Title: Simple Product Registration Web Application using Java Servlet and Hibernate

Introduction: The given code represents a simple web application that allows users to register products by entering their details such as ID, product name, and price. The application is built using Java Servlet technology for handling HTTP requests and responses, and Hibernate for persisting the product data into a database.

Components:

eproduct.java: This class represents the model for the product entity. It defines the attributes of a product, such as pid (product ID), name, and price. Getters and setters are provided for accessing and modifying the attribute values.

Servlet.java: This class extends HttpServlet and handles the POST request from the product registration form. It retrieves the product details entered by the user, creates an instance of the eproduct class, and saves it to the database using Hibernate's session and transaction management. If the operation is successful, the user is redirected to the "success" page; otherwise, an error is displayed.

Success.java: This servlet class is responsible for displaying a success message after the product registration is completed. It retrieves the PrintWriter object from the response and writes an HTML response containing the success message.

Register.jsp: This JSP (JavaServer Pages) file represents the product registration form. It contains fields for entering the product ID, name, and price. The form's action is set to "Servlet," indicating that the data should be submitted to the Servlet class for processing.

Functionality:

When a user accesses the "Register.jsp" page, the product registration form is displayed.

The user enters the product details and clicks the "Submit" button.

The form data is submitted to the "Servlet" class using the POST method.

The "Servlet" class retrieves the submitted data, initializes a Hibernate session, and begins a transaction.

An instance of the "eproduct" class is created with the submitted data.

The product instance is saved to the database using the Hibernate session's "save" method.

If the transaction is committed successfully, the user is redirected to the "success" page, which displays a success message.

If an error occurs during the transaction, the transaction is rolled back, and an error message is displayed.