

Github link:- https://github.com/santhoshsai4517/Simplilearn_Phase2_3.21

Index.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>Add product</title>
</head>
<body>
<form action="AddProductServlet" method="post">
    <!-- Form fields for product name, description, and price -->
    <!-- Add validation if needed -->
    <input type="text" name="name" placeholder="Product Name" required /><br />
    <textarea name="description" placeholder="Product Description"
required></textarea><br />
    <input type="text" name="price" placeholder="Price" required /><br />
    <input type="submit" value="Add Product" />
</form>

</body>
</html>
```

productAdded.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Product Added</title>
</head>
<body>
    <h1>Product Added Successfully</h1>
    <p>Your product has been added to the database.</p>

    <p><a href="index.jsp">Go back to the Home Page</a></p>
</body>
</html>
```

HibernateUtil.java

```
package com.ecommerce;

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;

    static {
        try {
            StandardServiceRegistry standardRegistry = new
StandardServiceRegistryBuilder()

.configure("hibernate.cfg.xml").build();
            Metadata metaData = new
MetadataSources(standardRegistry).getMetadataBuilder().build();
            sessionFactory =
metaData.getSessionFactoryBuilder().build();
        } catch (Throwable th) {
            throw new ExceptionInInitializerError(th);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
}
```

Product.java

```
package com.ecommerce;

import java.math.BigDecimal;

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 = "products")
public class Product {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;
    private String name;
    private String description;
    private BigDecimal price;
    // public Product(int id, String name, String description, BigDecimal
price) {
    //     super();
    //     this.id = id;
    //     this.name = name;
    //     this.description = description;
    //     this.price = price;
    // }
    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 String getDescription() {
        return description;
    }
    public void setDescription(String description) {
        this.description = description;
    }
    public BigDecimal getPrice() {
        return price;
    }
    public void setPrice(BigDecimal price) {
        this.price = price;
    }
}

```

Product.hbm.xml

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
  <class name="com.ecommerce.Product" table="products">
    <id name="id" column="id">
      <generator class="identity" />
    </id>
    <property name="name" />
    <property name="description" />
    <property name="price" />
  </class>
</hibernate-mapping>
```

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="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/ecommerce</proper
ty>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password"></property>

    <!-- Specify dialect -->
    <property
name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

    <!-- Specify mapping files -->
    <mapping resource="com/ecommerce/Product.hbm.xml" />
    <mapping class="com.ecommerce.Product" />

  </session-factory>
</hibernate-configuration>
```

AddProductServlet.java

```
import java.io.IOException;
import java.math.BigDecimal;

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 java.io.IOException;
import java.math.BigDecimal;
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 org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import com.ecommerce.HibernateUtil;
import com.ecommerce.Product;

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

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AddProductServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request,
    HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        // TODO Auto-generated method stub
    }
}
```

```

        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request,
HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        // Retrieve form data
        String name = request.getParameter("name");
        String description = request.getParameter("description");
        BigDecimal price = new BigDecimal(request.getParameter("price"));

        // Create a Product object
        Product product = new Product();
        product.setName(name);
        product.setDescription(description);
        product.setPrice(price);

        // Save the product using Hibernate
        SessionFactory sessionFactory = HibernateUtil.getSessionFactory();
        Session session = sessionFactory.openSession();
        Transaction transaction = session.beginTransaction();
        session.save(product);
        transaction.commit();
        session.close();

        // Redirect to a confirmation page or display a message
        response.sendRedirect("productAdded.jsp");
    }
}

```

Web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" id="WebApp_ID"
version="4.0">
    <display-name>HibernateLazyCollections</display-name>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
        <welcome-file>default.html</welcome-file>
        <welcome-file>default.htm</welcome-file>
        <welcome-file>default.jsp</welcome-file>
    </welcome-file-list>
    <servlet>

```

```
        <servlet-name>AddProductServlet</servlet-name>
        <servlet-class>AddProductServlet</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>AddProductServlet</servlet-name>
        <url-pattern>/AddProductServlet</url-pattern>
    </servlet-mapping>
</web-app>
```