

```
public class ClothingBrands {  
  
    private String name;  
    private String fabric;  
    private double cost;  
    private String manufacturer;  
  
    public ClothingBrands()  
    {  
        setName("default name");  
        setFabric("default fabric");  
        setCost(0.00);  
        setManufacturer("default manufacturer");  
    }  
  
    public ClothingBrands(String aName, String aFabric, double aCost, String aManufacturer)  
    {  
        setName(aName);  
        setFabric(aFabric);  
        setCost(aCost);  
        setManufacturer(aManufacturer);  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public String getFabric() {  
        return fabric;  
    }  
  
    public void setFabric(String fabric) {  
        this.fabric = fabric;  
    }  
  
    public double getCost() {  
        return cost;  
    }  
  
    public void setCost(double cost) {  
        this.cost = cost;  
    }  
}
```

```

public class ClothingBrandsDB {

    static String dsn = "SportsBrands";
    static String dbUrl = "jdbc:odbc:" + dsn;
    static String user = "";
    static String password = "";

    static Connection conn = null;

    private static Connection getConnection()
    {
        try
        {
            conn = DriverManager.getConnection(dbUrl, user, password);
            return conn;
        }
        catch (SQLException e)
        {
            System.err.println(e);
            return null;
        }
    }

    public ArrayList<ClothingBrands> getProducts()
    {
        String sql = "SELECT name, fabric, price, manufacturer "
            + "From SportsBrands "
            + "ORDER BY name";
        ArrayList<ClothingBrands> products = new ArrayList();

        try
        {
            Connection connection = getConnection();
            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery(sql);
            {
                while (rs.next())
                {
                    String name = rs.getString("Name");
                    String fabric = rs.getString("Fabric");
                    Double cost = rs.getDouble("Cost");
                    String manufacturer = rs.getString("Manufacturer");
                    ClothingBrands p = new ClothingBrands(name, fabric, cost, manufacturer);
                    products.add(p);
                }
            }
            return products;
        }
        catch (SQLException e)
        {
            System.err.println(e);
            return null;
        }
    }
}

```

```

,
public boolean updateProduct(ClothingBrands p)
{
    String sql = "UPDATE SportsBrands SET " +
        "Fabric = ?, " +
        "Cost = ? " +
        "WHERE name = ?";
    try (Connection connection = getConnection();
        PreparedStatement ps = connection.prepareStatement(sql))
    {
        ps.setString(1, p.getFabric());
        ps.setDouble(2, p.getCost());
        ps.setString(3, p.getName());
        ps.executeUpdate();
        return true;
    }
    catch (SQLException e)
    {
        System.err.println(e);
        return false;
    }
}

public ClothingBrands addProduct (String code, String description, double price, String author, String version)
{
    String sql = "INSERT INTO Products "
        + "(ProductCode, Description, Price, Author, Version)"
        + "VALUES "
        + "(" + code + ", " + description + ", " + price + ", " + author + ", " + version + ")";
    try {
        Connection connection = getConnection();
        Statement stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery(sql);
        {
            rs.close();
            return null;
        }
    }
    catch (SQLException e)
    {
        System.err.println(e);
        return null;
    }
}

```

```

-
<form>
    Primary Key: <input type="text" name="primaryKey" size="20"><br>
    <br> <input type="submit" name="submit" value="FIND RECORD TO UPDATE">
<br>
</form>
<br>

<%
    ClothingBrandsDB productDB = null;
    ClothingBrands product = null;

    String input = request.getParameter("primaryKey");

    if (input != null && !input.isEmpty()) {

        try
        {
            productDB = new ClothingBrandsDB();
            //test getProduct( ) method
            product = ClothingBrandsDB.getProduct(input);
        }
        catch(Exception err)
        {
            out.print("ERROR: " + err.toString() + "<br /><br />");
        }
    }
%>

<form action="Update2.jsp">
    Code: <input type="text" name="code" size="20"
        value=<%=product.getName() %>>
        <br /><br />
    Description: <input type="text" name="description" size="40"
        value="<%= product.getFabric() %>">
        <br /><br />
    Price: <input type="text" name="price" size="20"
        value=<%=product.getCost() %>><br />
        <br /><br />

    <%
        if(product.getCost() != 0.00)
        {
    %>
        Modify above information if necessary and then click on Update Record button
        <br /> <input type="submit" name="submit" value="Update Record">
    <% } %>
</form>

```