WEEK-5 Design a java application to read any data using hibernate framework

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

package klh.edu.in.BOOKCURD;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "book")

public class Book {

private long id;

private String title;

private String author;

private float price;

public Book() { }

@Id

@Column(name = "book\_id")

@GeneratedValue(strategy = GenerationType.IDENTITY)

public long getId() {

return id;

}

public void setId(long id) {

this.id = id;

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getAuthor() {

return author;

}

public void setAuthor(String author) {

this.author = author;

}

public float getPrice() {

return price;

}

public void setPrice(float price) {

this.price = price;

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

package klh.edu.in.BOOKCURD;

import java.util.Scanner;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.boot.MetadataSources;

import org.hibernate.boot.registry.StandardServiceRegistry;

import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

public class BookManager {

protected SessionFactory sessionFactory;

protected void setup() {

final StandardServiceRegistry registry = new StandardServiceRegistryBuilder()

.configure("hibernate.cfg.xml").build();

try {

sessionFactory = new MetadataSources(registry).buildMetadata().buildSessionFactory();

} catch (Exception ex) {

StandardServiceRegistryBuilder.destroy(registry);

}

}

protected void create() {

Book book = new Book();

book.setTitle("OOps");

book.setAuthor("Ganesh");

book.setPrice(1.9f);

Session session = sessionFactory.openSession();

session.beginTransaction();

session.save(book);

session.getTransaction().commit();

session.close();

}

protected void read() {

Session session = sessionFactory.openSession();

long bookId = 3;

Book book = session.get(Book.class, bookId);

System.out.println("Title: " + book.getTitle());

System.out.println("Author: " + book.getAuthor());

System.out.println("Price: " + book.getPrice());

session.close();

}

protected void update() {

Book book = new Book();

book.setId(2);

book.setTitle("AI");

book.setAuthor("VINAY");

book.setPrice(3.9f);

Session session = sessionFactory.openSession();

session.beginTransaction();

session.update(book);

session.getTransaction().commit();

session.close();

}

protected void delete() {

Book book = new Book();

book.setId(3);

Session session = sessionFactory.openSession();

session.beginTransaction();

session.delete(book);

System.out.println("Deleted");

session.getTransaction().commit();

session.close();

}

protected void exit() {

sessionFactory.close();

}

public static void main(String[] args) {

BookManager manager = new BookManager();

manager.setup();

@SuppressWarnings("resource")

Scanner myInput = new Scanner( System.in );

int choice;

System.out.print( "Enter your choice: " );

choice= myInput.nextInt();

switch (choice)

{

case 1:

manager.create();

break;

case 2:

manager.read();

break;

case 3:

manager.update();

break;

case 4:

manager.delete();

break;

default:

System.out.println("invalid option");

}

manager.exit();

}

}

