**Assignment 04**

# Movie Ticket Management System using Hibernate PersistenceTools Required

**Tools Required**

* **Eclipse IDE** (EE version preferred)
* **MySQL Database** (or any RDBMS)
* **Apache Tomcat** (Optional if using JSP/Servlets)
* **Hibernate ORM** (JARs or via Maven)
* **JDBC Driver** (MySQL Connector JAR)

**Step 1: Create MySQL Table**

sql

CopyEdit

CREATE DATABASE moviedb;

USE moviedb;

CREATE TABLE ticket (

id INT PRIMARY KEY AUTO\_INCREMENT,

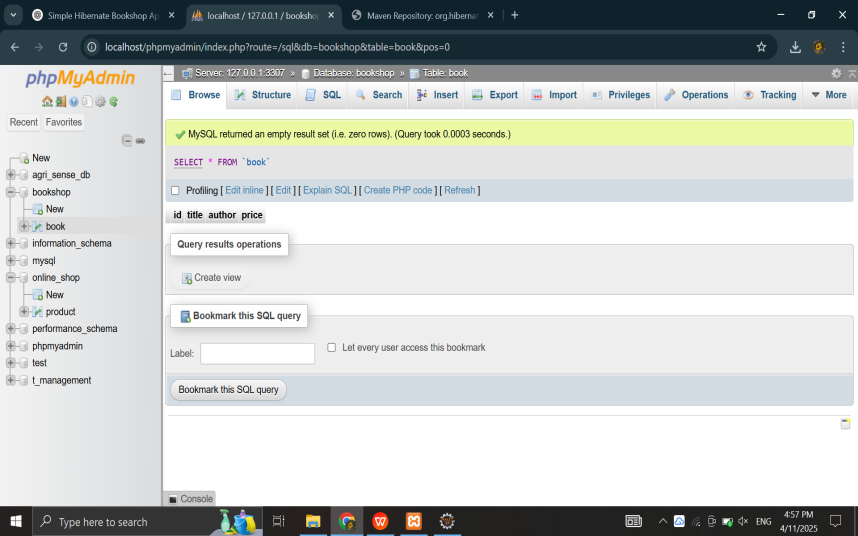
customer\_name VARCHAR(100),

movie\_name VARCHAR(100),

seat\_number VARCHAR(10),

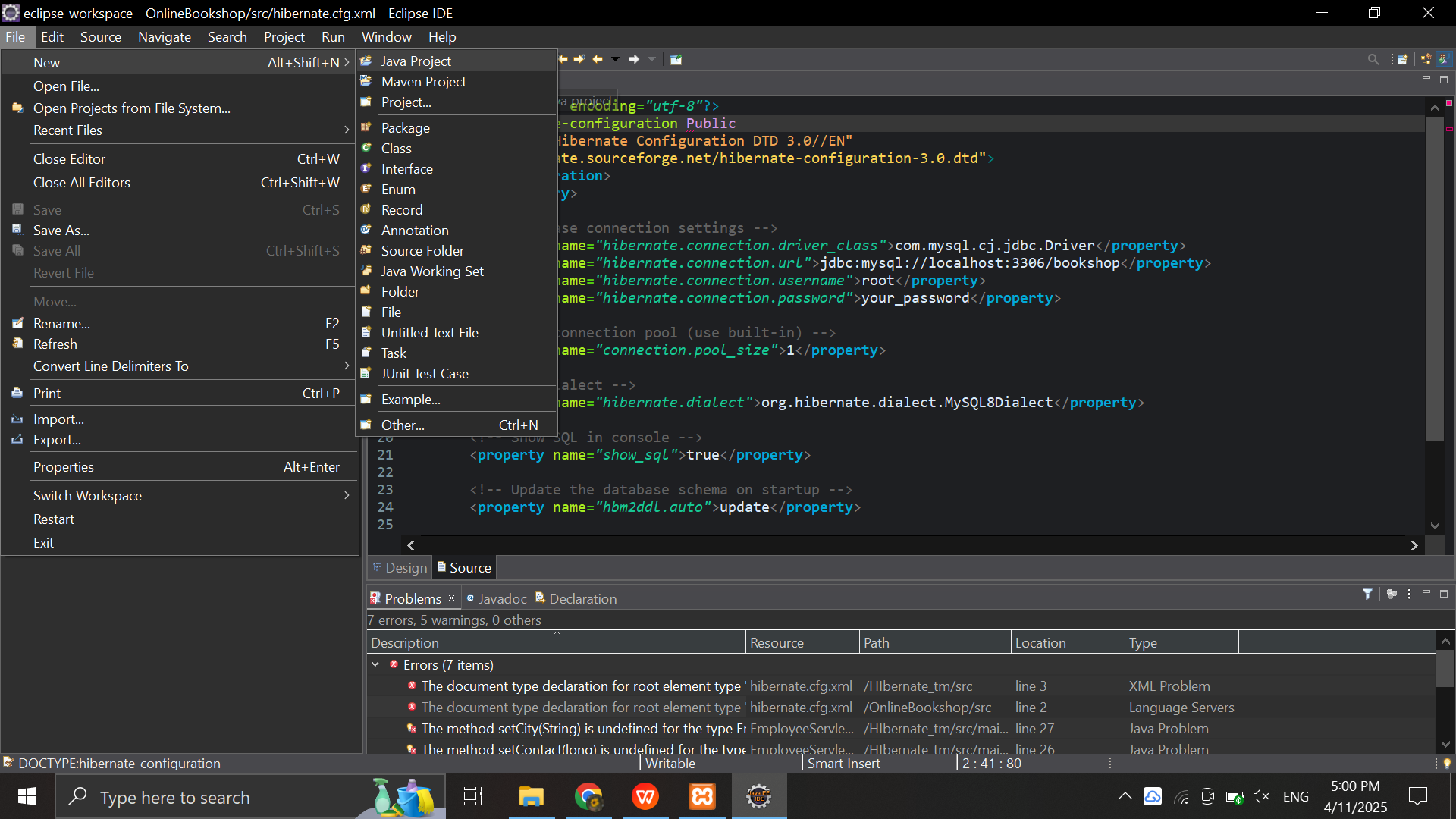
show\_date DATE

);



**Step 2: Set Up Eclipse Project**

* **Open Eclipse → File > New > Java Project → Name it MovieTicketSystem**
* **Right-click on project → New > Package → Name it com.movie**
* **Add required Hibernate and MySQL JARs to the lib folder and include in build path:**
  + **hibernate-core-x.x.x.Final.jar**
  + **mysql-connector-java-x.x.x.jar**



# Step 3: Create Hibernate Configuration File

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

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<property name="hibernate.connection.driver\_class">com.mysql.cj.jdbc.Driver</property>

<property name="hibernate.connection.url">jdbc:mysql://localhost:3306/moviedb</property>

<property name="hibernate.connection.username">root</property>

<property name="hibernate.connection.password">your\_password</property>

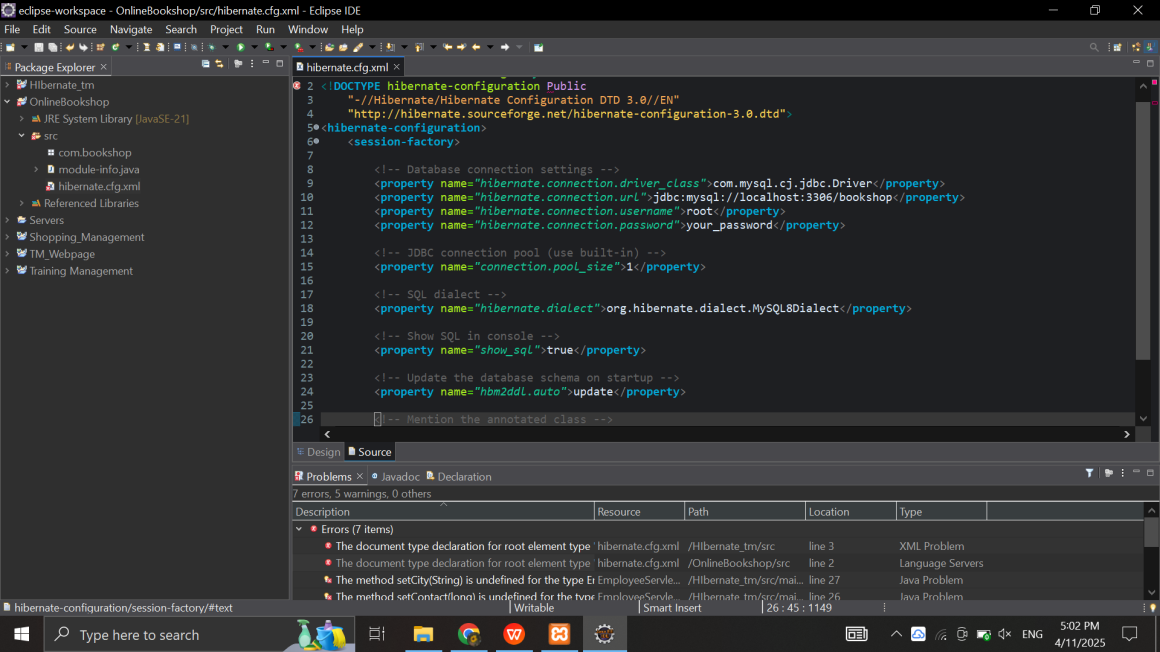
<property name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>

<property name="hibernate.hbm2ddl.auto">update</property>

<property name="show\_sql">true</property>

<mapping class="com.movie.Ticket"/>

</session-factory>

</hibernate-configuration>

# Step 4: Create Reservation Entity — Reservation.java

package com.movie;

import javax.persistence.\*;

@Entity

@Table(name = "ticket")

public class Ticket {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String customer\_name;

private String movie\_name;

private String seat\_number;

private java.sql.Date show\_date;

// Getters and Setters

public int getId() { return id; }

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

public String getCustomer\_name() { return customer\_name; }

public void setCustomer\_name(String customer\_name) { this.customer\_name = customer\_name; }

public String getMovie\_name() { return movie\_name; }

public void setMovie\_name(String movie\_name) { this.movie\_name = movie\_name; }

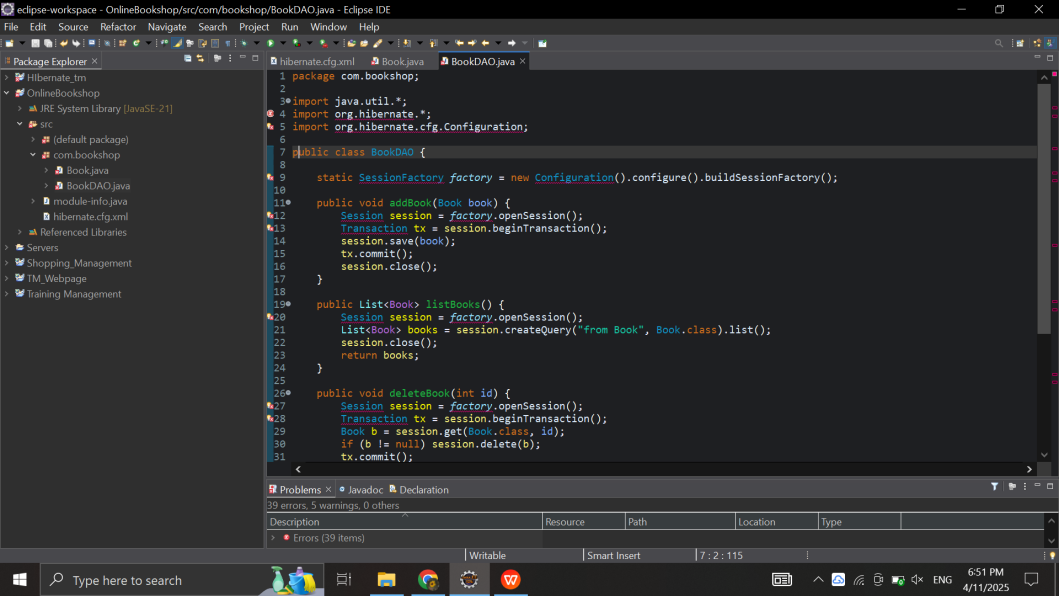
public String getSeat\_number() { return seat\_number; }

public void setSeat\_number(String seat\_number) { this.seat\_number = seat\_number; }

public java.sql.Date getShow\_date() { return show\_date; }

public void setShow\_date(java.sql.Date show\_date) { this.show\_date = show\_date; }

}



# Step 6: Create Main Class — Main.java

package com.railway;

import java.util.\*;

public class Main {

public static void main(String[] args) {

ReservationDAO dao = new ReservationDAO();

Reservation r1 = new Reservation();

r1.setPassenger\_name("John Doe");

r1.setTrain\_name("Express 101");

r1.setSeat\_number("A1");

r1.setTravel\_date(java.sql.Date.valueOf("2025-06-01"));

dao.addReservation(r1);

List<Reservation> res = dao.listReservations();

for (Reservation r : res) {

System.out.println(r.getId() + " " + r.getPassenger\_name());

}

dao.updateReservationSeat(1, "B2");

// dao.deleteReservation(1);

}

}

