

Adding a New Product in the Database: Source Code

(Phase 2 – Practice Project 3)

Name : K V Sagar

Git Link : [simpliLearn/Phase-2/30-PracProj-AddProductHibernate at main · K-V-Sagar/simpliLearn \(github.com\)](https://github.com/K-V-Sagar/simpliLearn/tree/main/Phase-2/30-PracProj-AddProductHibernate)

products.java

```
package com.samples.domain;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Table;

@Entity
@Table(name="products")
public class products {

    @Id
    @Column(name="productID")
    private String productID;

    @Column(name="productName")
    private String productName;

    @Column(name="quantity")
    private String quantity;
```

```
// constructor

public products(String productID, String productName, String quantity) {
    super();
    this.productID = productID;
    this.productName = productName;
    this.quantity = quantity;
}

// Getters and Setters

public String getProductID() {
    return productID;
}

public void setProductID(String productID) {
    this.productID = productID;
}

public String getProductName() {
    return productName;
}

public void setProductName(String productName) {
    this.productName = productName;
}

public String getQuantity() {
    return quantity;
}

public void setQuantity(String quantity) {
    this.quantity = quantity;
}

}
```

add.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>Enter Product Details</h1>
    <form action="addProductServlet" method="post">
        <table>
            <tr>
                <td>Product ID: </td>
                <td><input name="productID" /></td>
            </tr>
            <tr>
                <td>Product Name: </td>
                <td><input name="productName" /></td>
            </tr>
            <tr>
                <td>Quantity: </td>
                <td><input name="quantity" /></td>
            </tr>
            <tr>
                <td />
                <td><input type="submit" value="Add"/></td>
            </tr>
        </table>
    </form>

</body>
</html>
```

listProdDetails.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>

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

<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

    <h1>Product Details:</h1>
    <h3>( Printing from the ArrayList "details" )</h3>
```

```

        <table>
            <tr>
                <th>Product ID</th>
                <th>Product Name</th>
                <th>Quantity</th>
            </tr>
            <c:forEach var="prod" items="${Product_Details}" >
                <tr>
                    <td>${prod.productID}</td>
                    <td>${prod.productName}</td>
                    <td>${prod.quantity}</td>
                </tr>
            </c:forEach>
        </table>

</body>
</html>

```

addProductServlet.java

```

package com.samples.servlet;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.hibernate.Session;

import com.samples.domain.products;
import com.samples.utils.HibernateUtil;

/**
 * Servlet implementation class prodDetailsServlet
 */
@WebServlet("/addProductServlet")
public class addProductServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    List<products> details = new ArrayList<>();

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

        Session session = HibernateUtil.getSessionFactory().openSession();

        session.beginTransaction();

```

```

String prodID = request.getParameter("productID");
String prodName = request.getParameter("productName");
String quantity = request.getParameter("quantity");

products prod = new products(prodID, prodName, quantity);

session.save(prod);

session.getTransaction().commit();

session.close();

// displaying the table
details.add(prod);
// Product_details is used in for-each loop in the .jsp file
request.setAttribute("Product_Details", details);
// forwarding the request to the .jsp file
RequestDispatcher rd =
request.getRequestDispatcher("/listProdDetails.jsp");
rd.forward(request, response);

}

}

```

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>
        <!-- Database connection settings -->
        <property
name="connection.driver_class">com.mysql.jdbc.Driver</property>
        <property
name="connection.url">jdbc:mysql://localhost:3306/mydb</property>
        <property name="connection.username">root</property>
        <property name="connection.password">admin</property>
        <!-- SQL dialect -->
        <property
name="dialect">org.hibernate.dialect.MySQLDialect</property>

        <!-- Echo all executed SQL to stdout -->
        <property name="show_sql">true</property>

        <!-- Use annotation based mapping metadata -->
        <mapping class="com.samples.domain.products"/>
    </session-factory>
</hibernate-configuration>

```

HibernateUtil.java

```
package com.samples.utils;

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 HibernateUtil {

    private static final SessionFactory sessionFactory = buildSessionFactory();

    private static SessionFactory buildSessionFactory() {
        try {
            StandardServiceRegistry serviceRegistry = new
StandardServiceRegistryBuilder()
                .configure("hibernate.cfg.xml").build();
            Metadata metadata = new
MetadataSources(serviceRegistry).getMetadataBuilder().build();
            return metadata.getSessionFactoryBuilder().build();
        } catch (Exception ex) {
            System.out.println("SessionFactory creation failed.." + ex);
            throw ex;
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }

}
```

Project Structure:

