

Adding a New Product in the Database

Index.html

```
<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

    <form align="center" action="save" method="post" >

        Product Name: <input type="text" name="Pname" placeholder="Name">

        <br>

        <br>

        Product Description: <input type="text" name="Pdesc"

            placeholder="Discription"> <br>

        <br>

        Product Price: <input type="text" name="Pprice" placeholder="Price">

        <br>

        <br>

        <input type="submit" value="Add">

    </form>

</body>

</html>
```

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>

        <property name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
```

```

        <property
name="connection.url">jdbc:mysql://localhost:3306/productdata</property>
        <property name="connection.username">root</property>
        <property name="connection.password">978213</property>
        <property name="hbm.auto">update</property> <property
name="show_sql">true</property>

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

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

```

Product.hbm.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
    "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<hibernate-mapping package="com.simplilearn.demo">
    <class name="Product" table="productinfo"> <id
name="ID" column="ID">
        <generator class="increment"></generator>
    </id>
    <property name="name" type="string"></property>
    <property name="description" type="string"></property>
    <property name="price" type="string"></property>
</class>

</hibernate-mapping>

```

Product.java

```

package com.simplilearn.demo;

```

```
public class Product {  
    private int ID;  
    private String name;  
    private String description;  
    private String price;  
  
    public Product(String name, String description, String price)  
        { this.name = name;  
          this.description = description;  
          this.price = price;  
        }  
  
    public int getID() {  
        return ID;  
    }  
    public void setID(int iD) {  
        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 String getPrice() {  
    return price;  
}  
public void setPrice(String price) {  
    this.price = price;  
}  
}
```

HibernateUtils.java

```
package com.simplilearn.demo;  
  
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 HibernateUtils {  
  
    private static SessionFactory sessionFactory;  
  
    static {  
  
        try {  
  
            StandardServiceRegistry registry=new StandardServiceRegistryBuilder()  
                .configure("hibernate.cfg.xml").build();  
            Metadata metadata= new MetadataSources(registry).getMetadataBuilder().build();  
  
            sessionFactory= metadata.getSessionFactoryBuilder().build();  
  
        }catch (Exception e) {
```

```
        // TODO: handle exception
        e.printStackTrace();
    }

}

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

SaveServlet.java

```
package com.simplilearn.demo;

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.Transaction;
import org.hibernate.cfg.Configuration;

@WebServlet("/save")

public class SaveServlet extends HttpServlet {

    private static final long serialVersionUID = 1L;

    @Override

    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException,
IOException {

        resp.setContentType("text/html");
        PrintWriter out= resp.getWriter();

        //create configuration
        Configuration configuration = new Configuration();
        configuration.configure("hibernate.cfg.xml");

        //sessionFactory creating

        //sessionFactory - session object we are getting from factory, eg. i you want to buy car u go
to car factory

        //sessionFactory is your datasource means- how u will be connecting to your database
        //to create sessionFactory you will be needing configuration

        SessionFactory factory= HibernateUtils.getSessionFactory();
        Session session= factory.openSession();

        Transaction tx= session.beginTransaction();

        String prodname= req.getParameter("Pname");
        String proddesc= req.getParameter("Pdesc");

```

```
String prodprice= req.getParameter("Pprice");
```

```
Product s1=new Product(prodname, proddesc,  
prodprice); session.save(s1);
```

```
tx.commit();
```

```
session.close();
```

```
out.print("<h4>Data inserted Successfully</h4>");
```

```
}
```

```
@Override
```

```
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException,  
IOException {
```

```
    // TODO Auto-generated method stub
```

```
    doGet(req, resp);
```

```
}
```

```
}
```