Product Details Portal

[writeup]

The code you provided consists of three main components: a Java class, a servlet, and JSP files. Let's go through each component and explain its purpose:

1. `Product.java`:

This class represents a Product object with attributes such as `id`, `name`, and `price`. It also includes getter and setter methods for accessing and modifying the attributes. The `toString()` method is overridden to provide a string representation of the Product object.

2. `ProductServlet`:

This servlet handles the form submission from the `ProductForm.jsp` page. In the `doPost()` method, it retrieves the form data (id, name, and price) from the request parameters, creates a new `Product` object, and stores it in the session using the attribute name "product". Then, it redirects the user to the `ProductDisplay.jsp` page using `response.sendRedirect()`.

3. `ProductDisplay.jsp`:

This JSP page displays the details of the product stored in the session. It uses Expression Language (EL) to retrieve the product attributes (id, name, and price) from the session attribute "product" and displays them in an HTML table.

4. `ProductForm.jsp`:

This JSP page contains a form that allows users to enter the details of a product (id, name, and price). When the form is submitted, it sends a POST request to the `ProductServlet` for processing.

5. `web.xml`:

This file is the deployment descriptor for the web application. It maps the `ProductServlet` to the URL pattern "/ProductServlet" using the `<servlet>` and `<servlet-mapping>` tags. It also specifies the welcome file list, which determines the order in which the server looks for default files when accessing the root URL of the application.