Adding a New Product in the Database

Source code

**index.jsp**

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

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<form action ="insert.jsp" method="post">

Product ID<input type="text" name="id"><br>

Product Name<input type="text" name="name"><br>

Product Quantity<input type="text" name="quantity"><br>

<input type="submit">

</form>

</body>

</html>

**insert.jsp**

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

pageEncoding="UTF-8"%>

<%@page import="com.example.\*" %>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Insert title here</title>

</head>

<body>

<%

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

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

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

Product p = new Product();

p.setId(id);

p.setName(name);

p.setQuantity(quantity);

InsertDAO dao = new InsertDAO();

int res = dao.insert(p);

if(res>0)

{

out.print("Insertion done");

}

%>

</body>

</html>

**InsertDAO.java**

package com.example;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

public class InsertDAO {

public int insert(Product p) {

SessionFactory sf = Config.hibConfig();

Session s =sf.openSession();

Transaction t =s.beginTransaction();

int r =(Integer) s.save(p);

t.commit();

return r;

}

}

**Product.java**

package com.example;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Product {

@Id

@GeneratedValue(strategy=GenerationType.AUTO)

private int id;

@Column

private String name;

@Column

private int quantity;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

}

**Config.java**

package com.example;

import org.hibernate.SessionFactory;

import org.hibernate.boot.Metadata;

import org.hibernate.boot.MetadataSources;

import org.hibernate.boot.registry.StandardServiceRegistry;

import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

public class Config {

public static SessionFactory hibConfig() {

StandardServiceRegistry ssr=new StandardServiceRegistryBuilder().configure("hibernate.cfg.xml").build();

Metadata md=new MetadataSources(ssr).getMetadataBuilder().build();

SessionFactory sf=md.getSessionFactoryBuilder().build();

return sf;

}

}

**Hibernate.cfg.xml**

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

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

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

<property name="hibernate.connection.password">123456</property>

<property name="hibernate.connection.url">jdbc:mysql://localhost/db6</property>

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

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

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

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

<mapping class="com.example.Product"/>

</session-factory>

</hibernate-configuration>

Github Link:

https://github.com/ashwinbalaji44/Assignment.git