

Source code for AddingNewProduct:

1. com.java

```
package com.bean;

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

@Entity
@Table(name="furniture")
public class Product {
    @Id
    private int pid;
    private String pname;
    private float price;
    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 getPrice() {
        return price;
    }
    public void setPrice(float price) {
        this.price = price;
    }
    @Override
    public String toString() {
        return "Product [pid=" + pid + ", pname=" + pname + ", price=" + price + "]\n";
    }
}
```

2. **com.controller:**

```
package com.controller;

import java.io.IOException;
import java.io.PrintWriter;

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

import com.bean.Product;
import com.service.ProductService;

/**
 * Servlet implementation class ProductStoreController
 */
public class ProductStoreController extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ProductStoreController() {
        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 {
```

```
// TODO Auto-generated method stub
//doGet(request, response);
PrintWriter pw = response.getWriter();

response.setContentType("text/html");

int pid = Integer.parseInt(request.getParameter("pid"));
String pname = request.getParameter("pname");
float price = Float.parseFloat(request.getParameter("price"));

Product p = new Product();
p.setPid(pid);
p.setPname(pname);
p.setPrice(price);

ProductService ps = new ProductService();
String result = ps.storeProduct(p);

pw.print(result);
RequestDispatcher rd = request.getRequestDispatcher("storeProduct.html");
rd.include(request, response);

    }
}
```

3. com.dao:

```
package com.dao;

import java.util.List;

import javax.persistence.TypedQuery;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

import com.bean.Product;

public class ProductDao {

    public int storeProduct(Product product) {
        try {
            Configuration con = new Configuration();
            con.configure("hibernate.cfg.xml");
            SessionFactory sf = con.buildSessionFactory();
            Session session = sf.openSession();

            Transaction tran = session.getTransaction();
            tran.begin();

            session.save(product);

            tran.commit();
            return 1;
        } catch (Exception e) {
            System.out.println(e);
            return 0;
        }
    }
}
```

4. com.service:

```
package com.service;

import java.util.Iterator;
import java.util.List;

import com.bean.Product;
import com.dao.ProductDao;

public class ProductService {

    ProductDao pd = new ProductDao();

    public String storeProduct(Product product) {
        if(product.getPrice()<100) {
            return "Product price must be > 100";
        }else if(pd.storeProduct(product)>0) {
            return "Product stored successfully";
        }else {
            return "Product didn't store id must be unique";
        }
    }
}
```

5.hibrernite.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.cj.jdbc.Driver</property>
        <property name="hibernate.connection.password">Vyshi@1432</property>
        <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/Product
    </property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
        <mapping class="com.bean.Product"/>
    </session-factory>
</hibernate-configuration>
```

6.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>AddingProducts</display-name>
  <display-name>AddingProducts</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.jsp</welcome-file>
    <welcome-file>default.htm</welcome-file>
  </welcome-file-list>
  <servlet>
    <description></description>
    <display-name>ProductStoreController</display-name>
    <servlet-name>ProductStoreController</servlet-name>
    <servlet-class>com.controller.ProductStoreController</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>ProductStoreController</servlet-name>
    <url-pattern>/ProductStoreController</url-pattern>
  </servlet-mapping>
</web-app>
```

7. index.html:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h2>Product Management System</h2>
<a href="storeProduct.html">Add Product</a> !
</body>
</html>
```

8. storeProdcut.html:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h2>Add Product</h2>
<form action="ProductStoreController" method="post">
<label>PId</label>
<input type="number" name="pid"><br/>
<label>PName</label>
<input type="text" name="pname"><br/>
<label>Price</label>
<input type="number" name="price"><br/>
<input type="submit" value="Save Product">
<input type="reset" value="reset">
</form>
<br/>
<a href="index.html">Main Page</a>
</body>
</html>
```

9.pom.xml:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>AddingProducts</groupId>
  <artifactId>AddingProducts</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <build>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
```

```
<configuration>
  <release>18</release>
</configuration>
</plugin>
<plugin>
  <artifactId>maven-war-plugin</artifactId>
  <version>3.2.3</version>
</plugin>
</plugins>
</build>
<dependencies>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.16</version>
</dependency>
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-core</artifactId>
  <version>5.5.6.Final</version>
</dependency>

</dependencies>
</project>
```