

Add the new product in the database:

### 1.product.html:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
Enter Name<input type="text" name="txtName">
Enter Price<input type="value" name="txtprice">
<input type="submit" value="Save">
</form>
</body>
</html>
```

### 2.index.jsp:

```
<html>
<body>
<h2>Hello World!</h2>
<a href="productform.html">productform</a>
</body>
</html>
```

### 3.product.java:

```
package com.productretrieve;
public class Product {
    private int rollno;

    private int Price;
    private String Name;
    public int getRollno() {
        return rollno;
    }
    public void setRollno(int rollno) {
        this.rollno = rollno;
    }

    public String getName() {
        return Name;
    }
    public void setName(String Name) {
        this.Name=Name;
    }
    public int getPrice() {
        return Price;
    }
    public void setPrice(int Price) {
        this.Price=Price;
    }
}
```

```
}
```

#### 4.ProductDAO.java:

```
package com.productretrieve;
import java.io.IOException;

import java.io.PrintWriter;

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.cfg.Configuration;

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

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

        String name = request.getParameter("txtName");
        int price = Integer.parseInt(request.getParameter("txtprice"));
        // put values in Object
        Product stud = new Product();
        stud.setPrice(price);
        stud.setName(name);

        SessionFactory sessionFactory = new
Configuration().configure().buildSessionFactory();
        Session session = sessionFactory.openSession();
        session.beginTransaction();
        int i = (Integer) session.save(stud);
        session.getTransaction().commit();

        session.close();

        PrintWriter out = response.getWriter();
        if (i > 0)
            out.println("Record inserted");
    }
}
```

```

        else
            out.println("Record not inserted");
    }
}

```

## 5.Hibernate.cfg.xml:

```

<?xml version='1.0' encoding='utf-8'?>
<!--
    ~ Hibernate, Relational Persistence for Idiomatic Java
    ~
    ~ License: GNU Lesser General Public License (LGPL), version 2.1 or later.
    ~ See the lgpl.txt file in the root directory or
    ~ <a href="http://www.gnu.org/licenses/lgpl-2.1.html">.
    -->
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
    <session-factory>

        <property name="format_sql">false</property>

        <property
name="connection.url">jdbc:mysql://localhost:3306/newdata</property>
        <property
name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property name="connection.username">root</property>
        <property name="connection.password">Haritha@28</property>

        <!-- JDBC connection pool (use the built-in) -->
        <property name="connection.pool_size">1</property>

        <!-- SQL dialect -->
        <property name="dialect">org.hibernate.dialect.MySQL5Dialect</property>

        <!-- Disable the second-level cache -->
        <property
name="cache.provider_class">org.hibernate.cache.internal.NoCacheProvider</property>
    >

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

        <!-- Drop and re-create the database schema on startup -->
        <property name="hbm2ddl.auto">create</property>
        <property name="hbm2ddl.auto">update</property>

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

<mapping resource="product.hbm.xml"></mapping>

```

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

## 6.product.hbm.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping
    SYSTEM
    "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd" >
    <hibernate-mapping>
        <class name="com.productretrieve.Product" table="newtable">
            <id name="rollno" column="rollno">
                <generator class="increment"/>
            </id>
            <property name="name" length="20"></property>
            <property name="price" ></property>
        </class>
    </hibernate-mapping>
```