Retrieve Product Details Using Product ID Documentation

Project Overview

This project aims to create a simple servlet-based web application for retrieving product details from a MySQL database based on a provided product ID. The application consists of an HTML form to enter a product ID, a Java servlet for handling requests and querying the database, and a MySQL database with a "products" table.

Tools and Technologies Used

- **IDE:** Eclipse
- **Web Server:** Apache Tomcat
- **Database:** MySQL
- **JDBC Connector:** MySQL Connector/J

Project Structure

The project is organized into the following components:

- **HTML Form (index.html):**
  - Provides a simple HTML form for entering a product ID.
  - Submits the form data to the **`ProductDetailsServlet`** for processing.
- **Java Servlet (ProductDetailsServlet.java):**
  - Handles HTTP POST requests from the HTML form.
  - Retrieves the product details from the MySQL database using JDBC.
  - Generates HTML responses to display product details or an error message.
- **MySQL Database:**
  - Contains a "products" table with columns for product ID, name, price, and description.
  - Prepopulated with sample data.

Setting Up the Database

- Install MySQL and create a database named "ecommerce".
- Run the provided SQL script to create the "products" table and insert sample data.

```
CREATE DATABASE ecommerce;

USE ecommerce;

CREATE TABLE products (
  id INT PRIMARY KEY,
  name VARCHAR(255),
  price DECIMAL(10, 2),
  description TEXT
);

INSERT INTO products VALUES (1, 'Product 1', 19.99, 'Description of Product 1');
INSERT INTO products VALUES (6, 'Product iphone', 2229.99, 'Description of Product 2');

select * from products;
-- Add more products as needed
```

Configuring Eclipse and Tomcat

- Install Eclipse IDE and Apache Tomcat.
- Create a new Dynamic Web Project in Eclipse.
- Configure the project to use Apache Tomcat as the runtime.

JDBC Configuration

- Add the MySQL Connector/J JAR to the project's build path.
- Update the JDBC connection details in the **`ProductDetailsServlet`**:
  - Set the **`jdbcUrl`**, **`username`**, and **`password`** variables with your MySQL connection details.

Running the Application

- Deploy the project on Apache Tomcat.
- Run the Tomcat server and access the HTML form at **`http://localhost:8080/your-web-app-name/index.html`**.
- Enter a product ID and submit the form to view product details or an error message.