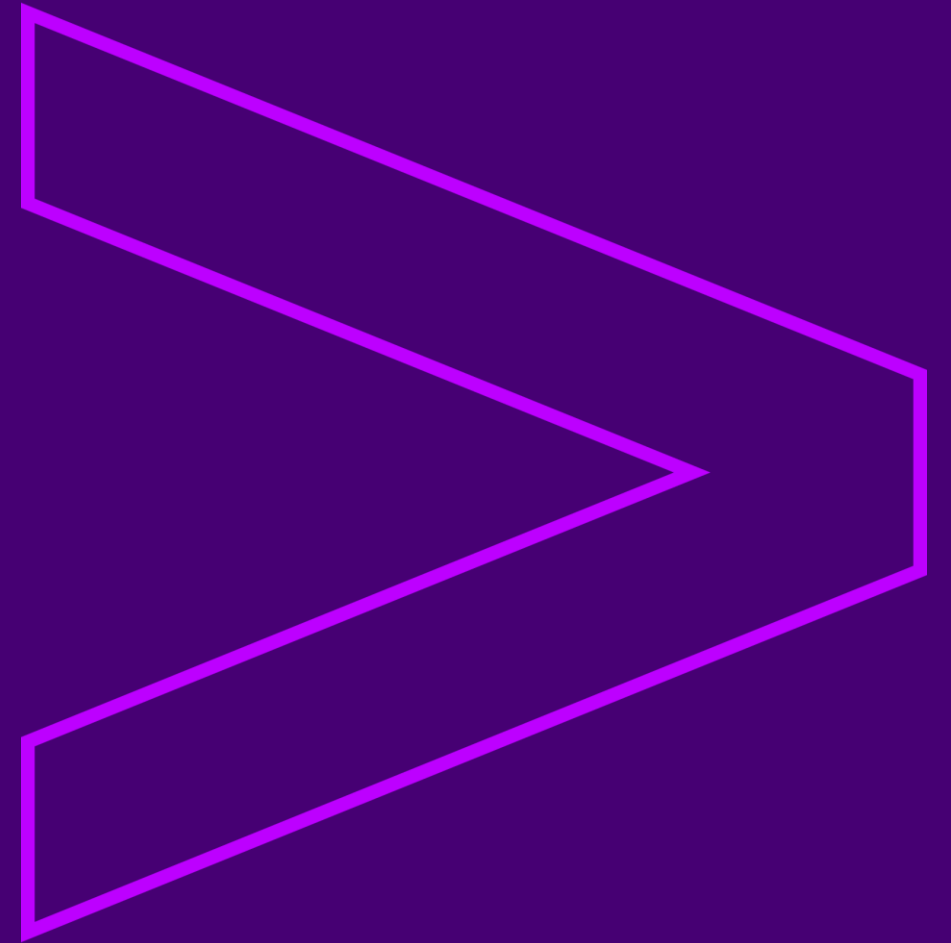


ANGULAR - NESTED ROUTES



MODULE OBJECTIVES

At the end of this module, you should be able to:

- Add Nested routes
- Create child routes with route params



AGENDA

- ▶ Nesting of Routes



Nesting Routes

- All the routes created so far are relative to the root component
- But when the application grows in size, we may want to have routes relative to other components as well
- These types of nested routes are also called as **child routes**.
- Adding child routes implies adding more `<router-outlet>` to the app, besides the `<router-outlet>` already added in AppComponent's template.

Child Routes – Problem Definition

- Consider the Employee List Page. If the user wants to click on the Empno and see the employee details in the same view along with the Employee List .

Child Routes Demo Application

[Home](#) [Employees List](#) [Departments List](#) [Products List](#)

Employees List

Employee No	Employee Name	Employee Age	Employee Salary
1231	Thangaraj	21	12345.11
1232	Sindhu	22	34512.22
1233	Snigdha	23	12343.33
1234	Bhagya	24	12344.44
1235	Bhavya	25	12345.55
1236	Pavana	26	12346.66
1237	Seema	27	12347.77
1238	Karthik	28	12348.88
1239	Prashantha	29	12349.99
1240	Manoj	30	12340

::::Employee Details::::

Employee Details of Employee No: 1234

Employee Details	
Employee No	1234
Employee Name	Bhagya
Employee Age	24
Employee Salary	12344.44

.....End of Page.....

Child Route - Configuration

- Create a new component EmployeeComponent and update the AppRoutingModuleModule to include a new route mapping

```
const routes: Routes=[
  {path:'home', component:HomeComponent},
  {path:'employees', component:EmployeesListComponent,
    children: [{path('/:empno',component:EmployeeComponent} ]
  },
  {path:'departments', component:DepartmentListComponent},
  {path:'products', component:ProductsListComponent},
  {path:'products/:prodno',component:ProductComponent},
  {path: '', redirectTo: '/home', pathMatch: 'full' },
  {path: '**', component:HomeComponent}
]
```

Child route
configured to accept
an additional
parameter and
render
EmployeeComponent

Modify EmployeeService

- Add an additional method **getEmployeeByEmployeeNo(eno:number)** in EmployeeService class which accepts the employee number and returns an Employee Object

```
import { Injectable } from '@angular/core';
import { Employee } from './employee';
@Injectable()
export class EmployeeService {
  empno:number;
  employee:Employee;
  constructor() { }
  employees = [
    {empno:1231,empname:'Thangaraj',age:21,salary:12345.11},
    {empno:1232,empname:'Sindhu',age:22,salary:34512.22},
    {empno:1233,empname:'Snigdha',age:23,salary:12343.33},
    {empno:1234,empname:'Bhagya',age:24,salary:12344.44},
    {empno:1235,empname:'Bhavya',age:25,salary:12345.55},
    {empno:1236,empname:'Pavana',age:26,salary:12346.66},
    {empno:1237,empname:'Seema',age:27,salary:12347.77},
    {empno:1238,empname:'Karthik',age:28,salary:12348.88},
    {empno:1239,empname:'Prashantha',age:29,salary:12349.99},
    {empno:1240,empname:'Manoj',age:30,salary:12340.00}
  ];
```

```
getEmployees()
{
  return this.employees;
}

getEmployeeByEmployeeNo(eno:number)
{
  for(let i=0;i<this.employees.length;i++)
  {
    if (this.employees[i].empno==eno)
    {
      this.employee = this.employees[i];
      return this.employee;
    }
  }
}
```

method accepts
employee number and
returns an Employee
Object.

Setting the routerLink Directive

- In employee-list.component.html, the routerLink directive is configured

```
<div align="center" style="background-color:#AD9820 ; width:600px;height:640px">
<h1 align="center"> Employees List </h1>
<table align="center" border="solid 3px" >
<thead>
<tr>
<td align="center"><b>Employee No</b></td>
<td align="center"><b>Employee Name</b></td>
<td align="center"><b>Employee Age</b></td>
<td align="center"><b>Employee Salary</b></td>
</tr>
</thead>
<tbody>
<tr *ngFor="let employee of employees">
<td align="center"><a
[routerLink]="['/employees/',employee.empno]">{{employee.empno}}</a></td>
<td align="center">{{employee.empname}}</td>
<td align="center">{{employee.age}}</td>
<td align="center">{{employee.salary}}</td>
</tr>
</tbody>
</table>
<router-outlet></router-outlet>
<h2 align="center">::::::::::::::::::::End of Page::::::::::::::::::::</h2>
</div>
```

routerLink is configured to accept an additional parameter

router-outlet to display Employee Details. (This router-outlet is inside employee-list.component.html)

Defining the EmployeeComponent

```
import { Component, OnInit } from '@angular/core';
import { Employee } from '../employee';
import { EmployeeService } from '../employee.service';
import { ActivatedRoute } from '@angular/router';
@Component({
  selector: 'app-employee',
  templateUrl: './employee.component.html',
})
export class EmployeeComponent implements OnInit {

  EmpNo:any;
  employee:Employee;
  constructor(private employeeService:EmployeeService,private route: ActivatedRoute){
  }
  ngOnInit()
  {
    this.route.paramMap.subscribe(params =>
    {
      this.EmpNo = params['empno']; // 'empno' is the same name used while configuring the route in app.module.ts
      console.log("Employee No Searched = "+this.EmpNo);
      this.employee = this.employeeService.getEmployeeByEmployeeNo(this.EmpNo); // retrieving the product details
    })
  }
}
```

Import
ActivatedRoute from
@angular/router

Extracting employee
number(parameter)
from the URL.

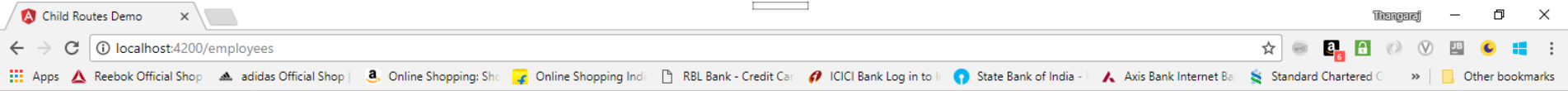
Invoking the
getEmployeeByEmployeeNo
method from
EmployeeService

Defining the EmployeeComponent's view

```
<div style="background-color:#AD206D ;
width:350px;height:240px">
<h3 align="center">:::Employee Details:::</h3>
<h4 align="center"> Employee Details of Employee No:
{{EmpNo}}</h4>
<table align="center" border="2" >
<thead>
<td [colSpan]=2 align="center"><b>Employee
Details</b></td>
</thead>
<tbody>
<tr>
<td align="left" >Employee No</td>
<td align="center">{{employee.empno}}</td>
</tr>
<tr>
<td align="left">Employee Name</td>
<td align="center">{{employee.empname}}</td>
</tr>
```

```
<tr>
<td align="left">Employee Age</td>
<td align="center">{{employee.age}}</td>
</tr>
<tr>
<td align="left">Employee Salary</td>
<td align="center">{{employee.salary}}</td>
</tr>
</tbody>
</table>
</div>
```

Child Routes Output



Child Routes Demo Application

[Home](#) [Employees List](#) [Departments List](#) [Products List](#)

Employees List

Employee No	Employee Name	Employee Age	Employee Salary
1231	Thangaraj	21	12345.11
1232	Sindhu	22	34512.22
1233	Prashantha	23	12343.33
1234	Bhagya	24	12344.44
1235	Bhavya	25	12345.55
1236	Pavana	26	12346.66
1237	Seema	27	12347.77
1238	Karthik	28	12348.88
1239	Prashantha	29	12349.99
1240	Manoj	30	12340

.....End of Page.....

Click on any employee number



Child Routes Output

Child Routes Demo

localhost:4200/employees/1232

Child Routes Demo Application

[Home](#) [Employees List](#) [Departments List](#) [Products List](#)

Employees List

Employee No	Employee Name	Employee Age	Employee Salary
1231	Thangaraj	21	12345.11
1232	Sindhu	22	34512.22
1233	Snigdha	23	12343.33
1234	Bhagya	24	12344.44
1235	Bhavya	25	12345.55
1236	Pavana	26	12346.66
1237	Seema	27	12347.77
1238	Karthik	28	12348.88
1239	Prashantha	29	12349.99
1240	Manoj	30	12340

::::Employee Details::::

Employee Details of Employee No: 1232

Employee Details	
Employee No	1232
Employee Name	Sindhu
Employee Age	22
Employee Salary	34512.22

.....End of Page.....

URL is appended with the selected employee number.

Child Route Displaying the selected Employee

MODULE SUMMARY

Now, you should be able to:

- Add Nested routes
- Demonstrate creating child routes with route params



QUESTIONS



THANK YOU