**Retrieving the Product Details Using the Product ID.**

**Project Details**

Dynamic Web Project to illustrate validation of the User Login

**Objective:**

To Create a servlet-based application that shows a form to enter a product ID. The product ID is then validated, and product details are retrieved from the database and displayed to the user. We need to create a product table in MySQL and prepopulate it with data. Use JDBC to do all database processing.

**Problem statement:**

As a part of developing an e-commerce web application, the admin backend requires a module that can retrieve product information based on the product ID.

**Requirements**

* Create an HTML page to take in a product ID
* Set up JDBC to work with the application
* Create a servlet that will take the product ID and use JDBC to query the database for the product
* If the product is found, the servlet will display the product details, otherwise it will show an error message

**Product Backlog:**

1. Creating a Product Search Page using HTML
2. Create a CSS file to format the product page.
3. Creating a database, Product Table and Insert value into It
4. Create a servlet to get requests and send responses to the Product Search Page.
5. Display Product Details , if product available else display appropriate message
6. Run and Test the application
7. Prepare documentation
8. Upload to GITHUB

**Technologies and tools Used**

• Servlet: to do the business logic and works a controller for the project.

• HTML: to design the pages.

• CSS: to format the contents.

• Eclipse: to write and run the code.

• Tomcat: to run and deploy servlet application.

**Flowcharts of the Application**

Display ‘Product Not Found’

‘

Display Product Details

IF PRODUCT EXISTS

YES

NOS

**Search Product by ID**

**Core concepts used in the project.**

• Object-Oriented: used to create and model objects for users and their credentials.

• HTML, CSS and JavaScript used to create pages

• Servlet is used to process request and provide response

**How to run the program**

• clone project

• Now run program on a server.

• Search Product Id

**Github Link:**

<https://github.com/bibhu-otz/FSD-COURSE-III-PRACTICE-PROJECTS/tree/main/LoginSystem>

**Source Code :**

**index.html**

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset=*"ISO-8859-1"*>  <title>Product Details</title>  <script src=*"https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"*></script>  <link href=*"https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"* rel=*"stylesheet"* integrity=*"sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"* crossorigin=*"anonymous"*>  <style>  **body**{  background-image: *linear-gradient(to right, rgba(255,0,0,0), rgba(255,0,0,1))*;    }  **h1**{    color: *blue*;  font-style: *oblique*;  display: *inline*;  margin-left:*270px*;  }  **table**{  border: *2px solid blue*;  margin-left: *340px*;  display:*none*;  }        </style>  </head>  <body>  <div class=*"container my-3"*>    <h1>Search Product Id : </h1>  <input class=*"py-3"* type=*"text"* name=*"productId"* id=*"productId"*>  <button class=*"btn btn-primary btn-lg mb-1"* onclick="getToResults()">Search</button>    </div>  <div class=*"container2 col-6"*>  <table class=*"table table-hover"* id=*"results"*>    </table>  </div>  <script src=*"https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"* integrity=*"sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"* crossorigin=*"anonymous"*></script>  <script type=*"text/javascript"*>  **function** getToResults() {    **var** params = {'productId' : document.getElementById("productId").value};  alert(params.productId);  $.ajax({  type: 'POST',  url: 'ProductDetails',  data: params,  async: **false**,  success: **function** (result) {  console.log(result);  document.getElementById('results').innerHTML=result;  document.getElementById('results').style="display:block";  }  });  }  </script>  </body>  </html> |

**ProductDetails.Java**

|  |
| --- |
| import java.io.IOException;  import java.io.InputStream;  import java.io.PrintWriter;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.ResultSetMetaData;  import java.util.Properties;  import javax.servlet.RequestDispatcher;  import javax.servlet.ServletException;  import javax.servlet.annotation.WebServlet;  import javax.servlet.http.HttpServlet;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import com.ecommerce.DBConnection;  @WebServlet("/ProductDetails")  public class ProductDetails extends HttpServlet {  private static final long serialVersionUID = 1L;    public ProductDetails() {  super();    }    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  try {  String id= request.getParameter("productId");  System.out.println(id);  response.setContentType("text/html");  PrintWriter out = response.getWriter();  InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  Properties props = new Properties();  props.load(in);    DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  String sql="SELECT \* FROM products WHERE productId='"+id+"'";  PreparedStatement pstm = conn.getConnection().prepareStatement(sql);  ResultSet rs = pstm.executeQuery();    /\* Printing column names \*/  ResultSetMetaData rsmd=rs.getMetaData();  int total=rsmd.getColumnCount();  out.print("<tr>");  for(int i=1;i<=total;i++)  {  out.print("<th scope=\"row\">"+rsmd.getColumnName(i)+"</th>");  }    out.print("</tr>");  /\* Printing result \*/    while(rs.next())  {  out.print("<tr><td>"+rs.getString(1)+"</td><td>"+rs.getString(2)+"</td><td>"+rs.getString(3)+"</td></tr>");    }    out.print("</table>");      }catch(Exception e) {  e.printStackTrace();    }  }    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  // TODO Auto-generated method stub  doGet(request, response);  }  } |

**Screenshots:**





