SPRING – JPA ASSIGNMENT

Submitted by –

Name : Ritesh Patil

Roll Number : 72288097K

Employee.java

```
@Data
@AllArgsConstructor
@NoArgsConstructor
@Entity
public class Employee {

    @Id
    @Column(name = "emp_id")
    private Integer empId;
    @Column(name = "emp_name")
    private String empName;
    @Column(name = "emp_dept")
    private String dept;
    @Column(name = "base_location")
    private String baseLocation;
    @Column
    private Integer address;

    public static EmployeeDTO createEmployeeDTO(Employee emp) {
        return new EmployeeDTO(emp.getEmpId(), emp.getEmpName(),
emp.getDept(), emp.getBaseLocation(), emp.getAddress());
    }
}
```

EmployeeDTO.java

```
public class EmployeeDTO {

    private Integer empId;
    private String empName;
    private String dept;
    private String baseLocation;
    private Integer address;

    public static Employee createEmployee(EmployeeDTO empDTO) {
        Employee emp = new Employee();
        emp.setEmpId(empDTO.getEmpId());
        emp.setEmpName(empDTO.getEmpName());
        emp.setDept(empDTO.getDept());
        emp.setBaseLocation(empDTO.getBaseLocation());
        emp.setAddress(empDTO.getAddress());
        return emp;
    }
}
```

EmployeeRepository.java

```
public interface EmployeeRepository extends JpaRepository<Employee, Integer>{

}
```

EmployeeService.java

```
public interface EmployeeService {
    public void insertEmployee(EmployeeDTO empDTO);
    public EmployeeDTO searchEmployee(Integer id);
    public List<EmployeeDTO> viewEmployees();
    public void editEmployee(Integer id, String dept);
    public void removeEmployee(Integer id);
}
```

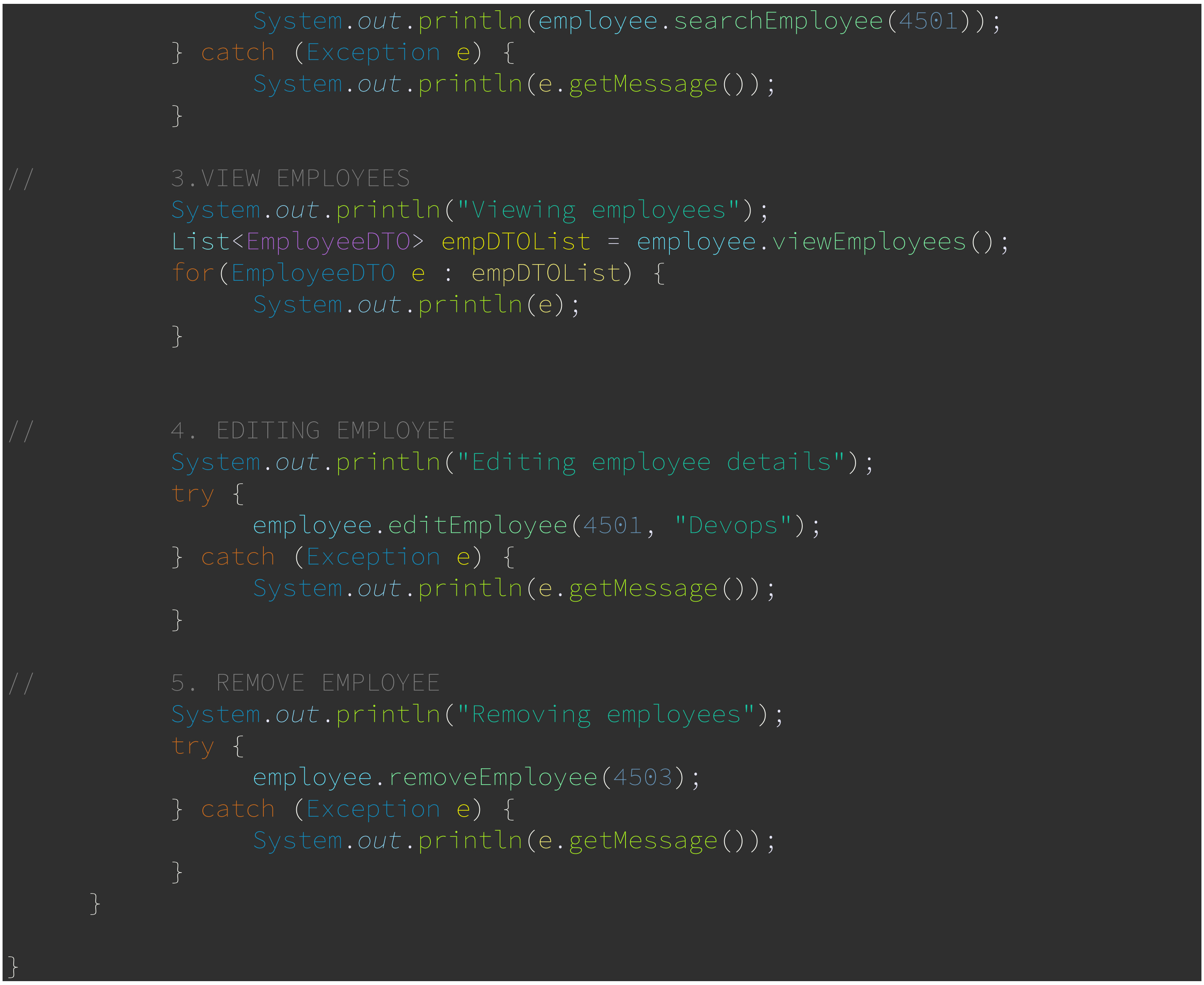
EmployeeServiceImpl.java

```
package com.infosys.orm.service;

import java.util.ArrayList;
import java.util.List;
import java.util.Optional;

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

import com.infosys.orm.entity.Employee;
import com.infosys.orm.entity.EmployeeDTO;
import com.infosys.orm.repository.EmployeeRepository;


@Service
public class EmployeeServiceImpl implements EmployeeService{

    @Autowired
    EmployeeRepository employee;

    @Override
    public void insertEmployee(EmployeeDTO empDTO) {
        Employee emp = EmployeeDTO.createEmployee(empDTO);
        employee.saveAndFlush(emp);
    }

    @Override
    public EmployeeDTO searchEmployee(Integer id) {
        Optional<Employee> emp = employee.findById(id);
        Employee foundEmployee = emp.get();
        return Employee.createEmployeeDTO(foundEmployee);
    }

    @Override
```

```
	public List<EmployeeDTO> viewEmployees() {
		List<EmployeeDTO> empDTOList = new ArrayList<>();
		List<Employee> empList = employee.findAll();
		for(Employee emp : empList) {
			EmployeeDTO empDTO = Employee.createEmployeeDTO(emp);
			empDTOList.add(empDTO);
		}
		return empDTOList;
	}

	@Override
	public void editEmployee(Integer id, String dept) {
		Optional<Employee> emp = employee.findById(id);
		Employee foundEmployee = emp.get();
		foundEmployee.setDept(dept);
		employee.save(foundEmployee);
	}

	@Override
	public void removeEmployee(Integer id) {
		employee.deleteById(id);
	}

}
```

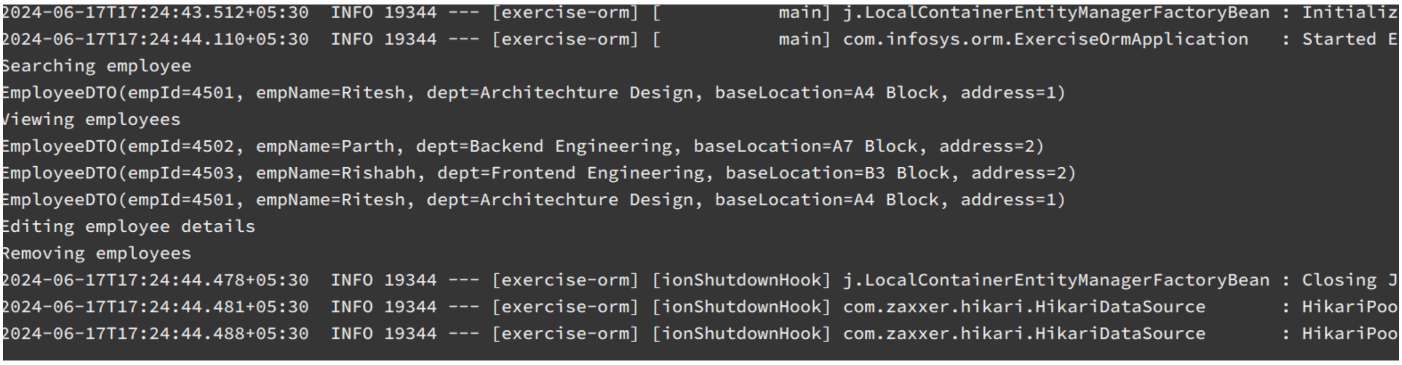
Main.java

```
public class ExerciseOrmApplication implements CommandLineRunner{

	@Autowired
	EmployeeService employee;

	public static void main(String[] args) {
		SpringApplication.run(ExerciseOrmApplication.class, args);
	}

	@Override
	public void run(String... args) throws Exception {

//		1. INSERTING EMPLOYEE
//		System.out.println("Adding employees");
//		EmployeeDTO empDTO = new EmployeeDTO(4501, "Ritesh", "Architechture 
Design", "A4 Block", 1);
//
//		try {
//			employee.insertEmployee(empDTO);
//			System.out.println("Inserted successfully");
//		}catch (Exception e) {
//			System.out.println(e.getMessage());
//		}


//		2. SEARCH EMPLOYEE
		System.out.println("Searching employee");
		try {
```



```
            System.out.println(employee.searchEmployee(4501));
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }

//      3.VIEW EMPLOYEES
        System.out.println("Viewing employees");
        List<EmployeeDTO> empDTOList = employee.viewEmployees();
        for(EmployeeDTO e : empDTOList) {
            System.out.println(e);
        }


//      4. EDITING EMPLOYEE
        System.out.println("Editing employee details");
        try {
            employee.editEmployee(4501, "Devops");
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }

//      5. REMOVE EMPLOYEE
        System.out.println("Removing employees");
        try {
            employee.removeEmployee(4503);
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }
    }

}
```

************************************* OUTPUT **



```
2024-06-17T17:24:43.512+05:30  INFO 19344 --- [exercise-orm] [           main] j.LocalContainerEntityManagerFactoryBean : Initializ
2024-06-17T17:24:44.110+05:30  INFO 19344 --- [exercise-orm] [           main] com.infosys.orm.ExerciseOrmApplication   : Started E
Searching employee
EmployeeDTO(empId=4501, empName=Ritesh, dept=Architechture Design, baseLocation=A4 Block, address=1)
Viewing employees
EmployeeDTO(empId=4502, empName=Parth, dept=Backend Engineering, baseLocation=A7 Block, address=2)
EmployeeDTO(empId=4503, empName=Rishabh, dept=Frontend Engineering, baseLocation=B3 Block, address=2)
EmployeeDTO(empId=4501, empName=Ritesh, dept=Architechture Design, baseLocation=A4 Block, address=1)
Editing employee details
Removing employees
2024-06-17T17:24:44.478+05:30  INFO 19344 --- [exercise-orm] [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing J
2024-06-17T17:24:44.481+05:30  INFO 19344 --- [exercise-orm] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPoo
2024-06-17T17:24:44.488+05:30  INFO 19344 --- [exercise-orm] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPoo
```