AjouterChambre.java

package controller;

import dao.GestionDaoImpl;

import model.Chambre;

import model.Hotel;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.io.IOException;

public class AjouterChambre extends HttpServlet {

private GestionDaoImpl gestionDao;

@Override

public void init() {

gestionDao = new GestionDaoImpl();

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

try {

String titre = request.getParameter("titre");

int nbreLits = Integer.parseInt(request.getParameter("nbreLits"));

int hotelId = Integer.parseInt(request.getParameter("hotelId"));

Hotel hotel = gestionDao.getHotelById(hotelId);

if (hotel == null) {

request.setAttribute("errorMessage", "Hotel with ID " + hotelId + " not found.");

request.getRequestDispatcher("errorPage.jsp").forward(request, response);

return;

}

Chambre chambre = new Chambre();

chambre.setTitre(titre);

chambre.setNbLits(nbreLits);

chambre.setHotel(hotel);

gestionDao.addChambre(chambre);

response.sendRedirect("ListeChambres?hotelId="+ hotelId);

} catch (NumberFormatException e) {

request.setAttribute("errorMessage", "Invalid number format for room or hotel ID.");

request.getRequestDispatcher("errorPage.jsp").forward(request, response);

} catch (Exception e) {

e.printStackTrace();

request.setAttribute("errorMessage", "An error occurred while adding the room.");

request.getRequestDispatcher("errorPage.jsp").forward(request, response);

}

}

}

AvantAjouterChambre

package controller;

import dao.GestionDaoImpl;

import model.Hotel;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.io.IOException;

import java.util.List;

public class AvantAjouterChambre extends HttpServlet {

private GestionDaoImpl gestionDao;

@Override

public void init() {

gestionDao = new GestionDaoImpl();

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

try {

List<Hotel> hotels = gestionDao.getAllHotels();

if (hotels == null || hotels.isEmpty()) {

request.setAttribute("errorMessage", "No hotels found in the database.");

} else {

request.setAttribute("hotels", hotels);

}

RequestDispatcher dispatcher = request.getRequestDispatcher("/formAjout.jsp");

dispatcher.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

request.setAttribute("errorMessage", "An error occurred while fetching the hotels.");

RequestDispatcher dispatcher = request.getRequestDispatcher("/errorPage.jsp");

dispatcher.forward(request, response);

}

}

}

ListeChambres

package controller;

import dao.GestionDaoImpl;

import model.Chambre;

import model.Hotel;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.io.IOException;

import java.util.List;

public class ListeChambres extends HttpServlet {

private GestionDaoImpl gestionDao;

@Override

public void init() {

gestionDao = new GestionDaoImpl();

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

try {

int hotelId = Integer.parseInt(request.getParameter("hotelId"));

List<Chambre> chambres = gestionDao.getChambresByHotelId(hotelId);

System.out.println("Hotel ID: " + hotelId);

System.out.println("Number of rooms fetched: " + (chambres != null ? chambres.size() : 0));

if (chambres != null && !chambres.isEmpty()) {

request.setAttribute("chambres", chambres);

request.setAttribute("hotelName", gestionDao.getHotelById(hotelId)); // Optional: Add hotel name for display

RequestDispatcher dispatcher = request.getRequestDispatcher("/chambres.jsp");

dispatcher.forward(request, response);

} else {

request.setAttribute("message", "No rooms found for this hotel.");

RequestDispatcher dispatcher = request.getRequestDispatcher("/chambres.jsp");

dispatcher.forward(request, response);

}

} catch (NumberFormatException e) {

response.getWriter().println("Invalid hotel ID format");

} catch (Exception e) {

e.printStackTrace(); // Log or handle exceptions as needed

response.getWriter().println("Error occurred while fetching rooms");

}

}

private void forwardToErrorPage(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

RequestDispatcher dispatcher = request.getRequestDispatcher("/errorPage.jsp");

dispatcher.forward(request, response);

}

}

Principale

package controller;

import dao.GestionDaoImpl;

import model.Hotel;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.io.IOException;

import java.util.List;

public class Principale extends HttpServlet {

private static final long serialVersionUID = 1L;

private GestionDaoImpl gestionDao;

@Override

public void init() throws ServletException {

// Initialize the GestionDaoImpl

gestionDao = new GestionDaoImpl();

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

try {

List<Hotel> hotels = gestionDao.getAllHotels();

request.setAttribute("hotels", hotels);

RequestDispatcher dispatcher = request.getRequestDispatcher("/home.jsp");

dispatcher.forward(request, response);

} catch (Exception e) {

e.printStackTrace();

request.setAttribute("errorMessage", "An error occurred while fetching the hotel list.");

RequestDispatcher dispatcher = request.getRequestDispatcher("/errorPage.jsp");

dispatcher.forward(request, response);

}

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

doGet(request, response);

}

}

GestionDaoImp

package dao;

import java.util.List;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.boot.Metadata;

import org.hibernate.boot.MetadataSources;

import org.hibernate.boot.registry.StandardServiceRegistry;

import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

import org.hibernate.query.Query;

import model.Chambre;

import model.Hotel;

public class GestionDaoImpl implements IGestionDao {

private SessionFactory sessionFactory;

public GestionDaoImpl() {

try {

StandardServiceRegistry serviceRegistry = new StandardServiceRegistryBuilder()

.configure("hibernate.cfg.xml")

.build();

Metadata metadata = new MetadataSources(serviceRegistry).getMetadataBuilder().build();

sessionFactory = metadata.getSessionFactoryBuilder().build();

} catch (Exception e) {

e.printStackTrace();

throw new RuntimeException("Failed to initialize Hibernate SessionFactory.");

}

}

@Override

public void addHotel(Hotel hotel) {

try (Session session = sessionFactory.openSession()) {

Transaction transaction = session.beginTransaction();

session.save(hotel);

transaction.commit();

} catch (Exception e) {

e.printStackTrace();

}

}

@Override

public Hotel getHotelById(int id) {

try (Session session = sessionFactory.openSession()) {

return session.get(Hotel.class, id);

}

}

@Override

public List<Hotel> getAllHotels() {

try (Session session = sessionFactory.openSession()) {

Query<Hotel> query = session.createQuery("from Hotel", Hotel.class);

return query.list();

}

}

@Override

public void addChambre(Chambre chambre) {

try (Session session = sessionFactory.openSession()) {

Transaction transaction = session.beginTransaction();

session.save(chambre);

transaction.commit();

} catch (Exception e) {

e.printStackTrace();

}

}

@Override

public List<Chambre> getChambresByHotelId(int hotelId) {

try (Session session = sessionFactory.openSession()) {

Query<Chambre> query = session.createQuery("from Chambre where hotel.id = :hotelId", Chambre.class);

query.setParameter("hotelId", hotelId);

return query.list();

}

}

}

IGestionDao

package dao;

import java.util.List;

import model.Chambre;

import model.Hotel;

public interface IGestionDao {

public void addHotel(Hotel hotel);

public Hotel getHotelById(int id);

public List<Hotel> getAllHotels();

public void addChambre(Chambre chambre);

public List<Chambre> getChambresByHotelId(int hotelId);

}

Chambre

package model;

import javax.persistence.\*;

*@Entity*

public class Chambre {

*@Id*

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

private int id;

private String titre;

private int nbLits;

*@ManyToOne*

*@JoinColumn*(name = "hotel\_id")

private Hotel hotel;

// Default constructor

public Chambre() {

}

public Chambre(Hotel hotel, String titre, int nombreDeLits) {

this.hotel = hotel;

this.titre = titre;

this.nbLits = nbLits;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getTitre() {

return titre;

}

public void setTitre(String titre) {

this.titre = titre;

}

public int getNbLits() {

return nbLits;

}

public void setNbLits(int nbLits) {

this.nbLits = nbLits;

}

public Hotel getHotel() {

return hotel;

}

public void setHotel(Hotel hotel) {

this.hotel = hotel;

}

}

Hotel

package model;

import java.util.List;

import javax.persistence.\*;

@Entity

public class Hotel {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String titre;

private String lieu;

@OneToMany(mappedBy = "hotel", cascade = CascadeType.ALL)

private List<Chambre> chambre;

public Hotel() {

}

public Hotel(String tite, String lieu) {

this.titre = titre;

this.lieu = lieu;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getTitre() {

return titre;

}

public void setTitre(String titre) {

this.titre = titre;

}

public String getLieu() {

return lieu;

}

public void setLieu(String lieu) {

this.lieu = lieu;

}

public List<Chambre> getChambres() {

return chambre;

}

public void setChambres(List<Chambre> chambres) {

this.chambre = chambres;

}

@OneToMany(mappedBy = "hotel", cascade = CascadeType.ALL)

private List<Chambre> chambres;

}

Chmabres.jsp

<%@ **page** language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ **taglib** prefix=*"c"* uri=*"*[*http://java.sun.com/jsp/jstl/core*](http://java.sun.com/jsp/jstl/core)*"* %>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Room List</**title**>

</**head**>

<**body**>

<**h1**>Rooms in ${hotelName.titre}</**h1**>

<**table** border=*"1"*>

<**thead**>

<**tr**>

<**th**>titre chambre</**th**>

<**th**>nombre de lits</**th**>

</**tr**>

</**thead**>

<**tbody**>

<**c:forEach** var=*"chambre"* items=*"*${chambres}*"*>

<**tr**>

<**td**>${chambre.titre}</**td**>

<**td**>${chambre.nbLits}</**td**>

</**tr**>

</**c:forEach**>

</**tbody**>

</**table**>

<**br**>

<**a** href=*"AvantAjouterChambre"*>Add New Room</**a**>

<**br**>

<**a** href=*"Principale"*>retourne Hotels</**a**>

</**body**>

</**html**>

formAjout.jsp

<%@ **page** language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ **taglib** prefix=*"c"* uri=*"*[*http://java.sun.com/jsp/jstl/core*](http://java.sun.com/jsp/jstl/core)*"* %>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Add New Room</**title**>

</**head**>

<**body**>

<**h1**>Add New Room</**h1**>

<**form** action=*"AjouterChambre"* method=*"post"*>

<**label** for=*"titre"*>titre chambre:</**label**>

<**input** type=*"text"* id=*"titre"* name=*"titre"* required><**br**><**br**>

<**label** for=*"nbreLits"*>nbr lits:</**label**>

<**input** type=*"number"* id=*"nbreLits"* name=*"nbreLits"* required><**br**><**br**>

<**label** for=*"hotelId"*>Select Hotel:</**label**>

<**select** id=*"hotelId"* name=*"hotelId"* required>

<**c:forEach** var=*"hotel"* items=*"*${hotels}*"*>

<**option** value=*"*${[hotel.id](http://hotel.id/)}*"*>${hotel.titre}</**option**>

</**c:forEach**>

</**select**><**br**><**br**>

<**button** type=*"submit"*>Add chambre</**button**>

</**form**>

<**br**>

<**a** href=*"Principale"*>retourne Hotels</**a**>

</**body**>

</**html**>

home.jsp

<%@ **page** language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ **taglib** prefix=*"c"* uri=*"*[*http://java.sun.com/jsp/jstl/core*](http://java.sun.com/jsp/jstl/core)*"* %>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Hotel List</**title**>

</**head**>

<**body**>

<**h1**>Gestion des hotels</**h1**>

<**h2**>List of Hotels</**h2**>

<**table** border=*"1"*>

<**thead**>

<**tr**>

<**th**>Hotel nom</**th**>

<**th**>Localisation</**th**>

<**th**>Details</**th**>

</**tr**>

</**thead**>

<**tbody**>

<**c:forEach** var=*"hotel"* items=*"*${hotels}*"*>

<**tr**>

<**td**>${hotel.titre}</**td**>

<**td**>${hotel.lieu}</**td**>

<**td**><**a** href=*"ListeChambres?hotelId=*${[hotel.id](http://hotel.id/)}*"*>View Rooms</**a**></**td**>

</**tr**>

</**c:forEach**>

</**tbody**>

</**table**>

<**br**>

<**a** href=*"AvantAjouterChambre"*>Add chambre</**a**>

</**body**>

</**html**>

Pom.xml

<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 http://maven.apache.org/maven-v4\_0\_0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>tn.enig</groupId>

<artifactId>AppJee2</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>ProjetMaven2-Hib Maven Webapp</name>

<url>http://maven.apache.org</url>

<dependencies>

<!-- Servlet Spec -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>servlet-api</artifactId>

<version>2.5</version>

<scope>compile</scope>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>javax.servlet.jsp-api</artifactId>

<version>2.3.1</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>jstl</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

</dependency>

<!-- accès base de données -->

<dependency>

<groupId>mysql</groupId>

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

<version>8.0.28</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.6.10.Final</version>

</dependency>

</dependencies>

<build>

<finalName>ProjetMaven2-Hib</finalName>

</build>

</project>