

Product Details Portal.

Index.jsp

Created simple form text fields to get data from the user like: product id, name, category, price

```
Product ID: <input type="text" name="productId"><br><br>
Product Name: <input type="text" name="productName"><br><br>
Category: <input type="text" name="productCategory"><br><br>
Price: <input type="text" name="productPrice"><br><br>
```

Defining a error class if any validation doesn't match then these error messages will appear.

```
<span class="error">${errors.notInt}</span>
<span class="error">${errors.invalidCateg}</span>
<span class="error">${errors.nullProdName}</span>
<span class="error">${errors.nullPrice}</span>
```

Output.jsp

Accessing all the product details using jstl library in the table format

```
<td><c:out value="${product.productId}"></c:out></td>
<td><c:out value="${product.productName}"></c:out></td>
<td><c:out value="${product.productCategory}"></c:out></td>
<td><c:out value="${product.productPrice}"></c:out></td>
```

Product.java

Created a class product with the following variables and generated getter setter for them.

```
private int productId;
private String productName;
private String productCategory;
private String productPrice;
```

```

public int getProductId() {
    return productId;
}

public void setProductId(int productId) {
    this.productId = productId;
}

```

productController.java

Created a map of strings to store all the error codes and execute them when needed.

```

Map<String, String> errors = new HashMap<String, String>();

```

Created a session. Gets the HttpSession object. If the request doesn't have a session associated with it, a new session is created.

```

HttpSession session = request.getSession();

```

Defined a regex expression here to validate the user entered data with this.

```

String regex = "[a-zA-Z0-9]*";

```

Checking if product category matches with the regex expression if not then inserting the key and value to the error map.

```

if(!matcher.matches()) {
    errors.put("invalidCateg", "Invalid category<br/>"); }

```

Checking all the parameters with same condition and inserting the values in map according to the parameter and printing out from output.jsp page.

```

if(request.getParameter("productName").length() == 0) {
    errors.put("nullProdName", "Enter a product name<br/>"); }

```

If error map is empty then , inserting the details in the product object p1.

```

product p1 = new product(Integer.parseInt(request.getParameter("productId")),
    request.getParameter("productName"), request.getParameter("productCategory"),
    request.getParameter("productPrice"));

```

The `setAttribute` method is used to store an object in session by allotting a unique string to the object. The `sendRedirect()` method of `HttpServletResponse` interface can be used to redirect response to another resource, it may be servlet, jsp or html file.

```
session.setAttribute("product", p1);  
response.sendRedirect("output.jsp");
```

if error map is not empty redirecting the the index.jsp page

```
session.setAttribute("errors", errors);  
response.sendRedirect("index.jsp");
```