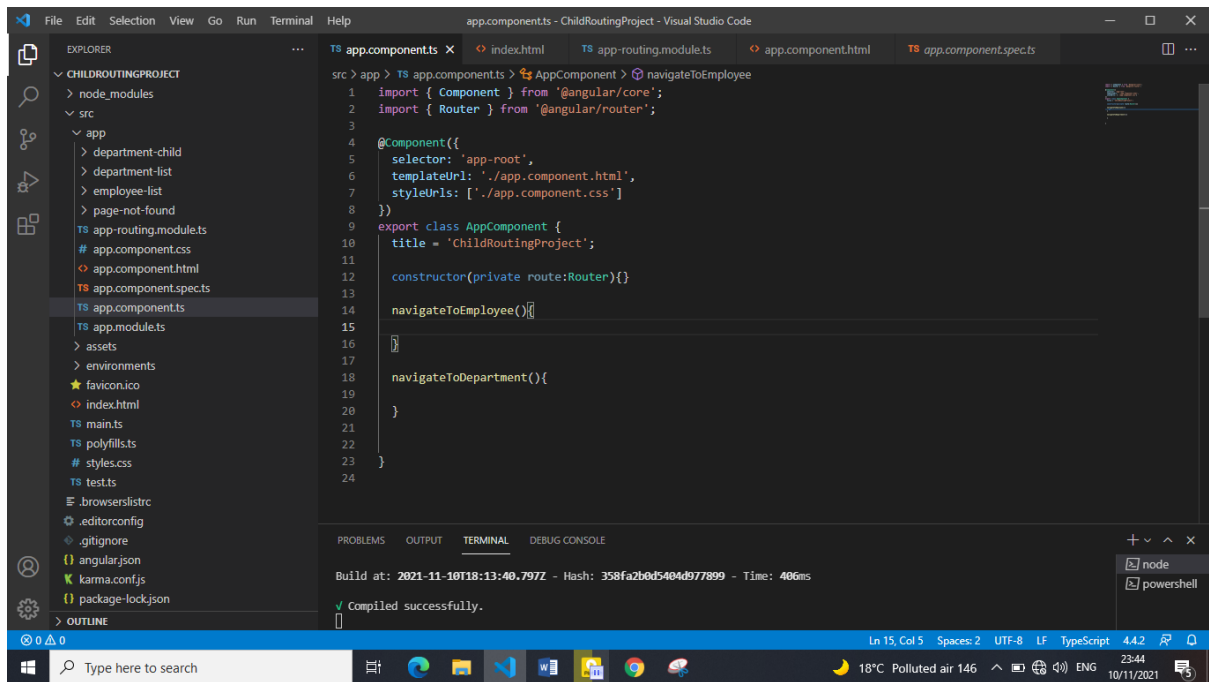
Step 3: want to call the department and dept-child routing at same time from app.component.html



```
src > app > TS app.component.html > ...
1 <h1 style="color: #ghostwhite;">AD Software Solutions</h1>
2
3
4
5
6 <div style="text-align: right;">
7   <button (click)="navigateToEmployee()">Employee</button> &nbsp; &nbsp;
8   <button (click)="navigateToDepartment()">Department</button> &nbsp; &nbsp;
9 </div>
10
11 <router-outlet></router-outlet>
```

```
Build at: 2021-11-10T18:13:40.797Z - Hash: 358fa2bd5404d977899 - Time: 406ms
Compiled successfully.
```

Step 4: create object of Router in app.component.ts in constructor

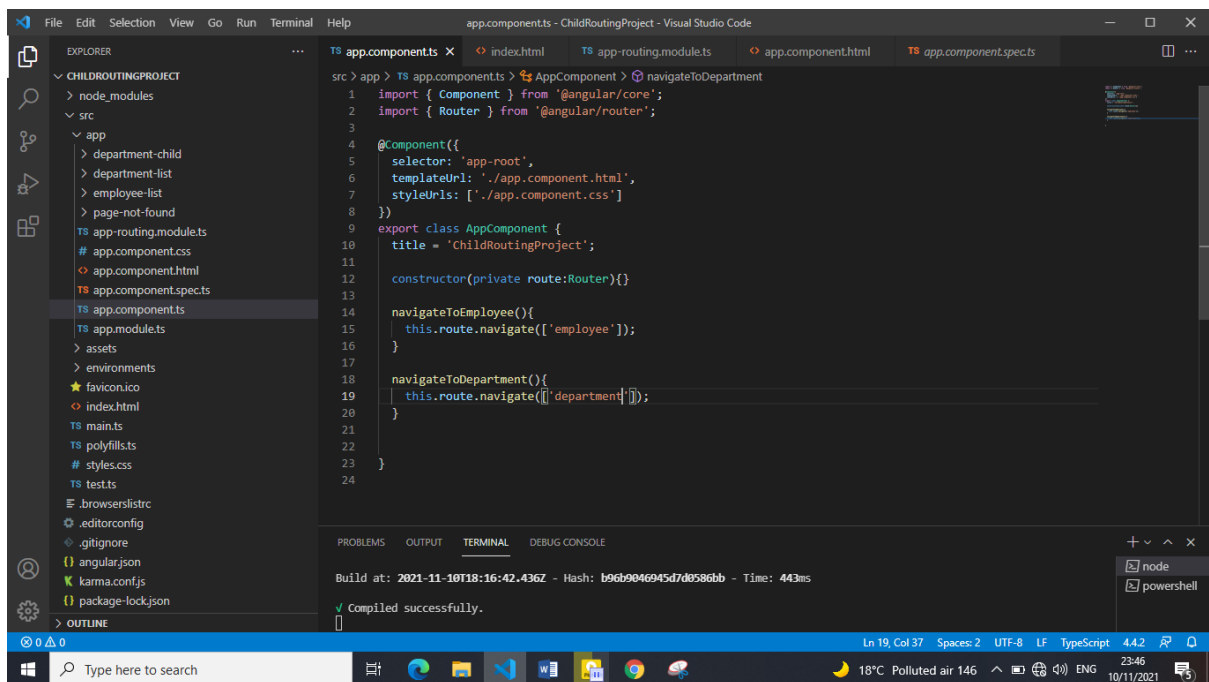


```
src > app > TS app.component.ts > AppComponent > navigateToEmployee
1 import { Component } from '@angular/core';
2 import { Router } from '@angular/router';
3
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.css']
8 })
9 export class AppComponent {
10   title = 'ChildRoutingProject';
11
12   constructor(private route:Router){}
13
14   navigateToEmployee(){}
15
16
17
18   navigateToDepartment(){
19
20   }
21
22
23
24 }
```

Build at: 2021-11-10T18:13:40.797Z - Hash: 358fa2b0d5404d977899 - Time: 406ms

Compiled successfully.

Step 5: add navigate method from route object, this will navigate to that particular routing.

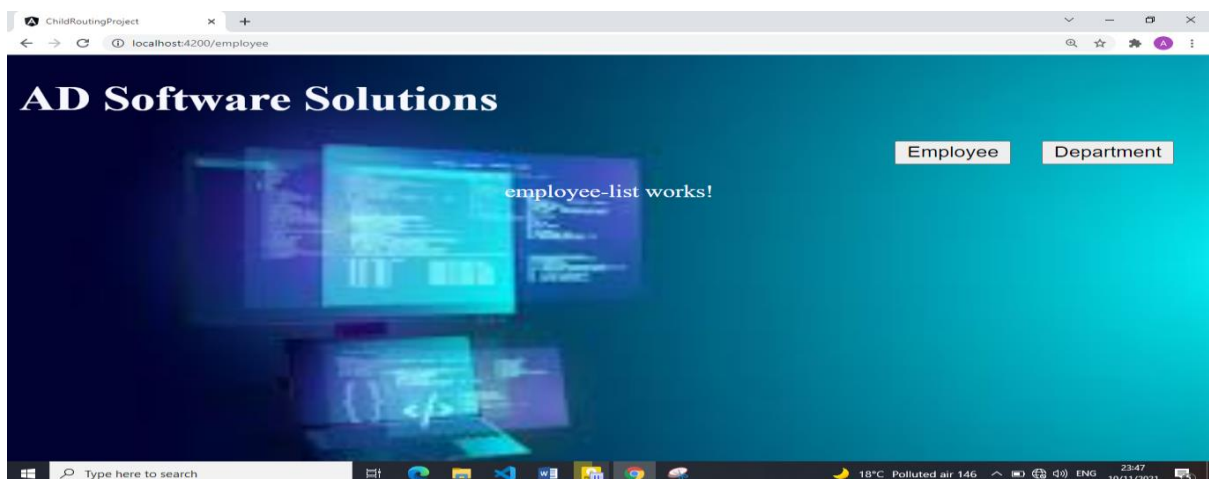


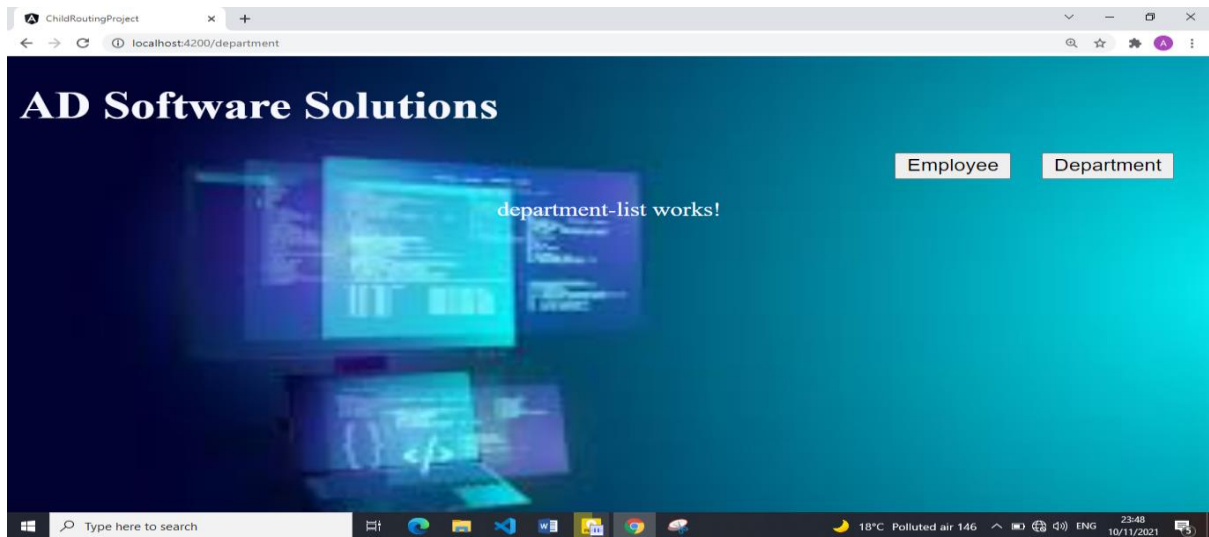
```
src > app > TS app.component.ts > AppComponent > navigateToDepartment
1 import { Component } from '@angular/core';
2 import { Router } from '@angular/router';
3
4 @Component({
5   selector: 'app-root',
6   templateUrl: './app.component.html',
7   styleUrls: ['./app.component.css']
8 })
9 export class AppComponent {
10   title = 'ChildRoutingProject';
11
12   constructor(private route:Router){}
13
14   navigateToEmployee(){
15     this.route.navigate(['employee']);
16   }
17
18   navigateToDepartment(){
19     this.route.navigate(['department']);
20   }
21
22
23
24 }
```

Build at: 2021-11-10T18:16:42.436Z - Hash: b96b9046945d700586bb - Time: 443ms

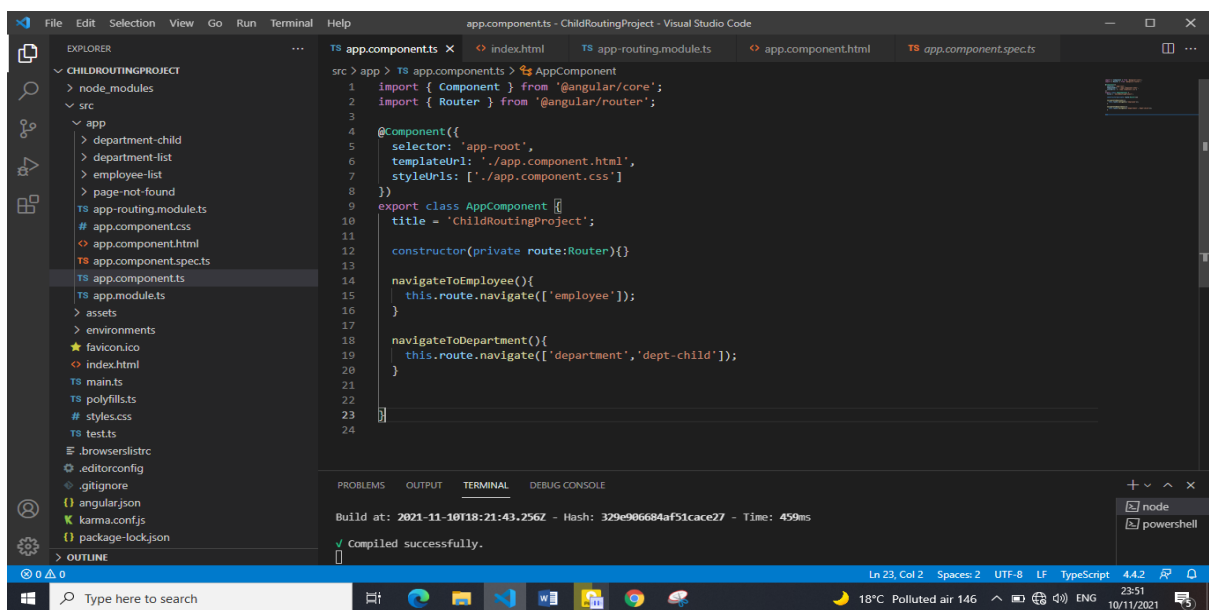
Compiled successfully.

After clicking on employee button, this will navigate to 'employee' routing and after clicking on department button, it will route to department component. PFB screenshot,





Step 6: But now we want to call child routing from department button clicking.

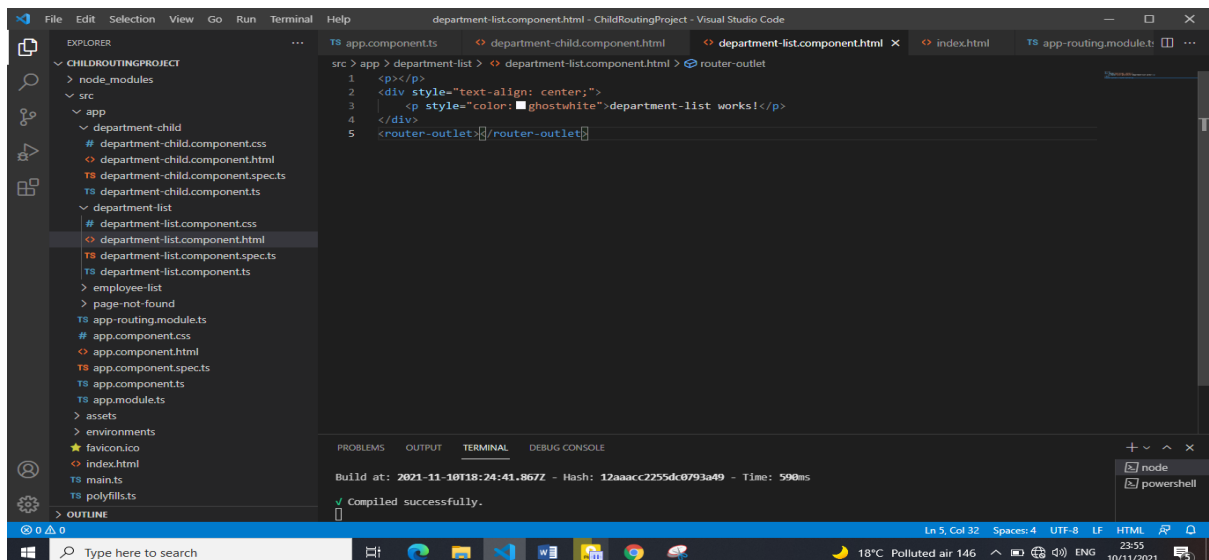


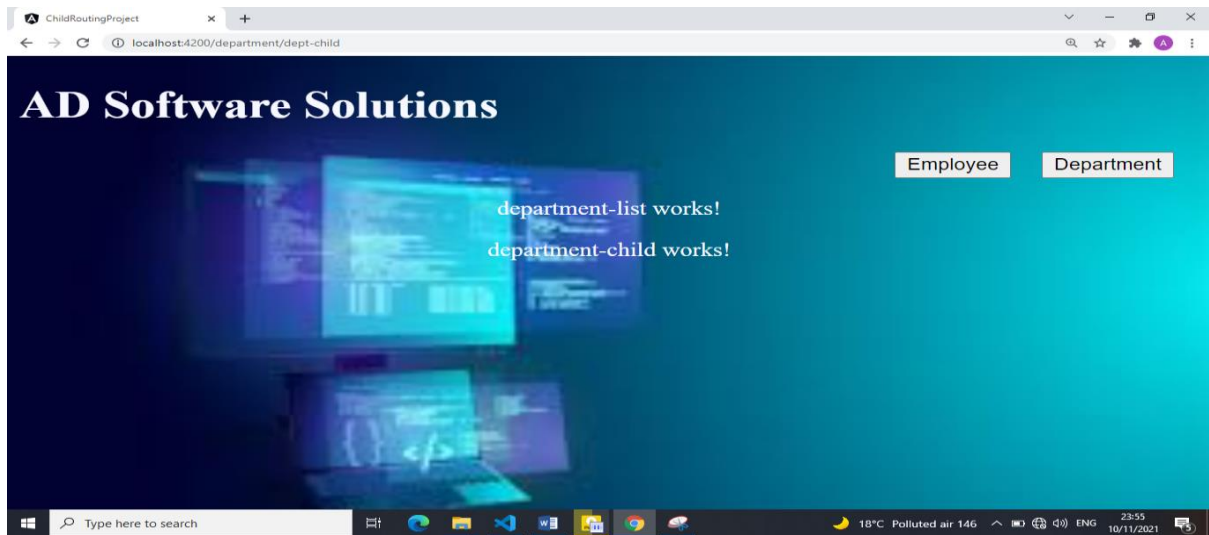
```

navigateToDepartment(){
  this.route.navigate(['department', 'dept-child']);
}

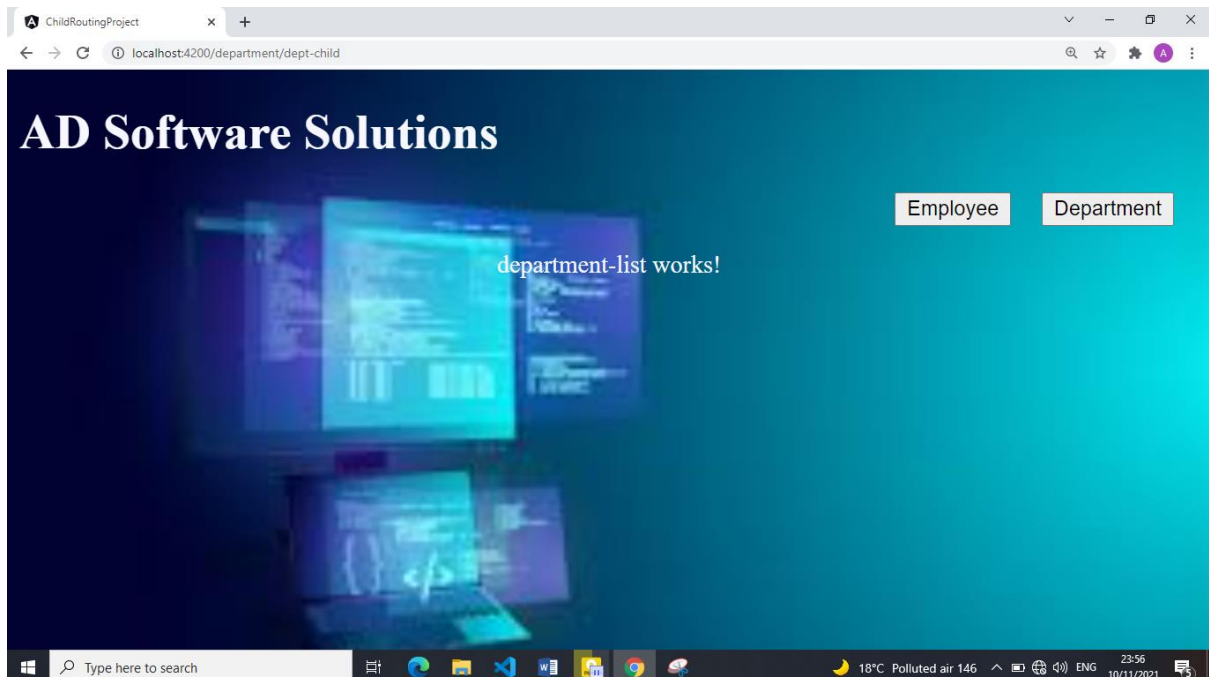
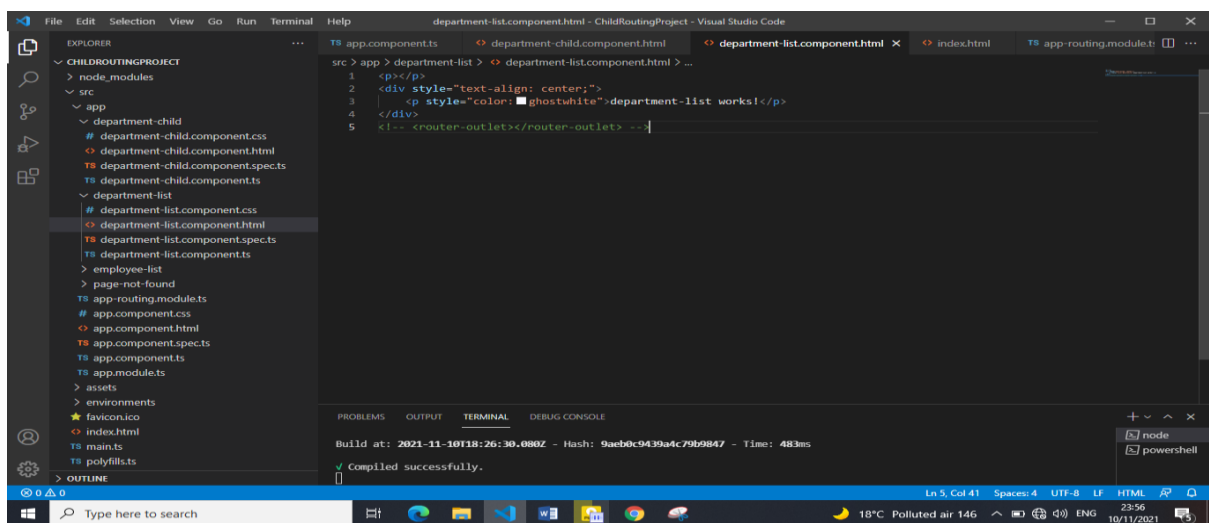
```

Step 7: add <router-outlet> tag parent component i.e department-list component.





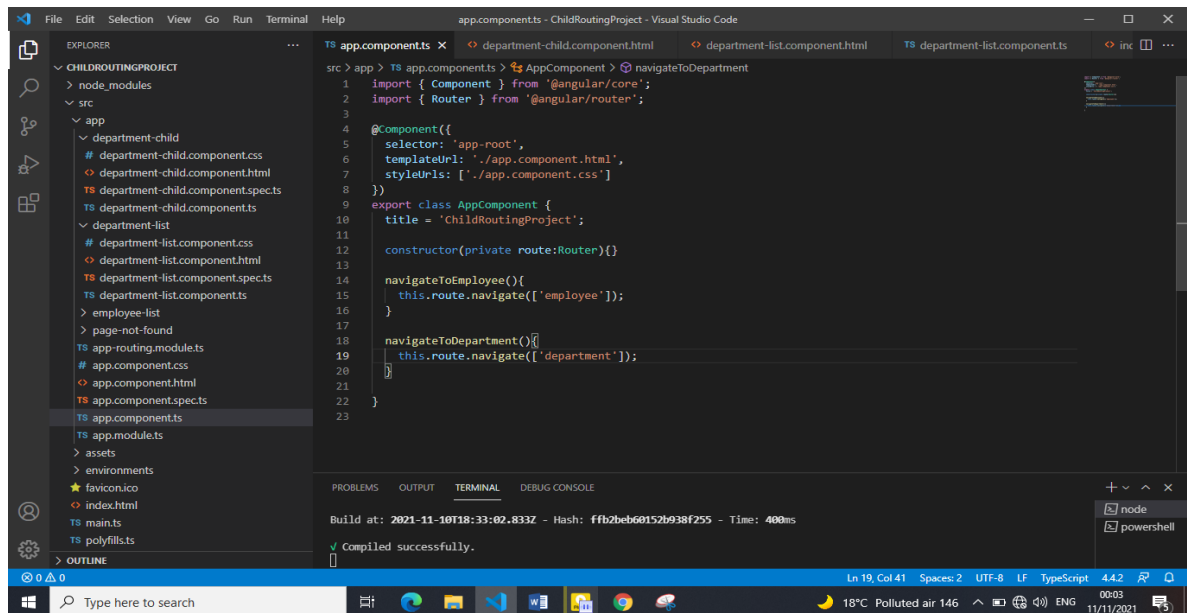
If we remove `<router-outlet>` tag from parent component then



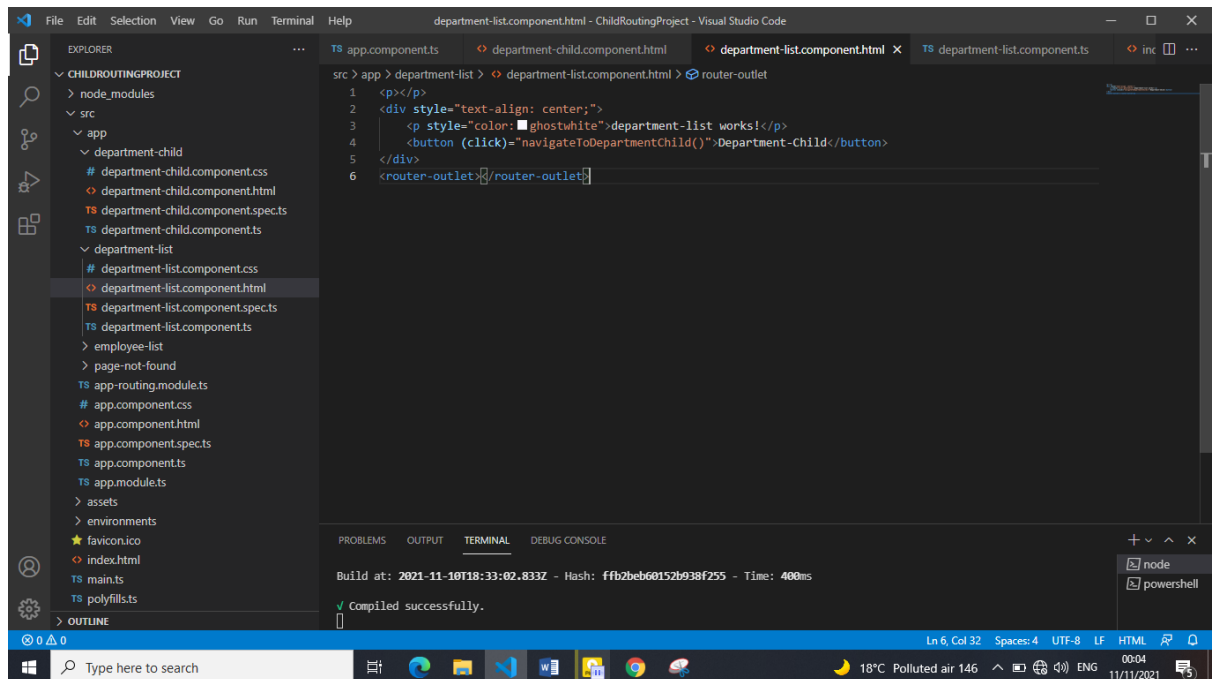
See the URL, its <http://localhost:4200/department/dept-child>, but only department-list works!! will be displayed, not department-child works!!

Step 8: If we don't want same time both components will load then, we have to do some modifications as follows,

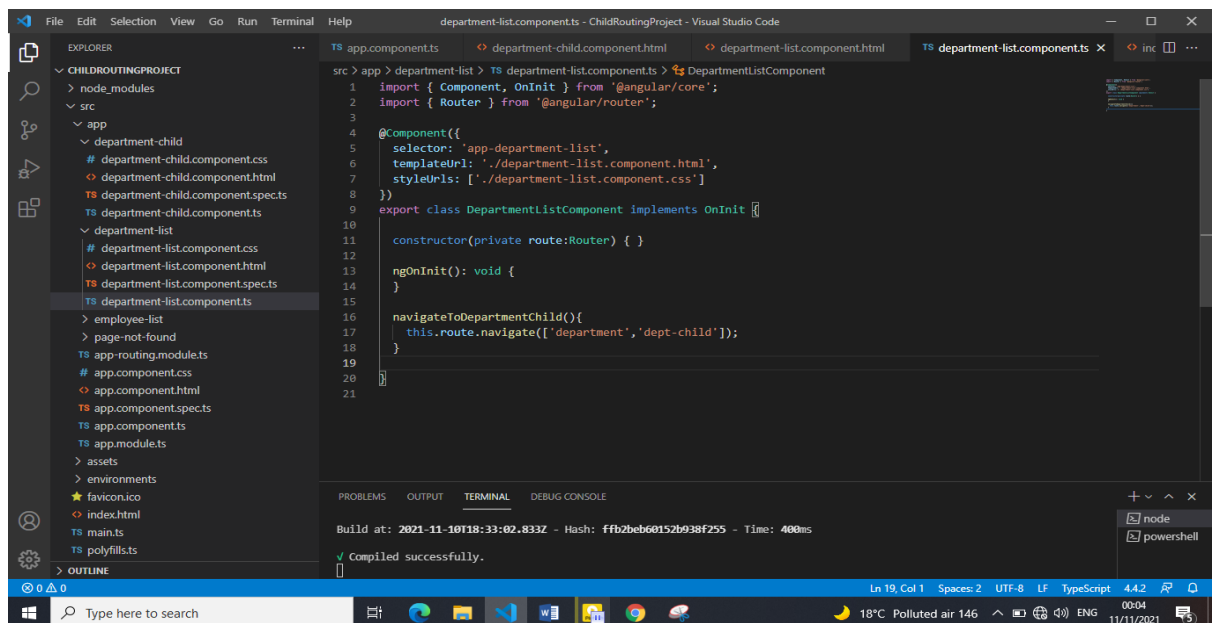
➔ Modify the routes in app.component.ts

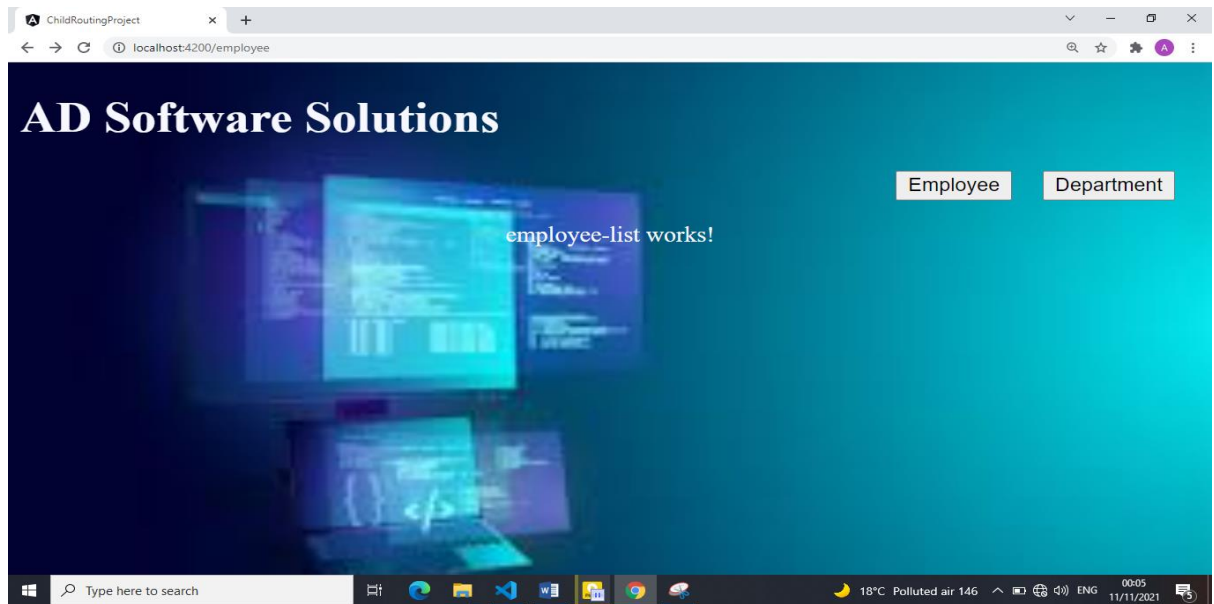


Add one button in department-list.component.html

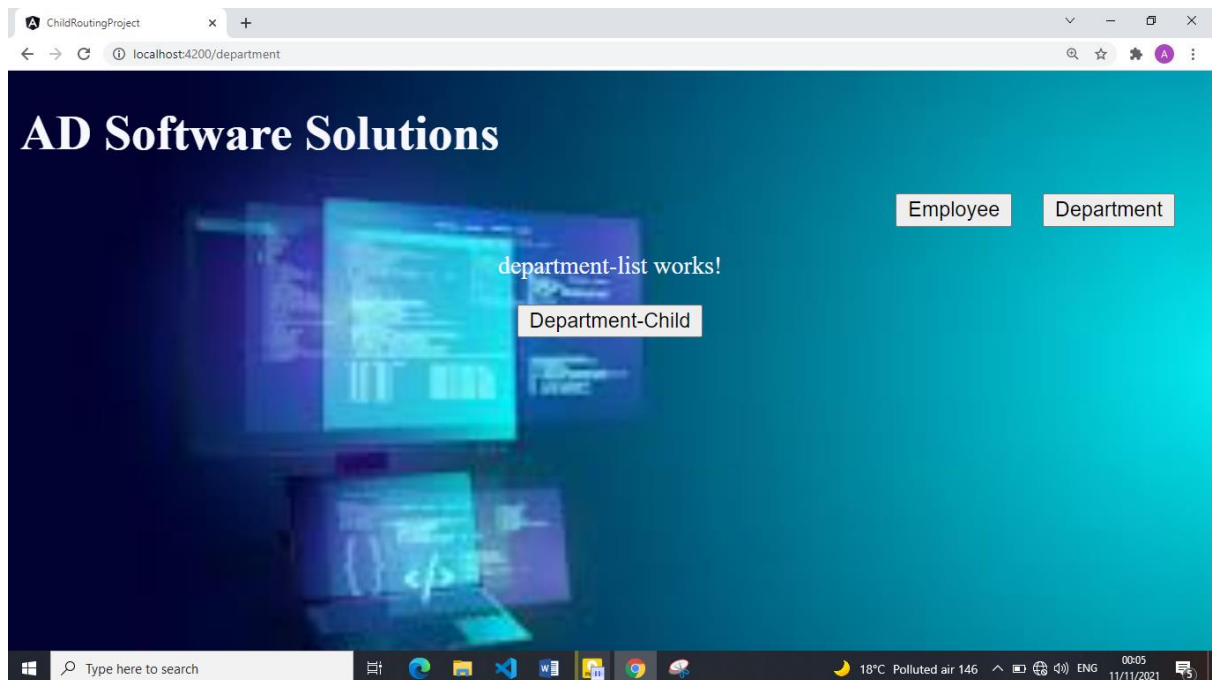


Create object of router in constructor and add method as,

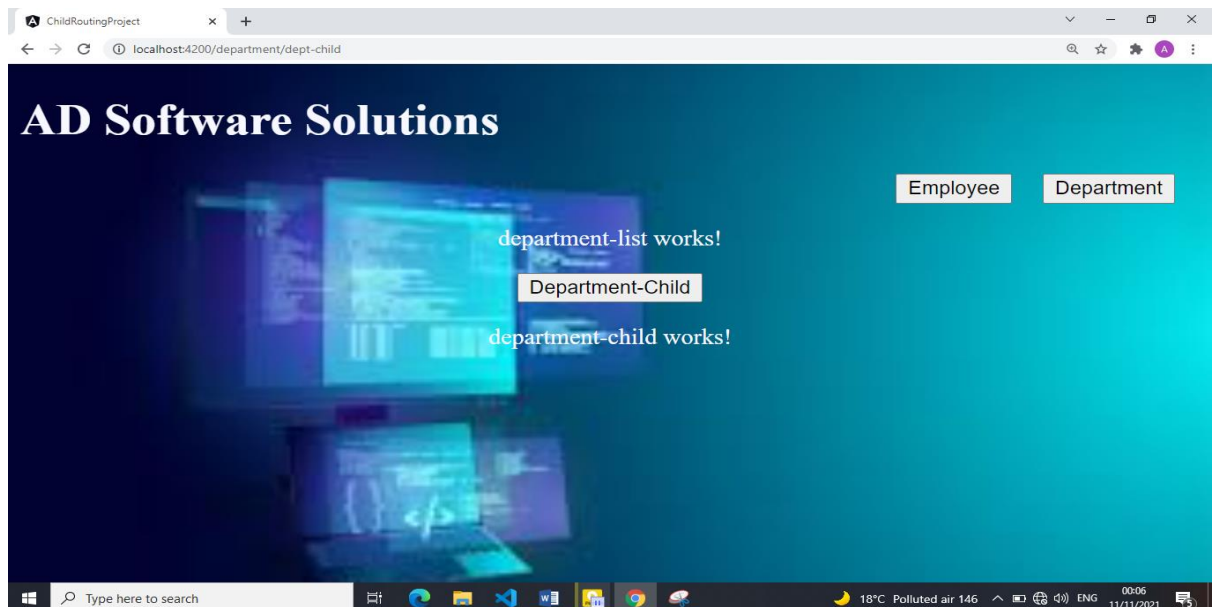




Click on Department button



Click on Department-child button

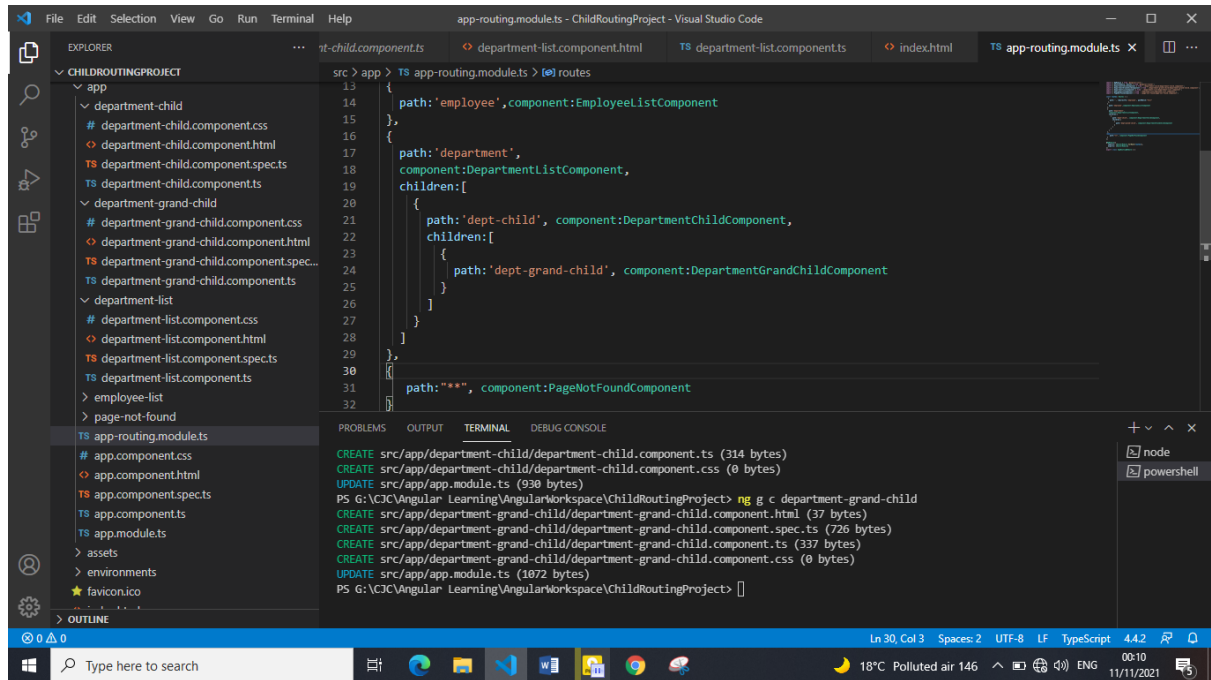


Step 9: Same, we want an again child component of department-child component, means grandchild then,

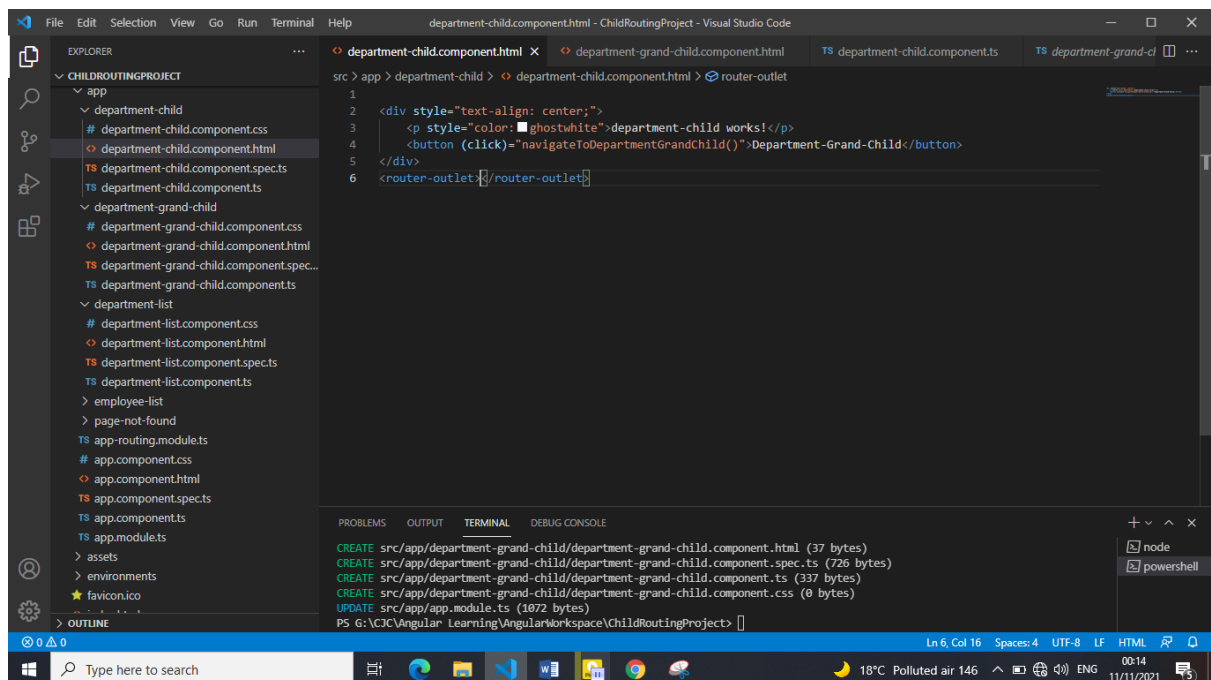
Create one component as department-grand-child

```
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject> ng g c department-grand-child
CREATE src/app/department-grand-child/department-grand-child.component.html (37 bytes)
CREATE src/app/department-grand-child/department-grand-child.component.spec.ts (726 bytes)
CREATE src/app/department-grand-child/department-grand-child.component.ts (337 bytes)
CREATE src/app/department-grand-child/department-grand-child.component.css (0 bytes)
UPDATE src/app/app.module.ts (1072 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject>
```

Add child routing in dept-child routing



Create button in department-child component

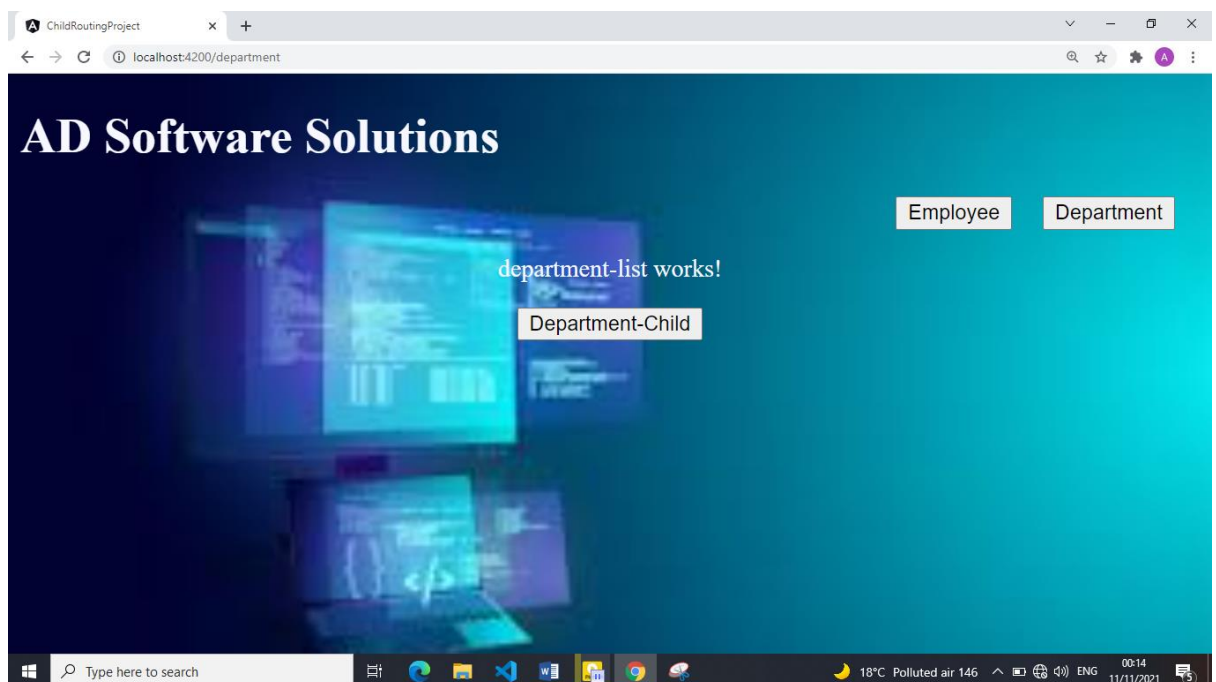
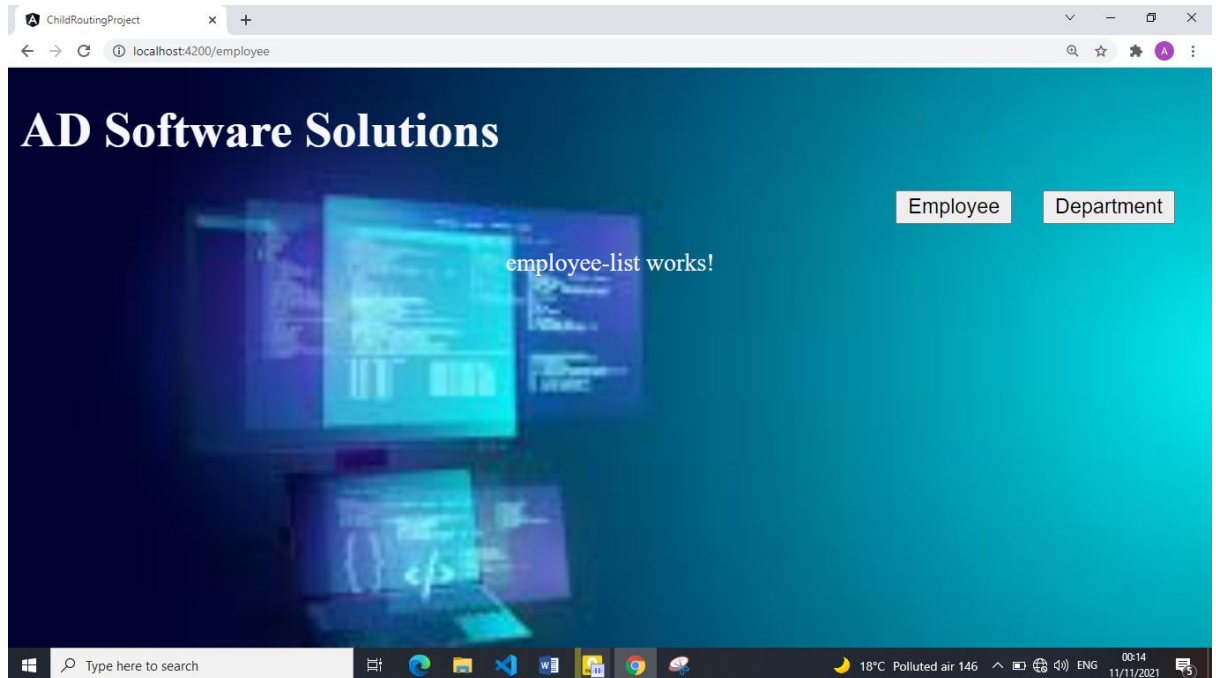


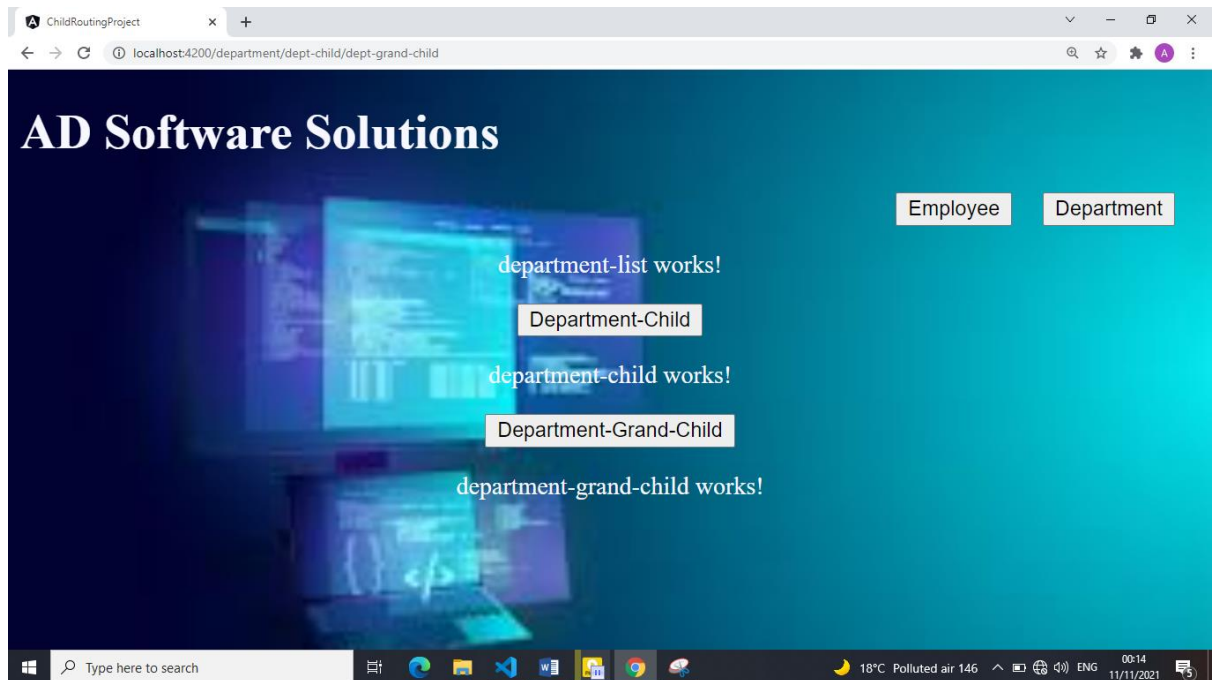
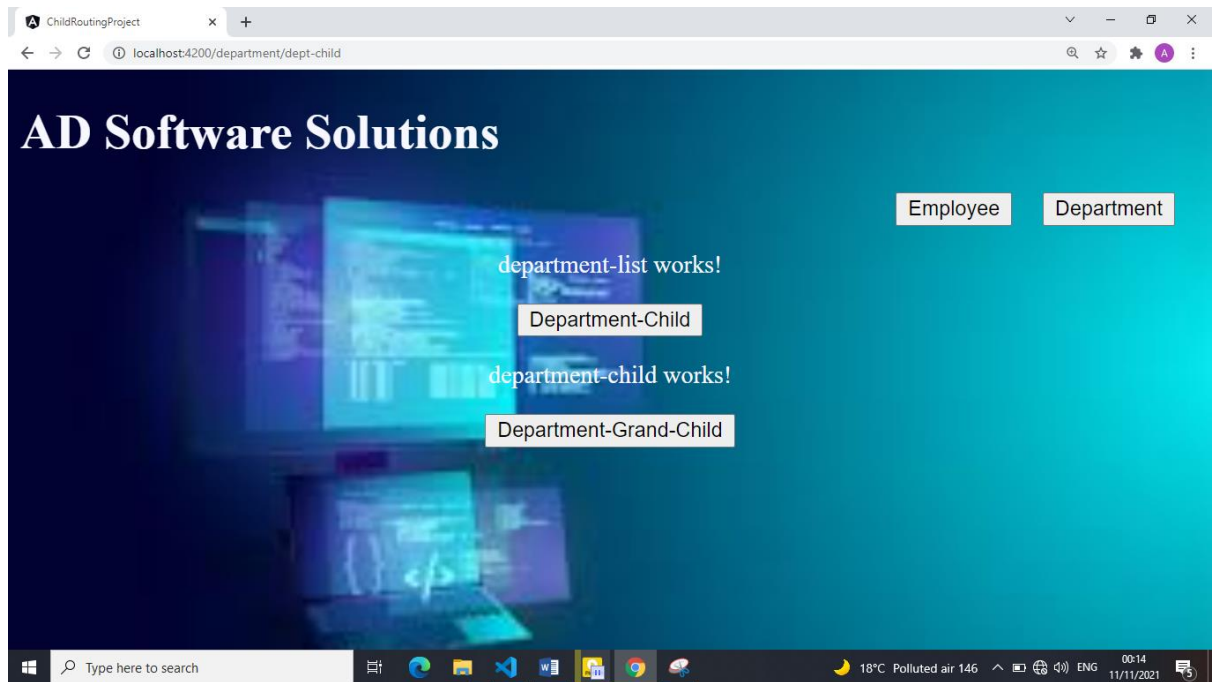

```
src > app > department-child > TS department-child.component.ts > DepartmentChildComponent > navigateToDepartmentGrandChild

1 import { Component, OnInit } from '@angular/core';
2 import { Router } from '@angular/router';
3
4 @Component({
5   selector: 'app-department-child',
6   templateUrl: './department-child.component.html',
7   styleUrls: ['./department-child.component.css']
8 })
9 export class DepartmentChildComponent implements OnInit {
10
11   constructor(private route:Router) { }
12
13   ngOnInit(): void {
14   }
15
16   navigateToDepartmentGrandChild() {
17     this.route.navigate(['department', 'dept-child', 'dept-grand-child']);
18   }
19
20 }
21
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

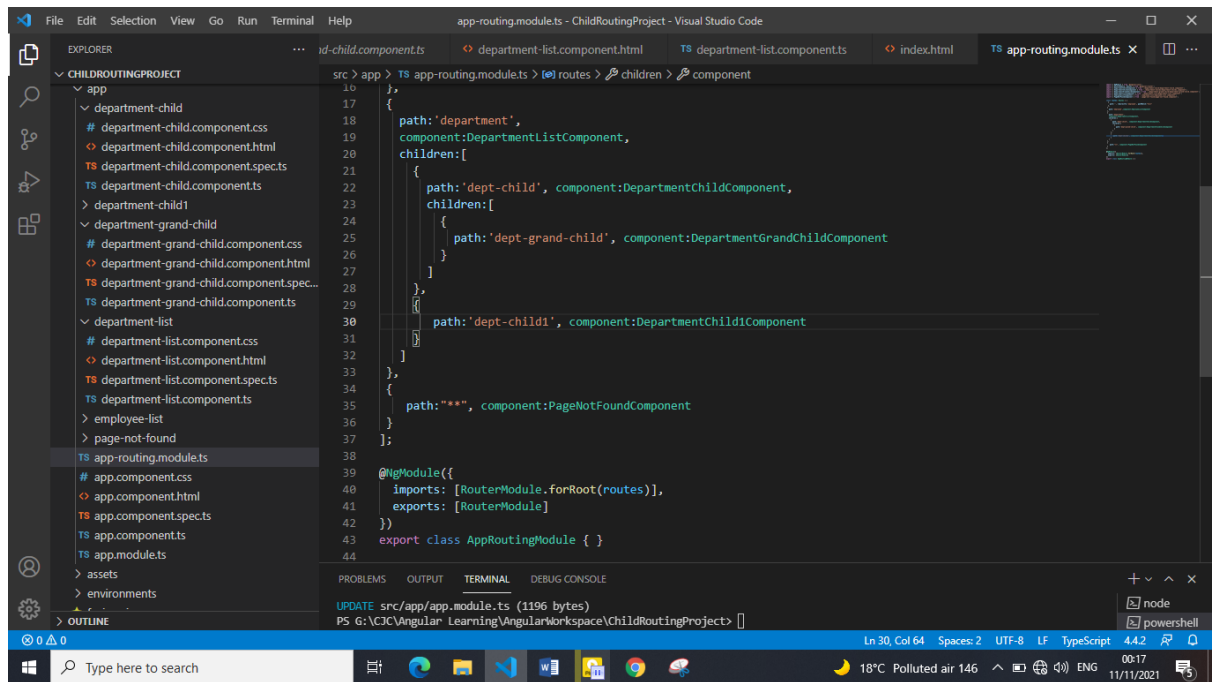
CREATE src/app/department-grand-child/department-grand-child.component.html (37 bytes)
CREATE src/app/department-grand-child/department-grand-child.component.spec.ts (726 bytes)
CREATE src/app/department-grand-child/department-grand-child.component.ts (337 bytes)
UPDATE src/app/app.module.ts (1872 bytes)
PS G:\CJC\Angular_Learning\AngularWorkspace\ChildRoutingProject>



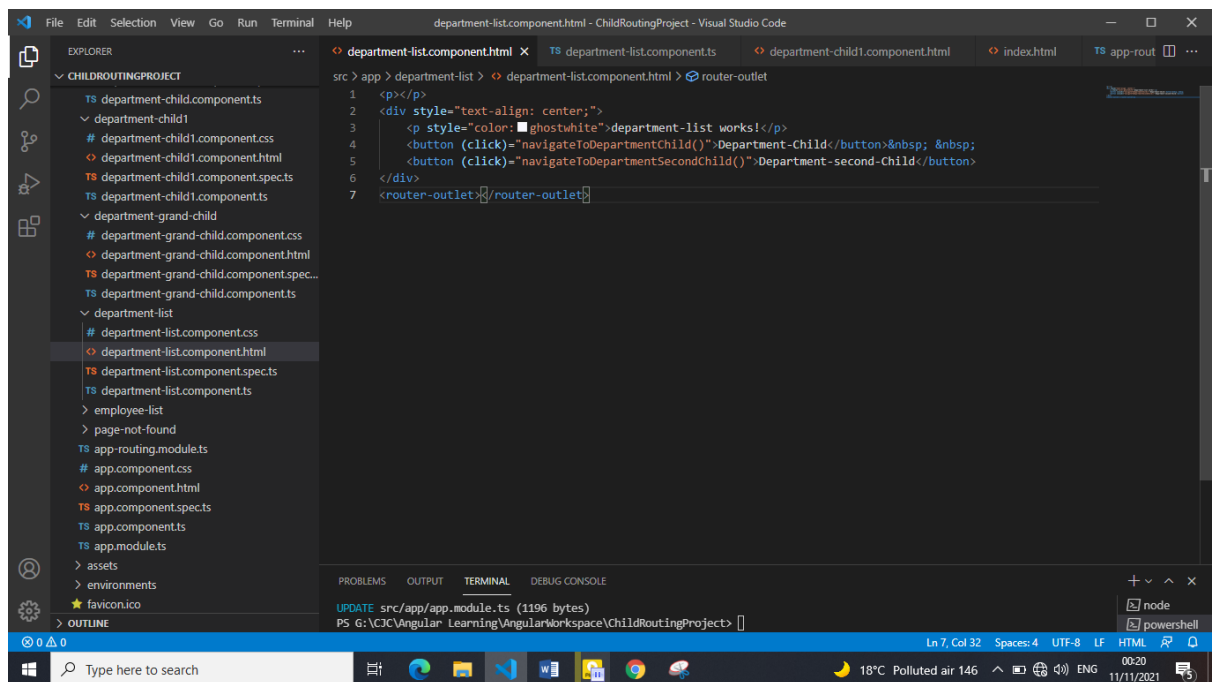


Step 10: If we want a one more child for department-list component

```
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject> ng g c department-child1
CREATE src/app/department-child1/department-child1.component.html (32 bytes)
CREATE src/app/department-child1/department-child1.component.spec.ts (697 bytes)
CREATE src/app/department-child1/department-child1.component.ts (318 bytes)
CREATE src/app/department-child1/department-child1.component.css (0 bytes)
UPDATE src/app/app.module.ts (1196 bytes)
PS G:\CJC\Angular Learning\AngularWorkspace\ChildRoutingProject>
```



Add one more button in



Define the method,

```
navigateToDepartmentSecondChild(){
  this.route.navigate(['department', 'dept-child1']);
}
```