DBConfig:

```
package com.addaproduct;



import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.Properties;

public class DBConfig {


      public static Connection getConnection(Properties props) {


            Connection connection=null;

            try {
                  //get url,username,password
            String driver= props.getProperty("driver");
            String url= props.getProperty("url");
            String username= props.getProperty("username");
            String password= props.getProperty("password");

            //to load driver
                  Class.forName(driver);

                  try {
                        connection= DriverManager.getConnection(url, username, password);
                  } catch (SQLException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                  }

            } catch (ClassNotFoundException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            }

            return connection;
      }

}
```

EProduct:

```
package com.addaproduct;


import java.math.BigDecimal;
import java.util.Date;


public class EProduct {

      private long ID;
      private String name;
      private BigDecimal price;
      private Date dateAdded;




      public long getID() {
            return ID;
      }

      public void setID(long iD) {
            ID = iD;
      }

      public String getName() {
            return name;
      }

      public void setName(String name) {
            this.name = name;
      }

      public BigDecimal getPrice() {
            return price;
      }

      public void setPrice(BigDecimal price) {
            this.price = price;
      }

      public Date getDateAdded() {
            return dateAdded;
      }

      public void setDateAdded(Date dateAdded) {
            this.dateAdded = dateAdded;
      }

}
```

HibernateUtil:

```
package com.addaproduct;

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 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){

            }



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

}
```

saveservlet:

```
package com.addaproduct;

import java.io.IOException;
import java.io.InputStream;
import java.io.PrintWriter;
import java.math.BigDecimal;
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.Properties;

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




@WebServlet("/save")
public class saveservlet extends HttpServlet {
      private static final long serialVersionUID = 1L;


      protected void doGet(HttpServletRequest request, HttpServletResponse 
response) throws ServletException, IOException {
            PrintWriter out = response.getWriter();

            Properties props = new Properties();

            InputStream in = 
getServletContext().getResourceAsStream("/WEB-INF/application.properties");
```

```
            props.load(in);

            Connection conn = DBConfig.getConnection(props);

            /*
             * if (conn != null) out.print("Connection is Established"); else
             * out.print("Error while connecting connection");
             */

            String name= request.getParameter("pname");
            String price= request.getParameter("pprice");


            try {
                  CallableStatement stmt= conn.prepareCall("{call
add_product(?,?)}");

                  stmt.setString(1, name);
                  stmt.setBigDecimal(2, new BigDecimal(price));


                  int x=stmt.executeUpdate();
                  if(x>0) {

                        System.out.println("Data inserted Successfully");
                        out.print("Data inserted Successfully");
                        response.sendRedirect("index.html");
                  }
                  else {
                        System.out.println("Error while uploading");
                        response.sendRedirect("index.html");
                  }


            } catch (SQLException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            }
      }




      protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
            // TODO Auto-generated method stub
            doGet(request, response);
      }

}
```