Phase-2 Practice Project

Adding a New Product in the Database

Source code

Product.java:

```java
package com.abc.pojo;


import java.util.Date;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Temporal;
import javax.persistence.TemporalType;

@Entity
public class Product {
@Id
@GeneratedValue(strategy = GenerationType.IDENTITY)
	private int pid;
	private String pname;
	private float cost;
	//when we have a util based date then we need to @ it with Temporal
	@Temporal(TemporalType.DATE)
	private Date orderdate;
	public int getPid() {
		return pid;
	}
	public void setPid(int pid) {
		this.pid = pid;
	}
	public String getPname() {
		return pname;
	}
	public void setPname(String pname) {
		this.pname = pname;
	}
	public float getCost() {
		return cost;
	}
	public void setCost(float cost) {
		this.cost = cost;
```

```java
        }
        public Date getOrderdate() {
                return orderdate;
        }
        public void setOrderdate(Date orderdate) {
                this.orderdate = orderdate;
        }



}
```

DBUtil:

```java
package com.abc.dButil;

import org.hibernate.Session;
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 DbUtil {

        StandardServiceRegistry  ssr=null;
        Metadata md=null;
        SessionFactory sf=null;
        Session session=null;

        public Session dbConn() {
                //SSR is used to map the config file and execute it .
                 ssr=new
StandardServiceRegistryBuilder().configure("hibernate.cfg1.xml").build();
                //Metadata of the xml file is read by this object
                md=new MetadataSources(ssr).getMetadataBuilder().build();
                //session-factory- db
                 sf=md.getSessionFactoryBuilder().build();
                //all the crud operations need to be done in Session
                session=sf.openSession();

                return session;
        }

}
```

ProductDAO:

```java
import java.util.List;

import org.hibernate.Session;
import org.hibernate.Transaction;
import org.hibernate.query.Query;

import com.abc.dButil.DbUtil;
import com.abc.pojo.Product;

public class ProductDAO {

        public int addProduct(Product product) {
                DbUtil dbconn=new DbUtil();
                Session session=dbconn.dbConn();
                Transaction trans=session.beginTransaction();
                int value=(Integer) session.save(product);
                trans.commit();
                session.close();
                return value;

        }

        public List<Product> display(){
                DbUtil dbconn=new DbUtil();
                Session session=dbconn.dbConn();
                Transaction trans=session.beginTransaction();
                //HQL
                Query query=session.createQuery("from Product");
                List<Product> list=query.list();
                trans.commit();
                session.close();
                return list;
        }

        public List<Product> delete(Product product) {
                DbUtil dbconn=new DbUtil();
                Session session=dbconn.dbConn();
                Transaction trans=session.beginTransaction();
                session.delete(product);
                trans.commit();
                session.close();
                return display();
        }

}
```

Hibernate.config.xml:

```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>
    <!--db driver , connection  -->
    <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/db2</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>
     <!--hibernate config  -->
    <property name="hbm2ddl.auto">update</property>
    <property name="show_sql">true</property>
     <!--pojo -->
    <mapping class="com.abc.pojo.Product"/>
  </session-factory>
</hibernate-configuration>
```

Index.jsp:

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
   <%@ page import="com.abc.dao.*" %>
  <%@ page import="com.abc.pojo.*" %>
  <%@page import="java.util.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="addproduct.jsp">
<input type="submit" value="addproduct">
</form>
<h1><i>List of products </i></h1>
<table border="1">
<tr><th>Pid</th><th>Pname</th><th>PCost</th><th>POrderDate</th><th>Edit
Action</th><th>Delete Action</th><tr>
<%
ProductDAO dao=new ProductDAO();
List<Product> list=dao.display();
for(Product p:list){
%>
<tr><td><%=p.getPid() %></td><td><%=p.getPname() %></td><td><%=p.getCost()
%></td><td><%=p.getOrderdate() %></td><td><a
```

```
href="edit.jsp?id=<%=p.getPid()%>">Edit</a></td><td><a
href="delete.jsp?id=<%=p.getPid()%>">Delete</a></td></tr>
<%} %>
</table>
</body>
</html>
```

addProduct.jsp:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1><i>Add a product</i></h1>
<form action="addController.jsp">
<table>
<tr><td>Pname</td><td><input type="text" name="pname"></td></tr>
<tr><td>Pcost</td><td><input type="text" name="pcost"></td></tr>
<tr><td>Orderdate</td><td><input type="date" name="pdoo"></td></tr>
<tr><td></td><td><input type="submit" value="add"></td></tr>
</table>
</form>
</body>
</html>
```

addController.jsp:

```
<%@page import="java.text.SimpleDateFormat"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
  <%@ page import="com.abc.dao.*" %>
  <%@ page import="com.abc.pojo.*" %>
  <%@page import="java.util.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<%
ProductDAO dao=new ProductDAO();
Product product=new Product();
product.setPname(request.getParameter("pname"));
product.setCost(Float.parseFloat(request.getParameter("pcost")));
//mysql always accept the date in the format of  yyyy-MM-dd
//convert the java based util date to sql date
```

```java
SimpleDateFormat dateFormat=new SimpleDateFormat("yyyy-MM-dd");
Date date=dateFormat.parse(request.getParameter("pdoo"));
product.setOrderdate(date);
int row=dao.addProduct(product);
if(row>0){
        response.sendRedirect("success.jsp");
}
else{
        response.sendRedirect("fail.jsp");
}
%>
</body>
</html>
```

Success.jsp:

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>the product got added successfully....</h1>
<%@ include file="index.jsp" %>
</body>
</html>
```

Delete.jsp:

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
    <%@ page import="com.abc.dao.*" %>
   <%@ page import="com.abc.pojo.*" %>

    <%@page import="java.util.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>deleted the product successfully....</h1>
<%@include file="button.jsp" %>
<h1><i>List of products </i></h1>
<table border="1">
<tr><th>Pid</th><th>Pname</th><th>PCost</th><th>POrderDate</th><th>Edit
Action</th><th>Delete Action</th><tr>
```

```jsp
<%
ProductDAO dao=new ProductDAO();
Product product=new Product();
product.setPid(Integer.parseInt(request.getParameter("id")));
List<Product> list=dao.delete(product);
for(Product p:list){
%>
<tr><td><%=p.getPid() %></td><td><%=p.getPname() %></td><td><%=p.getCost()
%></td><td><%=p.getOrderdate() %></td><td><a
href="edit.jsp?id=<%=p.getPid()%>">Edit</a></td><td><a
href="delete.jsp?id=<%=p.getPid()%>">Delete</a></td></tr>
<%} %>
</table>
</body>
</html>
```

Button.jsp:

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="addproduct.jsp">
<input type="submit" value="addproduct">
</form>
</body>
</html>
```