Product Details Portal Write-UP

Step-By-Step Workflow

1. User accesses the `index.jsp` page through a web browser.

2. The user fills in the product details in the form (id, name, color, and price) and clicks the "Submit" button.

3. The form is submitted to the `Product` servlet (`Product.java`) as specified in the `action` attribute of the form.

4. The `doPost` method of the `Product` servlet is invoked.

5. The `processRequest` method is called, passing the `HttpServletRequest` and `HttpServletResponse` objects.

6. Inside the `processRequest` method:

   a. A `PrintWriter` is obtained from the response object to write the HTML response.

   b. The response content type is set to "text/html;charset=UTF-8".

   c. The HTML response is started with the `<!DOCTYPE html>` declaration.

   d. The try-catch block is used to handle exceptions.

   e. The product details (id, name, color, and price) are retrieved from the request parameters.

   f. A new `ProductClass` object, `product1`, is created with the retrieved product details.

   g. The HTTP session is obtained using `request.getSession()`.

   h. The `product1` object is stored as an attribute named "data" in the session.

   i. The `product1.getProductDetails()` method is called to get the product details as an `ArrayList` and store it as an attribute named "data" in the request.

   j. A `RequestDispatcher` is obtained using `request.getRequestDispatcher("Profile.jsp")`, specifying the JSP file to which the request will be forwarded.

   k. The `forward` method of the `RequestDispatcher` is called, forwarding the request and response objects to the "Profile.jsp" file for further processing.

   l. The closing HTML tags (`</body>` and `</html>`) are written to the response output stream.

7. The control is passed to the "Profile.jsp" file.

8. Inside the "Profile.jsp" file:

   a. The `ArrayList<ProductClass>` object, `p1`, is retrieved from the request attribute "data" using `request.getAttribute("data")`.

b. The HTML structure of the product list table is written, including table headers for ID, Name, Color, and Price.

c. An iterator is created to iterate over the `p1` list.

d. Inside the while loop, the product details (ID, Name, Color, and Price) are retrieved from each `ProductClass` object and displayed in the table rows.

e. The closing HTML tags (`</table>`, `</div>`, `</body>`, and `</html>`) are written to complete the HTML structure.

9. The modified HTML response, with the product details displayed in a table, is sent back to the user's web browser.

10. The user sees the rendered "Profile.jsp" page with the product details displayed in a tabular format.

This workflow illustrates how the user input is processed by the servlet, the data is stored in the session and request attributes, and finally how the control is forwarded to the JSP file for rendering the response with the retrieved data.