**CTS DIGITAL NURTURE - 4.0 JAVA FSE**

**WEEK 3 -Spring Core and Maven**

**Spring Data JPA with Spring Boot, Hibernate**

**Spring Data JPA - Quick Example**

EmployeeManagementSystem

**EmployeeManagementSystemApplication.java**

package com.example.employeemanagementmystem;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class EmployeeManagementSystemApplication {

public static void main(String[] args) {

SpringApplication.*run*(EmployeeManagementSystemApplication.class, args);

}

}

**Employee.java**

package com.example.employeemanagementmystem.entity;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "employees")

public class Employee {

*@Id*

*@GeneratedValue*(strategy = *GenerationType*.*IDENTITY*)

private Long id;

private String name;

private String department;

private double salary;

// Getters and setters

public Long getId() { return id; }

public void setId(Long id) { this.id = id; }

public String getName() { return name; }

public void setName(String name) { this.name = name; }

public String getDepartment() { return department; }

public void setDepartment(String department) { this.department = department; }

public double getSalary() { return salary; }

public void setSalary(double salary) { this.salary = salary; }

*@Override*

public String toString() {

return "Employee{" +

"id=" + id +

", name='" + name + '\'' +

", department='" + department + '\'' +

", salary=" + salary +

'}';

}

}

**EmployeeRepository1.java**

package com.example.employeemanagementmystem.repository;

import com.example.employeemanagementmystem.entity.\*;

import org.springframework.data.jpa.repository.JpaRepository;

public interface EmployeeRepository1 extends JpaRepository<Employee, Long> {

Employee findByName(String name); // ✅ This enables custom query

}

**TestRunner.java**

package com.example.employeemanagementmystem.runner;

import com.example.employeemanagementmystem.entity.\*;

import com.example.employeemanagementmystem.repository.\*;

import org.springframework.boot.CommandLineRunner;

import org.springframework.stereotype.Component;

*@Component*

public class TestRunner implements CommandLineRunner {

private final EmployeeRepository1 repo;

public TestRunner(EmployeeRepository1 repo) {

this.repo = repo;

}

*@Override*

public void run(String... args) {

// Create

Employee e1 = new Employee();

e1.setName("Ramya");

e1.setDepartment("IT");

e1.setSalary(70000);

repo.save(e1);

// Read

System.***out***.println("All Employees: " + repo.findAll());

// Update

e1.setSalary(75000);

repo.save(e1);

// Custom Query

Employee byName = repo.findByName("Ramya");

System.***out***.println("Found by name: " + byName.getDepartment());

// Delete

repo.deleteById(e1.getId());

System.***out***.println("Deleted Employee");

}

}

OUTPUT:

