**DEPARTMENT OF COMPUTER SCIENCE**

**& ENGINEERING,**

**SHARDA UNIVERSITY**



**COURSE NAME**

**Advanced Java Programming Practical file**

**(CSP014)**

**Name**: **Vinay kumar**

**Semester: IV system id: 2023324785**

**Section: CS-H (Group 2)**

**Faculty: Sushant Jhingran**

**Q1) Implement collection framework interfaces in java.**

*Solution:*

package com.java.collection\_try;

import java.util.\*;

//The collection framework provides a set of classes and interfaces for storing and manipulating groups of data as a single unit.

//It has various data structures like lists, sets, and maps, along with algorithms to manipulate them.

public class Set\_and\_list {

public static void Set() {

Set<String> set = new HashSet<String>();

set.add("America");

set.add("Poland");

set.add("India");

set.add("Sri Lanka");

set.forEach(Desh->System.***out***.print(Desh + ", "));

}

public static void List() {

List<Integer> list = new ArrayList<>();

list.add(100);

list.add(50);

list.add(40);

list.add(30);

list.forEach(num->System.***out***.println(num));

}

public static void main(String[] args) {

*Set*();

*List*();

}

//Output:

// Sri Lanka, America, Poland, India, 100

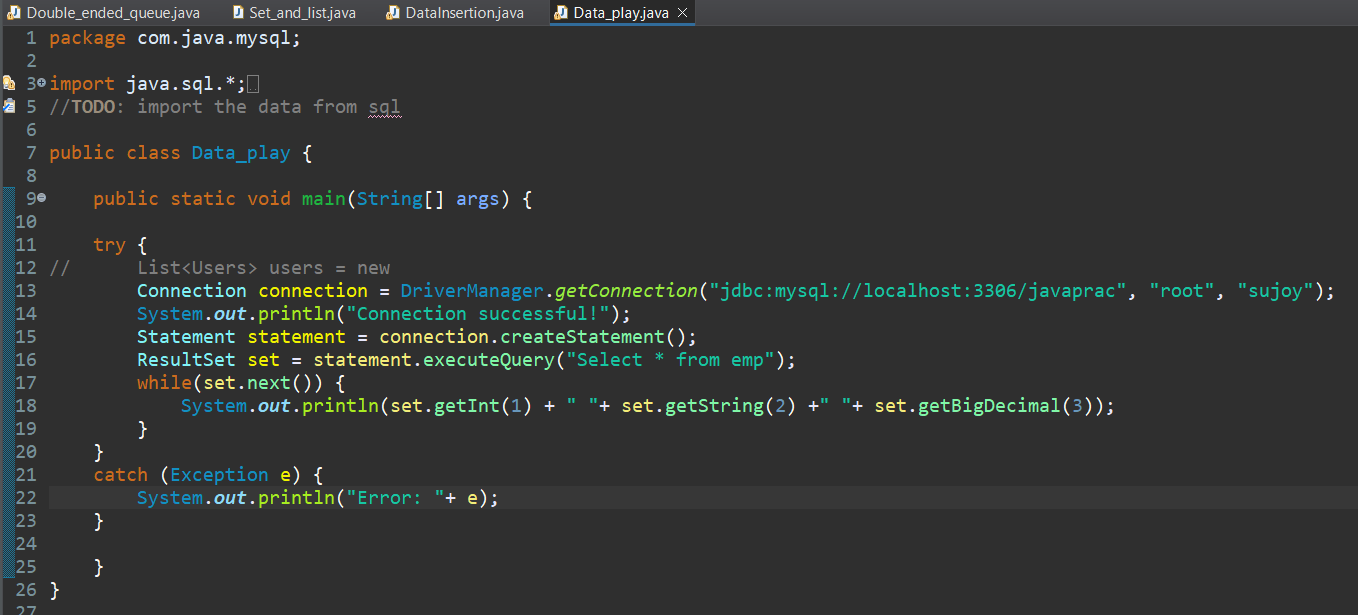
// 50

// 40

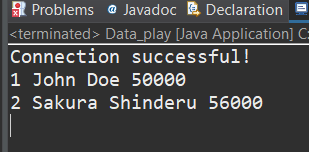
// 30

**Q2) Create program using JDBC connectivity.**

*Solution:*



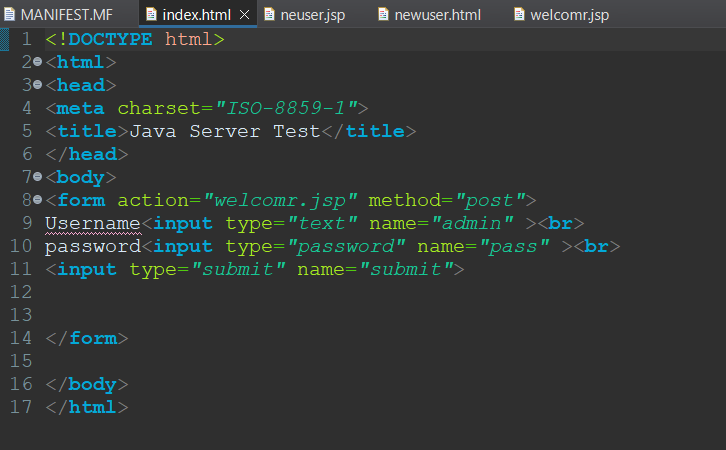
*//Output:*



**Q3) Develop web based program using servlet and JSP.**

*Solution:*

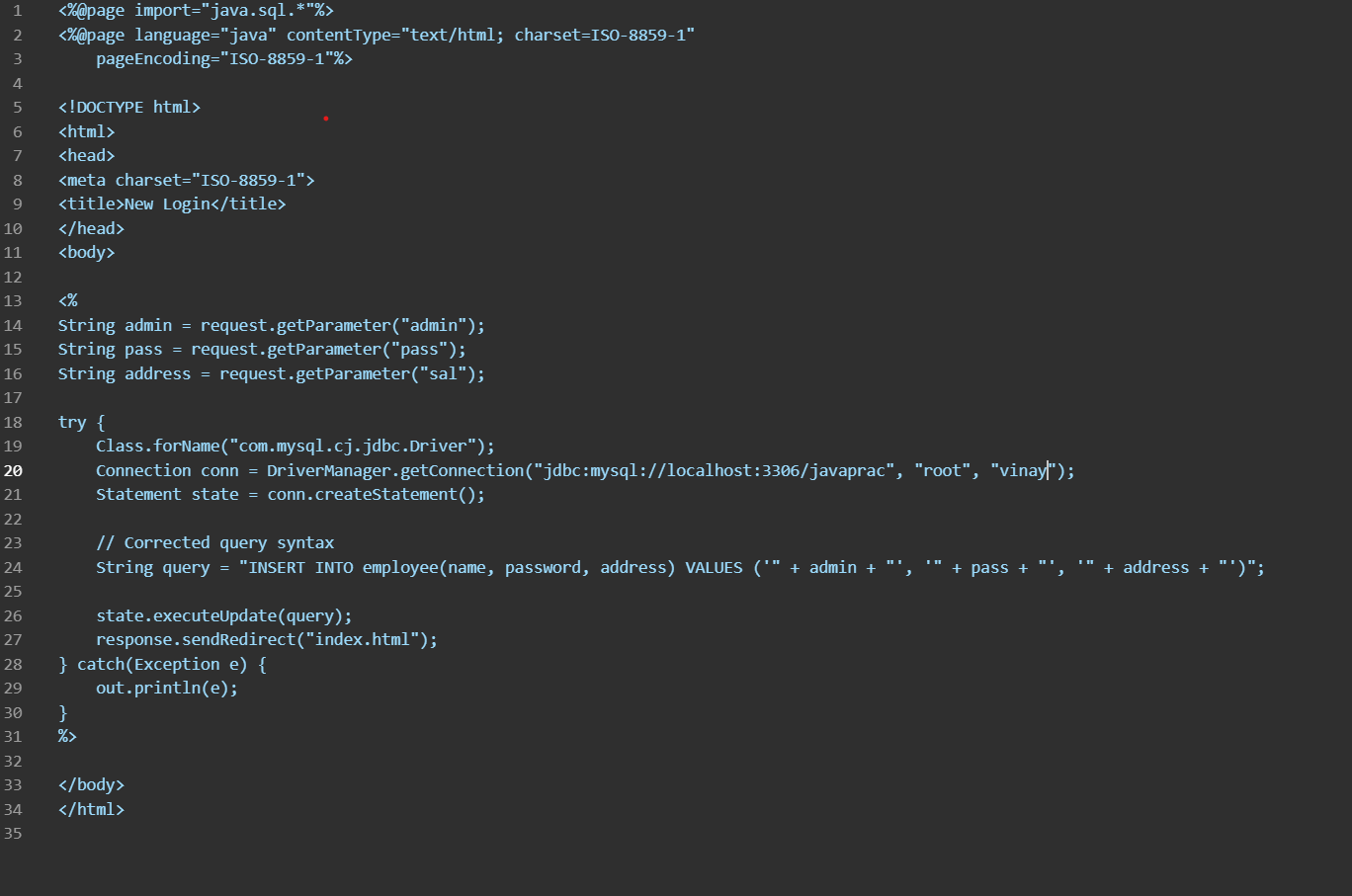
1. *Index.html*



1. *welcomr.jsp*



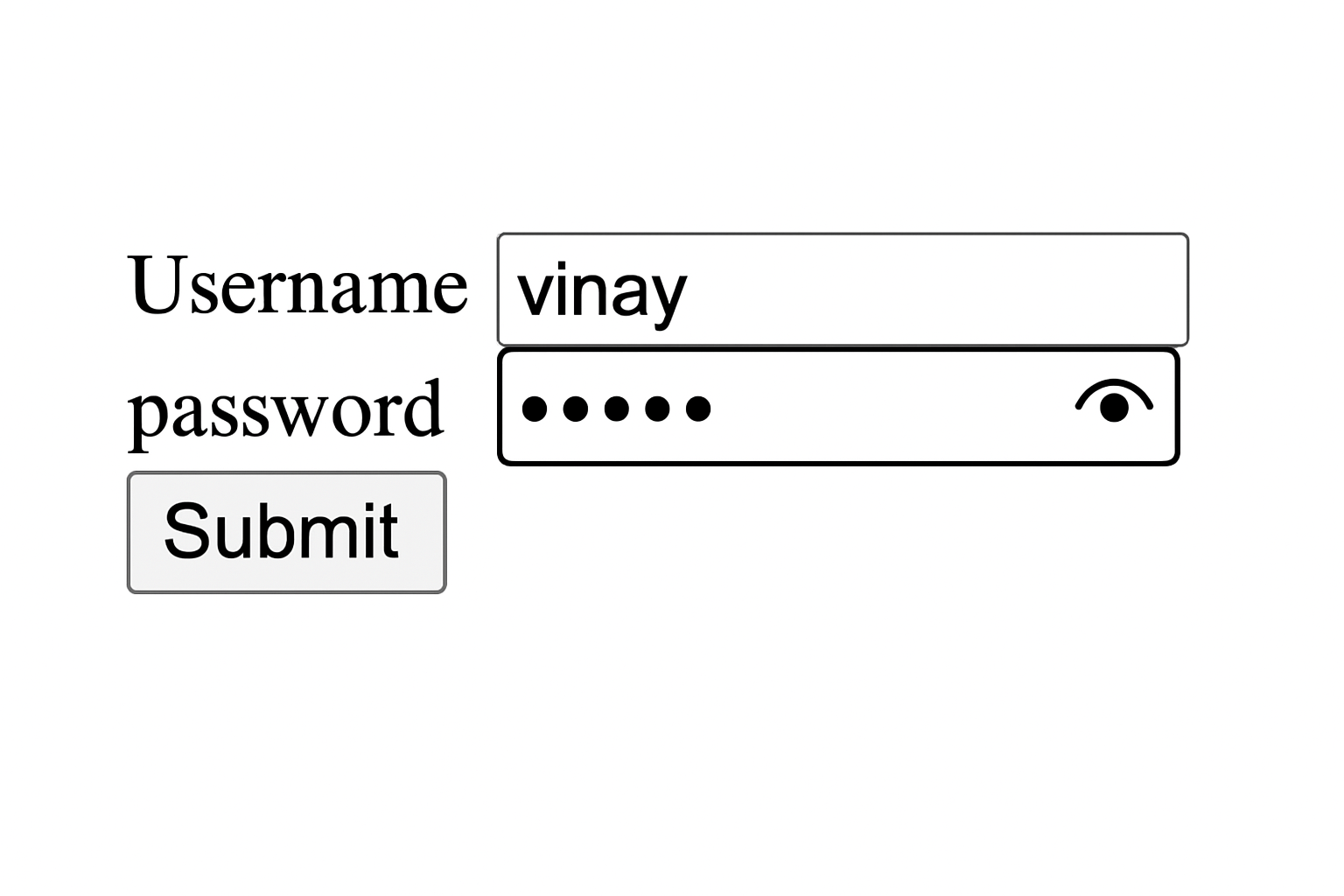
1. *neuser.jsp*

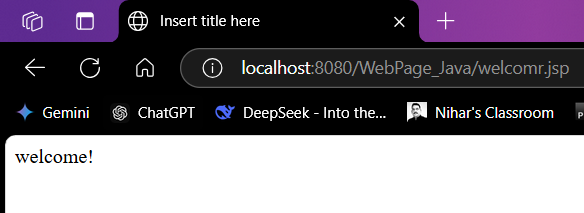


1. *newuser.html*



*//OUTPUT:*

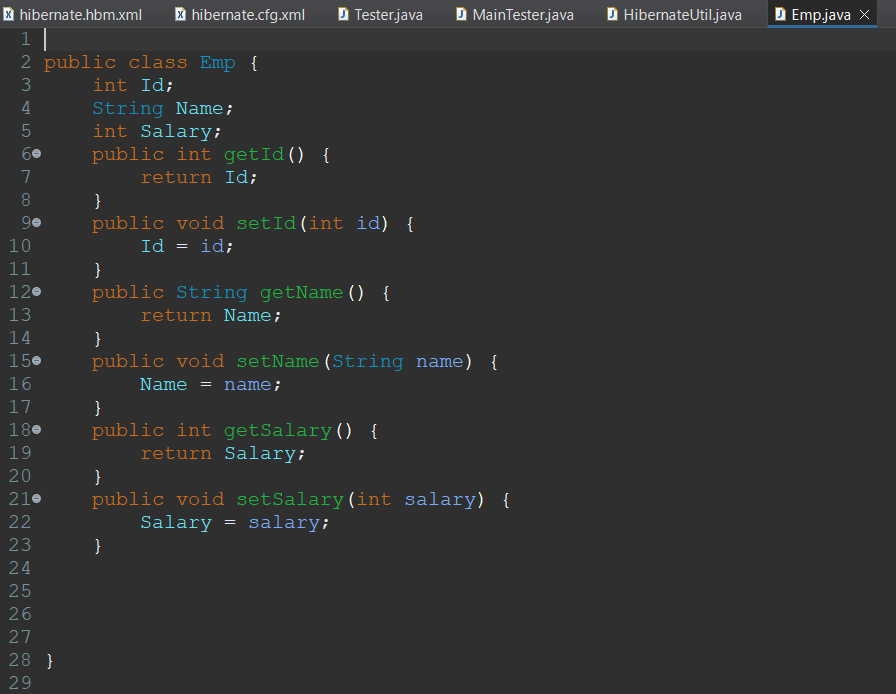


****

**Q4) Implement CRUD operations using JPA and Hibernate.**

*Solution:*

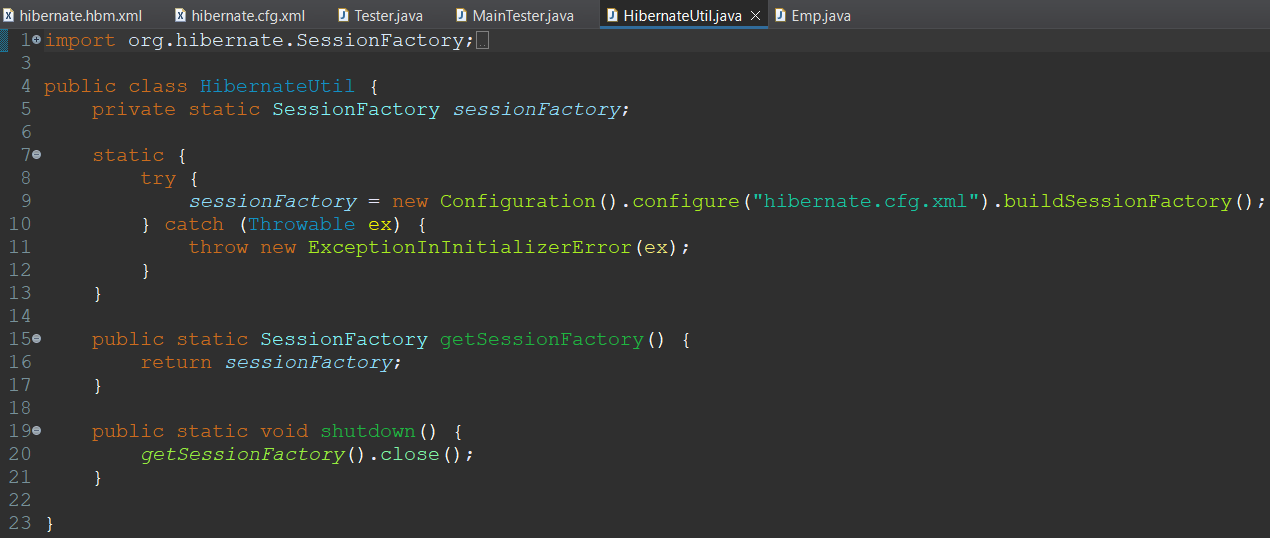
1. *Emp.java*



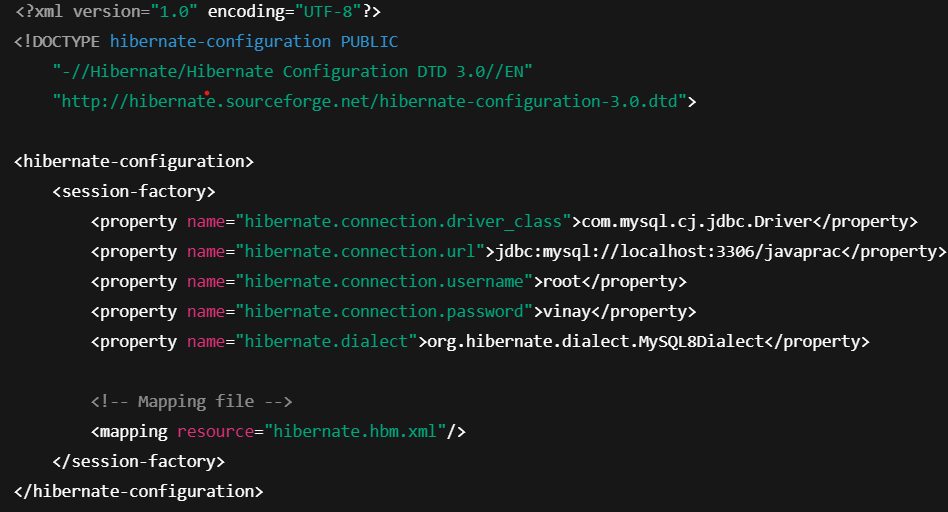
1. *Hibernate.hbm.xml*



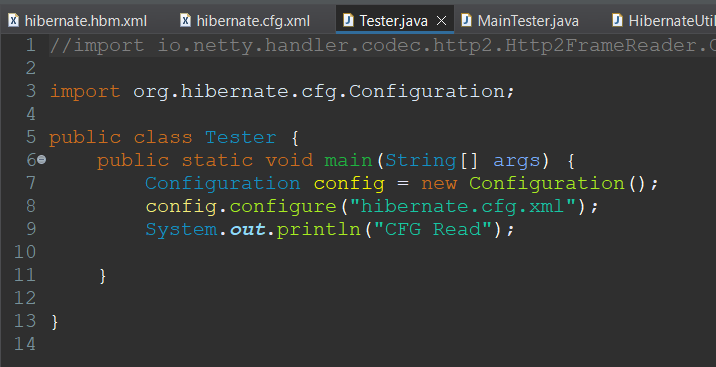
1. *HibernateUtil.java*



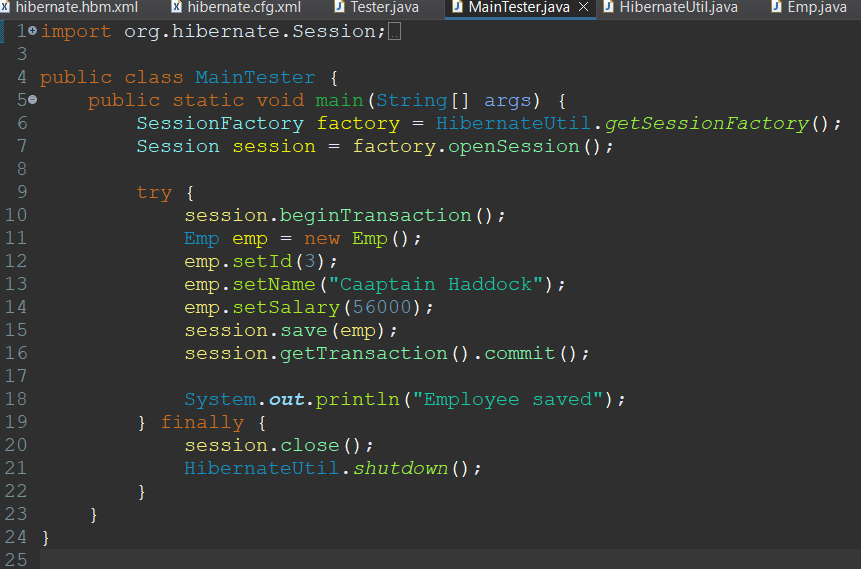
1. *hibernate.cfg.xml*



1. *Tester.java*

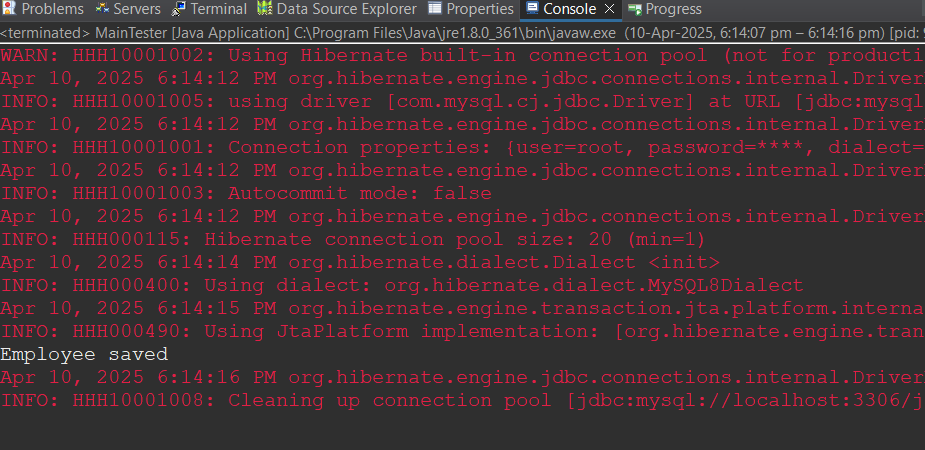


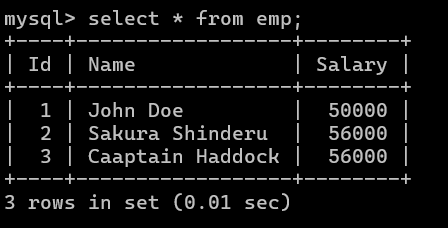
1. *MainTester.java*



1. *pom.xml*
2. <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
3. <modelVersion>4.0.0</modelVersion>
4. <groupId>com.hiber</groupId>
5. <artifactId>hibernate-practice</artifactId>
6. <version>0.0.1-SNAPSHOT</version>
7. <dependencies>
8. <dependency>
9. <groupId>org.hibernate</groupId>
10. <artifactId>hibernate-core</artifactId>
11. <version>5.6.15.Final</version>
12. </dependency>
13. <dependency>
14. <groupId>mysql</groupId>
15. <artifactId>mysql-connector-java</artifactId>
16. <version>8.0.24</version>
17. <scope>runtime</scope>
18. </dependency>
19. </dependencies>
20. </project>

*//OUTPUT:*





**Q5) Implement CRUD operations using SpringBoot.**

Solution:

1. *pom.xml*

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>3.4.4</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.assignment</groupId>

<artifactId>vinay</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>vinay</name>

<description>Spring Boot assignment by Vinay </description>

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

<java.version>21</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-thymeleaf</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

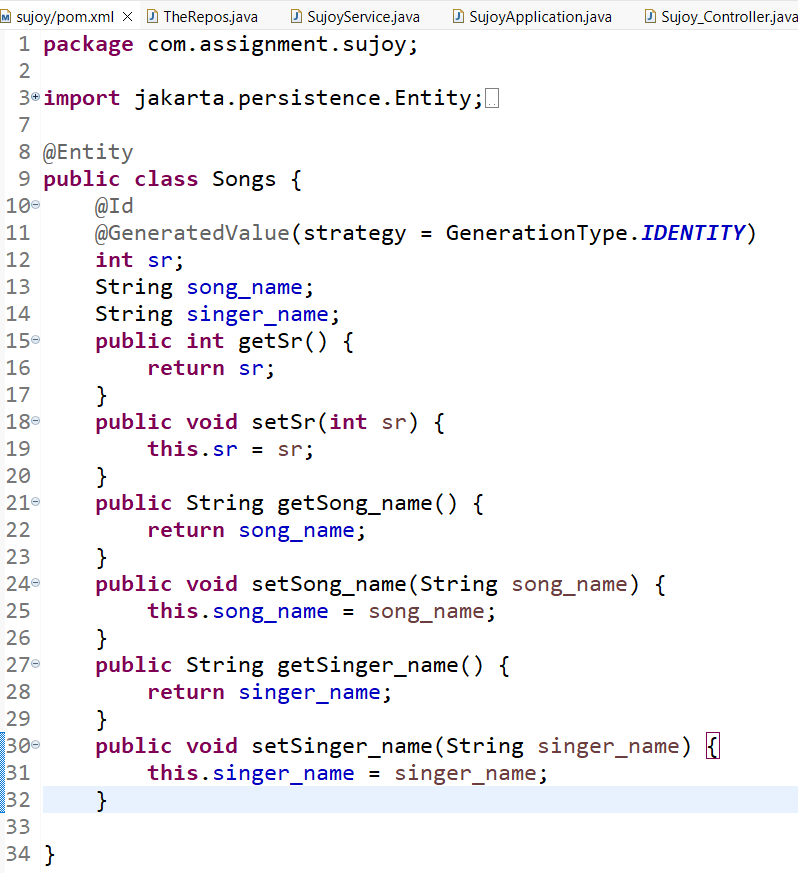
</plugin>

</plugins>

</build>

</project>

1. *Songs.java*



1. *TheRepos.java*

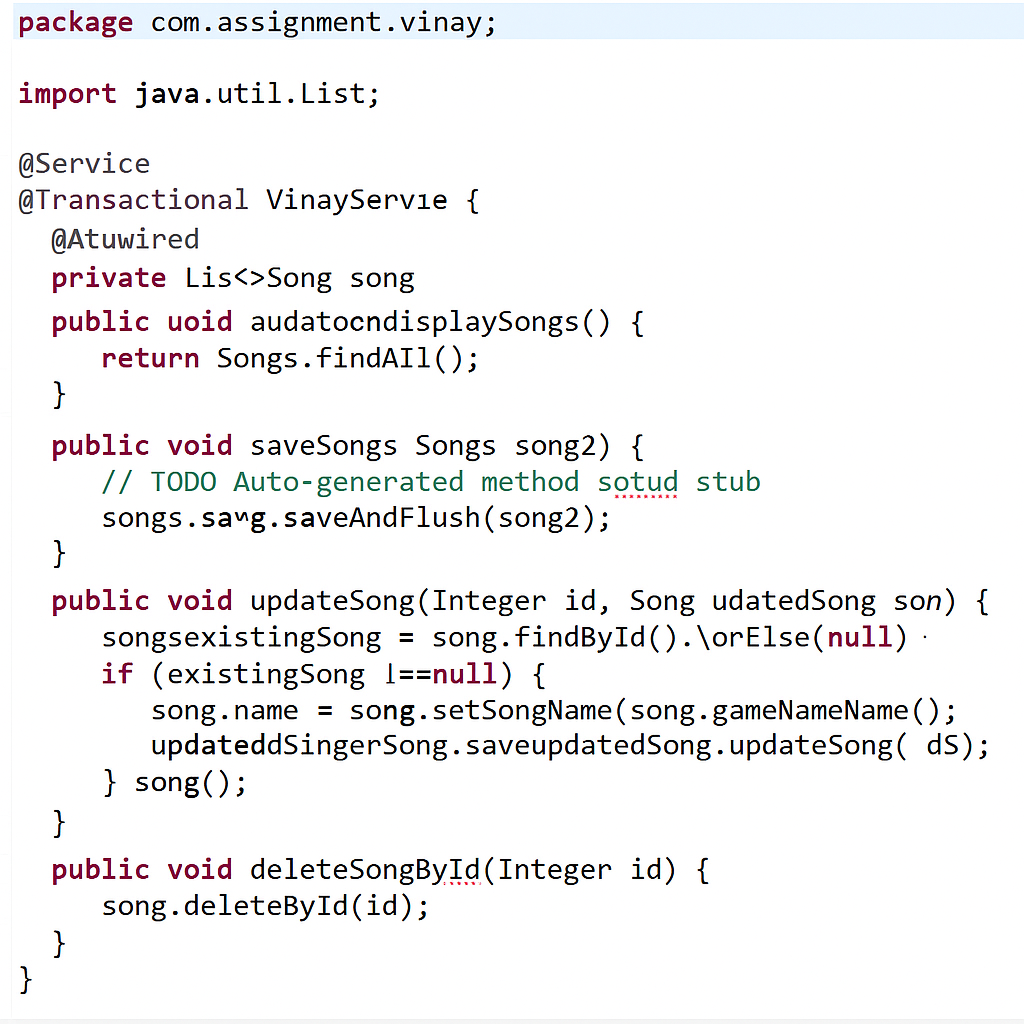
**package** com.assignment.vinay;

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

**public** **interface** TheRepos **extends** JpaRepository<Songs, Integer>{

}

1. *VinayService.java*



1. *VinayApplication.java*

package com.assignment.vinay;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class VinayApplication {

public static void main(String[] args) {

SpringApplication.run(VinayApplication.class, args);

}

}

1. *Vinay\_Controller.java*

package com.assignment.vinay;

import java.util.List;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

@Controller

public class Vinay\_Controller {

@Autowired

VinayService service;

@RequestMapping("/show")

public String show(Model vinayModel) {

List<Songs> list = service.displaySongs();

for(Songs i: list) {

System.out.println(i.getSr()+" "+i.getSong\_name()+" "+i.getSinger\_name());

}

vinayModel.addAttribute("data", list);

return "vinay\_index";

}

@RequestMapping(value = "/addSong", method = RequestMethod.POST)

public String addSongs(@ModelAttribute Songs song)

{

service.saveSongs(song);

return "redirect:/show";

}

@RequestMapping(value = "/updateSong/{sr}", method = RequestMethod.POST)

public String updateSong(@PathVariable Integer sr, @ModelAttribute Songs updatedSong) {

service.updateSong(sr, updatedSong);

return "redirect:/show";

}

@RequestMapping(value = "/deleteSong/{sr}", method = RequestMethod.GET)

public String deleteSong(@PathVariable Integer sr) {

service.deleteSongById(sr);

return "redirect:/show";

}

}

1. *application.properties*

spring.application.name=vinay

server.port = 5567

spring.datasource.url = jdbc:mysql://localhost:3306/javaprac

spring.datasource.username = root

spring.datasource.password = vinay

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.thymeleaf.prefix=classpath:/templates/

spring.thymeleaf.suffix=.html

//OUTPUT:

