Source Code:

Handler:

package com.project.SportyShoes;

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

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

@ControllerAdvice

public class GlobalExceptionHandeler {

@ExceptionHandler(Exception.class)

public String handleException1(Exception e) {

e.printStackTrace();

return "Some error has occurred, please contact to administrator";

}

}

Application:

package com.project.SportyShoes;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SportyShoesApplication {

public static void main(String[] args) {

SpringApplication.run(SportyShoesApplication.class, args);

}

}

Catagories Controler

package com.project.SportyShoes.controller;

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;

import com.project.SportyShoes.entity.Categories;

import com.project.SportyShoes.service.CategoriesService;

@Controller

public class CategoriesController {

@Autowired

private CategoriesService catserv;

@RequestMapping("/listcat")

public String viewListPage(Model model)

{

List<Categories> listCategories= catserv.listAll();

model.addAttribute("listCategories", listCategories);

return "manage";

}

@RequestMapping("/new")

public String addNewCatPage(Model model)

{

Categories categories= new Categories();

model.addAttribute("categories", categories);

return "new\_category";

}

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

public String saveCategory(@ModelAttribute("categories") Categories categories, Model m)

{

catserv.save(categories);

List<Categories> listCategories= catserv.listAll();

m.addAttribute("listCategories", listCategories);

return "manage";

}

@RequestMapping("/delete/{id}")

public String deleteCategory(@PathVariable(name ="id") long id, Model m)

{

catserv.delete(id);

List<Categories> listCategories= catserv.listAll();

m.addAttribute("listCategories", listCategories);

return "manage";

}

}

Login:

package com.project.SportyShoes.controller;

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

import org.springframework.stereotype.Controller;

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

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

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

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

import org.springframework.web.servlet.ModelAndView;

import com.project.SportyShoes.entity.Login;

import com.project.SportyShoes.service.LoginService;

@Controller

public class LoginController {

@Autowired

private LoginService ls;

@GetMapping("/")

public ModelAndView login() {

ModelAndView mav = new ModelAndView("login");

mav.addObject("user", new Login());

return mav;

}

@PostMapping("/adminlogin")

public String signUp(@ModelAttribute("user") Login user ) {

Login oauthUser = null;

System.out.println("from login"+ user);

if(user.getPassword() != "" ) {

System.out.println("logging in");

oauthUser = ls.signgin(user.getUsername(), user.getPassword());

return "menu";

} else {

if (user.getNewPassword() != "" ) {

System.out.println("Changing new password");

oauthUser = ls.changePassword(user.getUsername(), user.getNewPassword());

if(oauthUser == null ) {

return "invalid\_credentials";

}

return "menu";

}

return "invalid\_credentials";

}

}

//@GetMapping("/forgotPassword")

//public String update(@ModelAttribute("user") Login user){

//Login changepw = userService.updatePassword(user.);

//System.out.println(changepw);

//if(Objects.nonNull(changepw))

//{

//return "redirect:/";

//} else {

//return "redirect:/login";

//}

//}

//{

//}

@RequestMapping("/logout")

public String logoutDo()

{

return "conform\_logout";

}

//@PostMapping("/reset")

//public String resetPassword(@ModelAttribute("userchange") Login user) {

//System.out.println("form reset");

//Boolean changePassword = ls.changePassword(user.getUsername(), user.getPassword());

//if(!changePassword) {

// return "usernotfound";

//}

//return "menu";

//}

/\* @GetMapping("/forgotpw")

public String resetPassword1(@ModelAttribute("change") Login change ) {

System.out.println("from login"+ change);

Boolean oauthUser = ls.changePassword(change.getUsername(), change.getPassword());

System.out.print(oauthUser);

if(Objects.nonNull(oauthUser))

{

return "menu";

} else {

return "invalid\_credentials";

}

}

\*/

}

Product:

package com.project.SportyShoes.controller;

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.RequestMapping;

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

import com.project.SportyShoes.entity.Categories;

import com.project.SportyShoes.entity.Product;

import com.project.SportyShoes.service.ProductService;

@Controller

public class ProductController {

@Autowired

public ProductService prodserv;

@RequestMapping("/listprod")

public String viewListProductPage(Model model)

{

List<Product> listProduct= prodserv.listAll();

model.addAttribute("listProduct", listProduct);

return "product\_manage";

}

@RequestMapping("/newproduct")

public String addNewProductPage(Model model)

{

Product product= new Product();

model.addAttribute("product", product);

return "new\_product";

}

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

public String saveCategory(@ModelAttribute("product") Product product, Model md)

{

prodserv.save(product);

List<Product> listProduct= prodserv.listAll();

md.addAttribute("listProduct", listProduct);

return "product\_manage";

}

}

User Details:

package com.project.SportyShoes.controller;

import java.util.ArrayList;

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.GetMapping;

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

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

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

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

import org.springframework.web.servlet.ModelAndView;

import com.project.SportyShoes.entity.UserDetails;

import com.project.SportyShoes.service.UserDetailsService;

@Controller

public class UserDetailsController {

@Autowired

UserDetailsService userservice;

//@GetMapping("/list")

//public List<UserDetails> getUserDetails()

//{

//List<UserDetails> listuser = new ArrayList<UserDetails>();

//listuser= userservice.getUsers();

// return listuser;

//}

@GetMapping("/index")

public String viewUserDetails(Model model)

{

List<UserDetails> listUsers= userservice.listAll();

System.out.println("listUsers" + listUsers);

model.addAttribute("listUsers",listUsers);

return "/index";

}

// @GetMapping("/find")

// public List<UserDetails> findusername(@RequestParam("username") String username)

// {

// List<UserDetails> searchuser =new ArrayList<UserDetails>();

// searchuser = userservice.get(username);

// System.out.println("user"+ searchuser);

// return searchuser;

// }

@GetMapping("/user/{username}")

public String findUserName(@PathVariable(name="username")String username)

{

System.out.println("i am here");

ModelAndView mav = new ModelAndView("find");

List<UserDetails> userdetails=userservice.get(username);

mav.addObject("userdetails", userdetails);

return "finduser";

}

}

**Entity**

**package com.project.SportyShoes.entity;**

**import lombok.Data;**

**import lombok.EqualsAndHashCode;**

**import javax.persistence.Column;**

**import javax.persistence.EntityListeners;**

**import javax.persistence.MappedSuperclass;**

**import java.util.Date;**

**@MappedSuperclass**

**@Data**

**@EqualsAndHashCode**

**@EntityListeners(EntityListener.class)**

**public abstract class BaseEntity {**

**@Column(nullable = false, updatable = false)**

**private Date createDate;**

**@Column(insertable = false)**

**private Date lastUpdateDate;**

**public Date getCreateDate() {**

**return createDate;**

**}**

**public void setCreateDate(Date createDate) {**

**this.createDate = createDate;**

**}**

**public Date getLastUpdateDate() {**

**return lastUpdateDate;**

**}**

**public void setLastUpdateDate(Date lastUpdateDate) {**

**this.lastUpdateDate = lastUpdateDate;**

**}**

**}**

**package com.project.SportyShoes.entity;**

**import javax.persistence.Entity;**

**import javax.persistence.GeneratedValue;**

**import javax.persistence.GenerationType;**

**import javax.persistence.Id;**

**import javax.persistence.Table;**

**@Entity**

**@Table(name="categories")**

**public class Categories {**

**@Id**

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

**private long id;**

**private String categoryName;**

**public long getId() {**

**return id;**

**}**

**public void setId(long id) {**

**this.id = id;**

**}**

**public Categories() {**

**super();**

**// TODO Auto-generated constructor stub**

**}**

**public Categories(long id, String categoryName) {**

**super();**

**this.id = id;**

**this.categoryName = categoryName;**

**}**

**@Override**

**public String toString() {**

**return "Categories [id=" + id + ", categoryName=" + categoryName + "]";**

**}**

**public String getCategoryName() {**

**return categoryName;**

**}**

**public void setCategoryName(String categoryName) {**

**this.categoryName = categoryName;**

**}**

**/\*public String women;**

**public String men;**

**public String kids;**

**public Categories() {**

**super();**

**// TODO Auto-generated constructor stub**

**}**

**public Categories(long id, String women, String men, String kids) {**

**super();**

**this.id = id;**

**this.women = women;**

**this.men = men;**

**this.kids = kids;**

**}**

**public long getId() {**

**return id;**

**}**

**public void setId(long id) {**

**this.id = id;**

**}**

**public String getWomen() {**

**return women;**

**}**

**public void setWomen(String women) {**

**this.women = women;**

**}**

**public String getMen() {**

**return men;**

**}**

**public void setMen(String men) {**

**this.men = men;**

**}**

**public String getKids() {**

**return kids;**

**}**

**public void setKids(String kids) {**

**this.kids = kids;**

**}**

**@Override**

**public String toString() {**

**return "Categories [id=" + id + ", women=" + women + ", men=" + men + ", kids=" + kids + "]";**

**}\*/**

**}**

**package com.project.SportyShoes.entity;**

**import javax.persistence.PrePersist;**

**import javax.persistence.PreUpdate;**

**import java.util.Date;**

**public class EntityListener {**

**@PrePersist**

**public void prePersist(BaseEntity entity) {**

**Date now = new Date();**

**entity.setCreateDate(now);**

**}**

**@PreUpdate**

**public void preUpdate(BaseEntity entity) {**

**Date now = new Date();**

**entity.setLastUpdateDate(now);**

**}**

**}**

**package com.project.SportyShoes.entity;**

**import javax.persistence.Entity;**

**import javax.persistence.GeneratedValue;**

**import javax.persistence.GenerationType;**

**import javax.persistence.Id;**

**import javax.persistence.Table;**

**@Entity**

**@Table(name="product")**

**public class Product {**

**@Id**

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

**private long id;**

**private String productName;**

**private String category;**

**private long price;**

**private Boolean status;**

**private String size;**

**public Product() {**

**super();**

**// TODO Auto-generated constructor stub**

**}**

**public long getId() {**

**return id;**

**}**

**public void setId(long id) {**

**this.id = id;**

**}**

**public String getProductName() {**

**return productName;**

**}**

**public void setProductName(String productName) {**

**this.productName = productName;**

**}**

**public String getCategory() {**

**return category;**

**}**

**public void setCategory(String category) {**

**this.category = category;**

**}**

**public long getPrice() {**

**return price;**

**}**

**public void setPrice(long price) {**

**this.price = price;**

**}**

**public Boolean getStatus() {**

**return status;**

**}**

**public void setStatus(Boolean status) {**

**this.status = status;**

**}**

**public String getSize() {**

**return size;**

**}**

**public void setSize(String size) {**

**this.size = size;**

**}**

**@Override**

**public String toString() {**

**return "Product [id=" + id + ", productName=" + productName + ", category=" + category + ", price=" + price**

**+ ", status=" + status + ", size=" + size + "]";**

**}**

**public Product(long id, String productName, String category, long price, Boolean status, String size) {**

**super();**

**this.id = id;**

**this.productName = productName;**

**this.category = category;**

**this.price = price;**

**this.status = status;**

**this.size = size;**

**}**

**}**

**package com.project.SportyShoes.entity;**

**import java.sql.Date;**

**import javax.persistence.Entity;**

**import javax.persistence.GeneratedValue;**

**import javax.persistence.GenerationType;**

**import javax.persistence.Id;**

**import javax.persistence.Table;**

**@Entity(name = "userdetails")**

**@Table(name="userdetails")**

**public class UserDetails extends BaseEntity {**

**@Id**

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

**public long userId;**

**public String username;**

**public String emailId;**

**public String phone;**

**public String productPerchased;**

**public Date perchaseDate;**

**public String category;**

**public UserDetails() {**

**super();**

**// TODO Auto-generated constructor stub**

**}**

**public UserDetails(long userId, String username, String emailId, String phone, String productPerchased,**

**Date perchaseDate, String category) {**

**super();**

**this.userId = userId;**

**this.username = username;**

**this.emailId = emailId;**

**this.phone = phone;**

**this.productPerchased = productPerchased;**

**this.perchaseDate = perchaseDate;**

**this.category = category;**

**}**

**public long getUserId() {**

**return userId;**

**}**

**public void setUserId(long userId) {**

**this.userId = userId;**

**}**

**public String getUsername() {**

**return username;**

**}**

**public void setUsername(String username) {**

**this.username = username;**

**}**

**public String getEmailId() {**

**return emailId;**

**}**

**public void setEmailId(String emailId) {**

**this.emailId = emailId;**

**}**

**public String getPhone() {**

**return phone;**

**}**

**public void setPhone(String phone) {**

**this.phone = phone;**

**}**

**public String getProductPerchased() {**

**return productPerchased;**

**}**

**public void setProductPerchased(String productPerchased) {**

**this.productPerchased = productPerchased;**

**}**

**public Date getPerchaseDate() {**

**return perchaseDate;**

**}**

**public void setPerchaseDate(Date perchaseDate) {**

**this.perchaseDate = perchaseDate;**

**}**

**public String getCategory() {**

**return category;**

**}**

**public void setCategory(String category) {**

**this.category = category;**

**}**

**@Override**

**public String toString() {**

**return "UserDetails [userId=" + userId + ", username=" + username + ", emailId=" + emailId + ", phone=" + phone**

**+ ", productPerchased=" + productPerchased + ", perchaseDate=" + perchaseDate + ", category=" + category**

**+ "]";**

**}**

**}**

**jdbc:**

**package com.project.SportyShoes.jdbc;**

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

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

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

**import org.springframework.data.repository.query.Param;**

**import org.springframework.stereotype.Repository;**

**import com.project.SportyShoes.entity.Login;**

**import lombok.val;**

**@Repository**

**public interface LoginRepository extends JpaRepository<Login, Long> {**

**Login findByUsernameAndPassword(String username, String password);**

**Login findByUsername(String username);**

**//Login findByEmail(String emailId);**

**//Login findByResetPassword(String resetPassword);**

**@Modifying**

**@Query("update Login log set log.password = :password where log.id = :id")**

**void changePassword(@Param(value="password") String password, @Param(value = "id") Long id);**

**}**

**package com.project.SportyShoes.jdbc;**

**import java.util.List;**

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

**import org.springframework.stereotype.Repository;**

**import com.project.SportyShoes.entity.UserDetails;**

**@Repository**

**public interface UserDetailsRepository extends JpaRepository<UserDetails, Long> {**

**public List<UserDetails> findByusername(String username);**

**}**

**package com.project.SportyShoes.service;**

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

**import org.springframework.stereotype.Service;**

**import com.project.SportyShoes.entity.Login;**

**import com.project.SportyShoes.entity.UserDetails;**

**import com.project.SportyShoes.jdbc.LoginRepository;**

**@Service**

**public class LoginService{**

**@Autowired**

**private LoginRepository repo;**

**public Login signgin(String username, String password) {**

**Login user = repo.findByUsernameAndPassword(username, password);**

**System.out.println("from service" + user);**

**return user;**

**}**

**public Login changePassword(String username, String newpassword) {**

**// TODO Auto-generated method stub**

**Login uplogin = null;**

**Login user1 = repo.findByUsername(username);**

**if(user1 !=null ) {**

**System.out.println("user1" + user1);**

**repo.changePassword(newpassword, user1.getId());**

**uplogin = repo.findByUsername(username);**

**System.out.println("Password Changed");**

**return uplogin;**

**}**

**return null;**

**}**

**}**

**// Login login=repo.findByEmail(email);**

**// if(login != null)**

**//{**

**// login.setResetPassword(resetPassword);**

**//repo.save(login);**

**//}**

**//else {**

**//System.out.println("couldnot find the emailid");**

**//}**

**// }**

**//catch (Exception e) {**

**//e.printStackTrace(e.getMessage());**

**//}**

**//}**

**//public Login getByResetPassword(String resetPassword)**

**//{**

**// return repo.findByResetPassword(resetPassword);**

**//}**

**//}**

**package com.project.SportyShoes.service;**

**import java.util.ArrayList;**

**import java.util.List;**

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

**import org.springframework.stereotype.Service;**

**import com.project.SportyShoes.entity.UserDetails;**

**import com.project.SportyShoes.jdbc.UserDetailsRepository;**

**@Service**

**public class UserDetailsService {**

**@Autowired**

**UserDetailsRepository userRepo;**

**//public List<UserDetails> listAll(){**

**//List<UserDetails> ud=new ArrayList<UserDetails>();**

**//ud= userRepo.findAll();**

**//System.out.println("userdetails "+ud);**

**//return ud;**

**//}**

**public List<UserDetails> listAll()**

**{**

**return userRepo.findAll();**

**}**

**public List<UserDetails> get(String username) {**

**return userRepo.findByusername(username);**

**}**

**/\*public List<UserDetails> searchUsers(String username) {**

**List<UserDetails> su= new ArrayList<UserDetails>();**

**su= userRepo.findByusername(username);**

**return su;**

**}\*/**

**}**

**Pom**

<?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*](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>2.5.1</version>

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

</parent>

<groupId>com.project</groupId>

<artifactId>SportyShoes</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>SportyShoes</name>

<description>phase3 project for Spring Boot</description>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

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

<artifactId>spring-boot-starter-data-jdbc</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-starter-webflux</artifactId>

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>javax.persistence</artifactId>

<version>2.1.0</version>

<scope>compile</scope>

</dependency>

<dependency>

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

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

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

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

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

<scope>test</scope>

</dependency>

<dependency>

<groupId>io.projectreactor</groupId>

<artifactId>reactor-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

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

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

</dependency>

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<optional>true</optional>

</dependency>

<dependency>

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

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

</dependency>

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<scope>runtime</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>