Employee.java:

package com.cognizant.springlearn.model;

public class Employee {

private int id;

private String name;

private double salary;

private boolean permanent;

private String department;

// Getters and setters

// toString method

}

employee.xml:

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans.xsd">

<bean id="employeeList" class="java.util.ArrayList">

<constructor-arg>

<list>

<bean class="com.cognizant.springlearn.model.Employee">

<property name="id" value="1"/>

<property name="name" value="John"/>

<property name="salary" value="50000"/>

<property name="permanent" value="true"/>

<property name="department" value="HR"/>

</bean>

<bean class="com.cognizant.springlearn.model.Employee">

<property name="id" value="2"/>

<property name="name" value="Jane"/>

<property name="salary" value="60000"/>

<property name="permanent" value="false"/>

<property name="department" value="Finance"/>

</bean>

</list>

</constructor-arg>

</bean>

</beans>

**EmployeeService.java:**

package com.cognizant.springlearn.service;

import com.cognizant.springlearn.model.Employee;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class EmployeeService {

public List<Employee> getAllEmployees() {

ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml");

return (List<Employee>) context.getBean("employeeList");

}

public Employee getEmployeeById(int id) {

return getAllEmployees().stream()

.filter(e -> e.getId() == id)

.findFirst()

.orElse(null);

}

}

EmployeeController.java:

package com.cognizant.springlearn.controller;

import com.cognizant.springlearn.model.Employee;

import com.cognizant.springlearn.service.EmployeeService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

@CrossOrigin(origins = "\*")

public class EmployeeController {

@Autowired

private EmployeeService employeeService;

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

@GetMapping("/employees/{id}")

public Employee getEmployee(@PathVariable int id) {

return employeeService.getEmployeeById(id);

}

}

employee.service.ts

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

import { Observable } from 'rxjs';

export interface Employee {

id: number;

name: string;

salary: number;

permanent: boolean;

department: string;

}

@Injectable({

providedIn: 'root'

})

export class EmployeeService {

private baseUrl = 'http://localhost:8080';

constructor(private http: HttpClient) { }

getAllEmployees(): Observable<Employee[]> {

return this.http.get<Employee[]>(`${this.baseUrl}/employees`);

}

getEmployeeById(id: number): Observable<Employee> {

return this.http.get<Employee>(`${this.baseUrl}/employees/${id}`);

}

}

employee-list.component.ts:

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

import { EmployeeService, Employee } from '../employee.service';

import { Router } from '@angular/router';

@Component({

selector: 'app-employee-list',

templateUrl: './employee-list.component.html'

})

export class EmployeeListComponent implements OnInit {

employees: Employee[] = [];

constructor(private employeeService: EmployeeService, private router: Router) {}

ngOnInit() {

this.employeeService.getAllEmployees().subscribe(data => {

this.employees = data;

});

}

editEmployee(id: number) {

this.router.navigate(['/edit', id]);

}

}

edit-employee.component.ts:

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

import { ActivatedRoute } from '@angular/router';

import { EmployeeService, Employee } from '../employee.service';

@Component({

selector: 'app-edit-employee',

templateUrl: './edit-employee.component.html'

})

export class EditEmployeeComponent implements OnInit {

employee: Employee | undefined;

constructor(private route: ActivatedRoute, private employeeService: EmployeeService) {}

ngOnInit() {

const id = +this.route.snapshot.paramMap.get('id')!;

this.employeeService.getEmployeeById(id).subscribe(data => {

this.employee = data;

});

}

}

employee-list.component.html:

<h2>Employee List</h2>

<table>

<tr>

<th>ID</th><th>Name</th><th>Salary</th><th>Department</th><th>Actions</th>

</tr>

<tr \*ngFor="let emp of employees">

<td>{{ emp.id }}</td>

<td>{{ emp.name }}</td>

<td>{{ emp.salary }}</td>

<td>{{ emp.department }}</td>

<td><button (click)="editEmployee(emp.id)">Edit</button></td>

</tr>

</table>

edit-employee.component.html:

<h2>Edit Employee</h2>

<div \*ngIf="employee">

<p>ID: {{ employee.id }}</p>

<p>Name: {{ employee.name }}</p>

<p>Salary: {{ employee.salary }}</p>

<p>Department: {{ employee.department }}</p>

</div>