**CSE1007**

**Java Programming**

**Digital Assignment - 1**

**Varun Singh Inda**

**16BEE0023**

**Question**

**Design a table and connect it to servlet program for supermarket billing**

**Solution**

**Environment Used :-** Netbeans

**Server Used :-** Apache Tomcat Server

**Database Used :-** Derby Netbeans

This program helps the billing at the supermarket. It adds all the items which are present in the database and does not add the items which are entered and not present in the database.

|  |  |
| --- | --- |
| Item | Rate |
| Soap | 40 |
| Detergent | 90 |
| Biscuits | 20 |
| Sauce | 110 |
| Snacks | 60 |
| Spices | 70 |
| Chips | 20 |
| Cosmetics | 80 |
| Medicines | 120 |
| Toys | 200 |

**It will not add the items which are not present in the database**

**Process**

1. The index.html is created so as to make the frontend of the file.
2. ShoppingCart.java is created to handle the backend.
3. Database named bill is created in the netbeans derby.
4. The user is asked to put the item name and the quantity of the item.
5. The name and the quantity of the item is stored in the cookies.
6. The add item button adds the item and redirects to the same index.html page to add another item.
7. The view button lists all the items.
8. When the view button is pressed then all the items stored in the cookie array will be compared with the items present in the database.
9. The items present exactly in the database will be shown in the final list.
10. The program ends when all the items present in the cookie array are checked.

**Index Program**

<!DOCTYPE html>

<!--

To change this license header, choose License Headers in Project Properties.

To change this template file, choose Tools | Templates

and open the template in the editor.

-->

<html>

<head>

<title>TODO supply a title</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<form method="get" action="ShoppingCart">

Enter Item Name <input type="text" name="item"><br>

Enter Item Quantity <input type="text" name="qty"><br>

<input type="submit" value="Add item" name="add">

<input type="submit" value="Show the bill" name="list">

</form>

</body>

</html>

**Servlet Program**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.sql.\*;

public class ShoppingCart extends HttpServlet{

public void service(HttpServletRequest req ,HttpServletResponse res) throws ServletException, IOException {

String user = "root";

String pass = "root";

String url = "jdbc:derby://localhost:1527/bill";

String sql="select \* from price where item=?";

String str1 = req.getParameter("item"); // item name

String str2 = req.getParameter("qty"); // item quantity

String str3 = req.getParameter("add"); // submit button by name add

String str4 = req.getParameter("list"); // submit button by name list

res.setContentType("text/html");

PrintWriter out = res.getWriter();

if(str3 != null){

Cookie c1 = new Cookie(str1, str2);

res.addCookie(c1);

res.sendRedirect("index.html");

}

else if(str4 != null){

try{

Connection con = DriverManager.getConnection(url, user, pass);

PreparedStatement st = con.prepareStatement(sql);

Cookie clientCookies[] = req.getCookies();

out.print("<B>" + "Item ||" + " Price || " + " Quantity " );

for( int i = 0; i < clientCookies.length; i++){

st.setString(1, clientCookies[i].getName());

ResultSet rs = st.executeQuery();

if(rs.next()){

out.print("<B>" + clientCookies[i].getName() + " " + Integer.parseInt(clientCookies[i].getValue())\*rs.getInt("rate") +" " + clientCookies[i].getValue() + "</B><BR>");

}

}

}catch(Exception e){

System.out.println(e);

}

}

out.close( ) ;

}

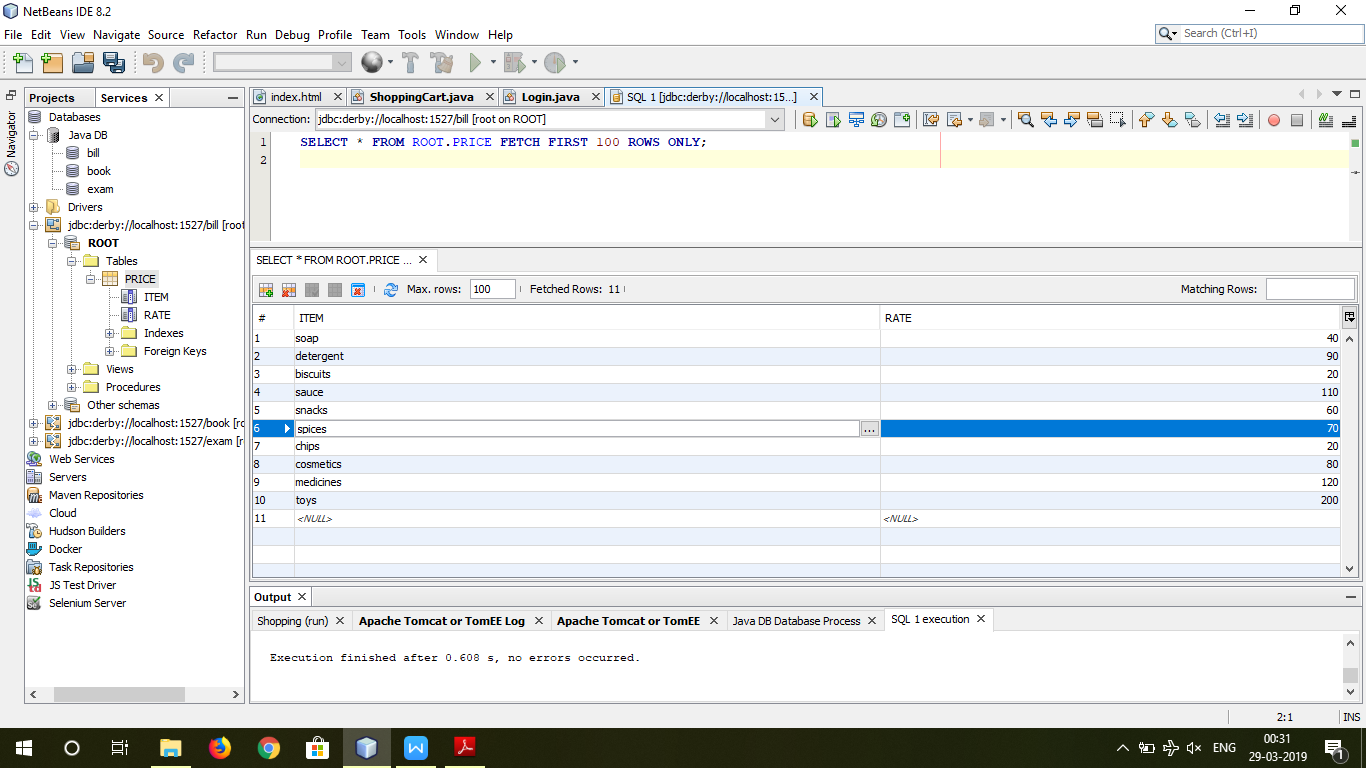
}

**SQL Database**

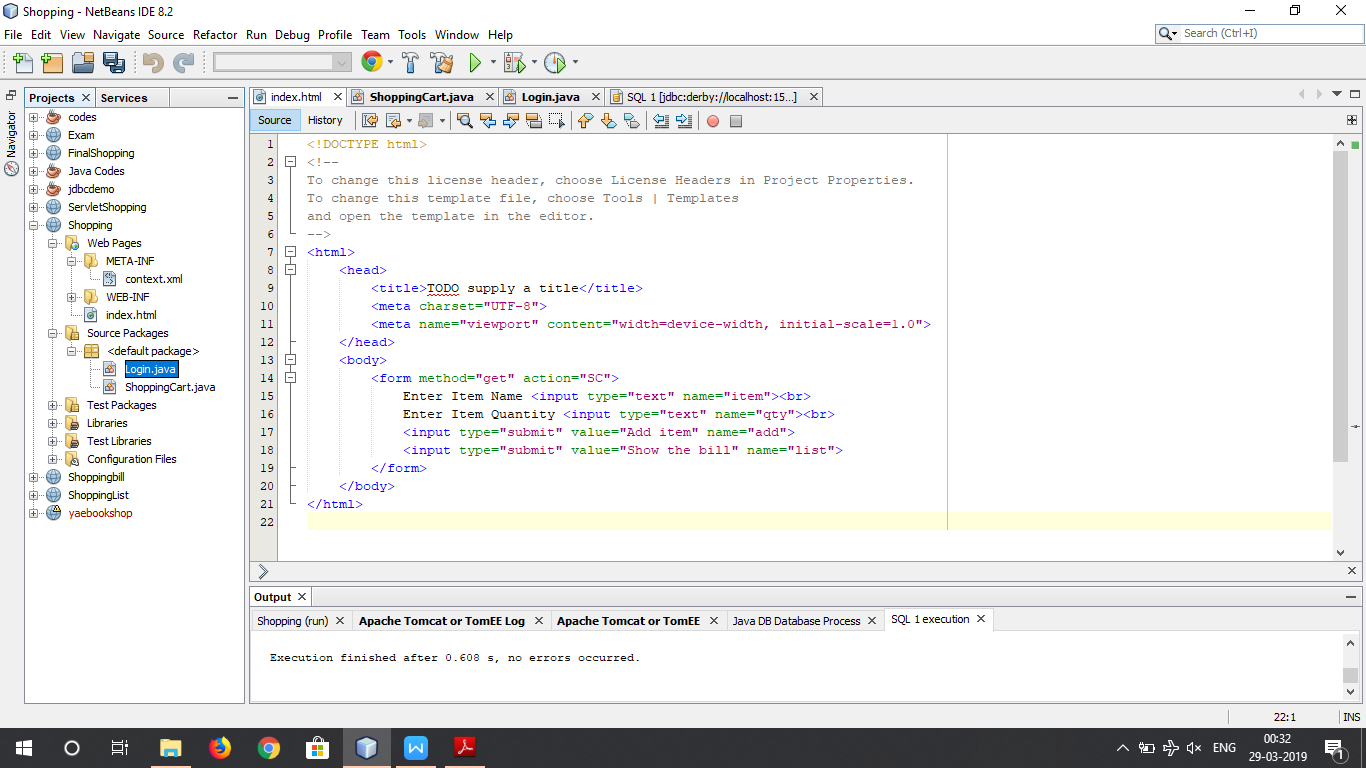
|  |  |
| --- | --- |
| Item | Rate |
| Soap | 40 |
| Detergent | 90 |
| Biscuits | 20 |
| Sauce | 110 |
| Snacks | 60 |
| Spices | 70 |
| Chips | 20 |
| Cosmetics | 80 |
| Medicines | 120 |
| Toys | 200 |

**Pictures**

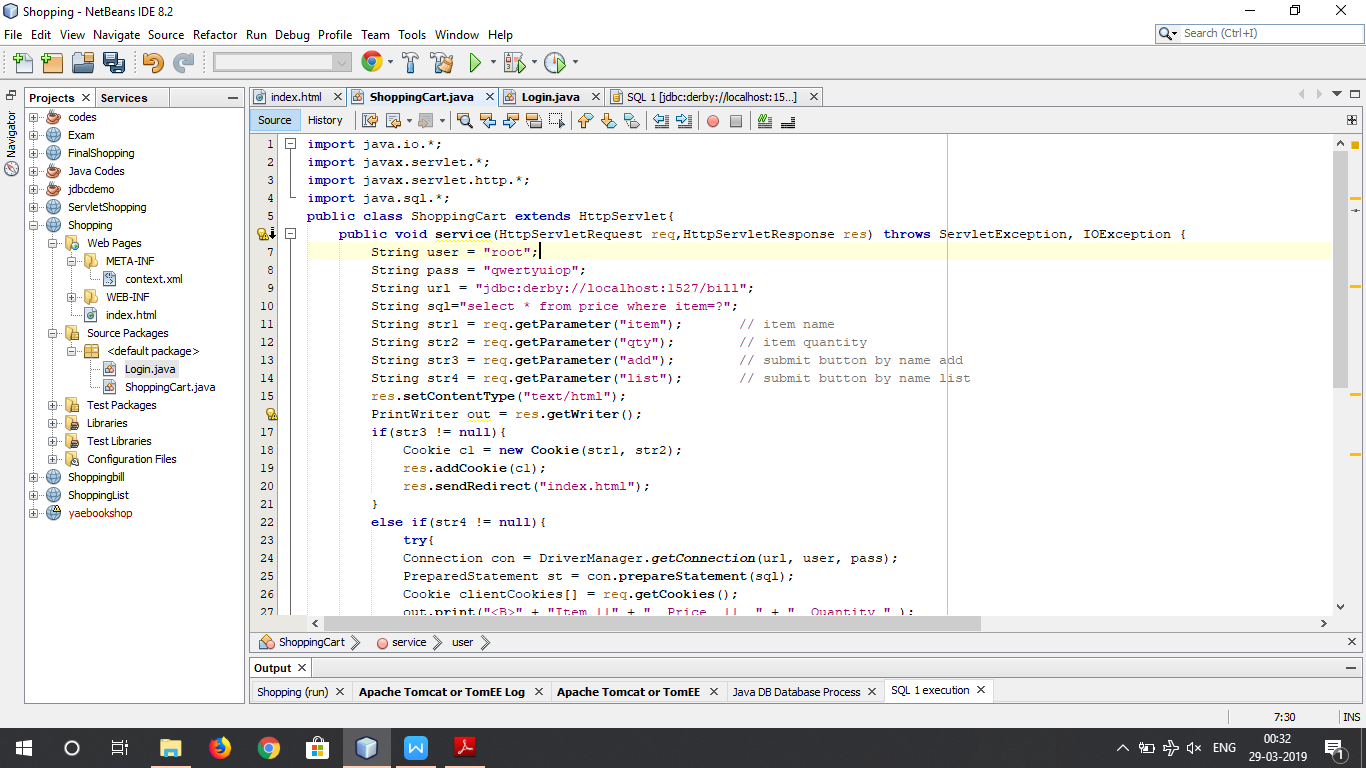
**SQL Database**

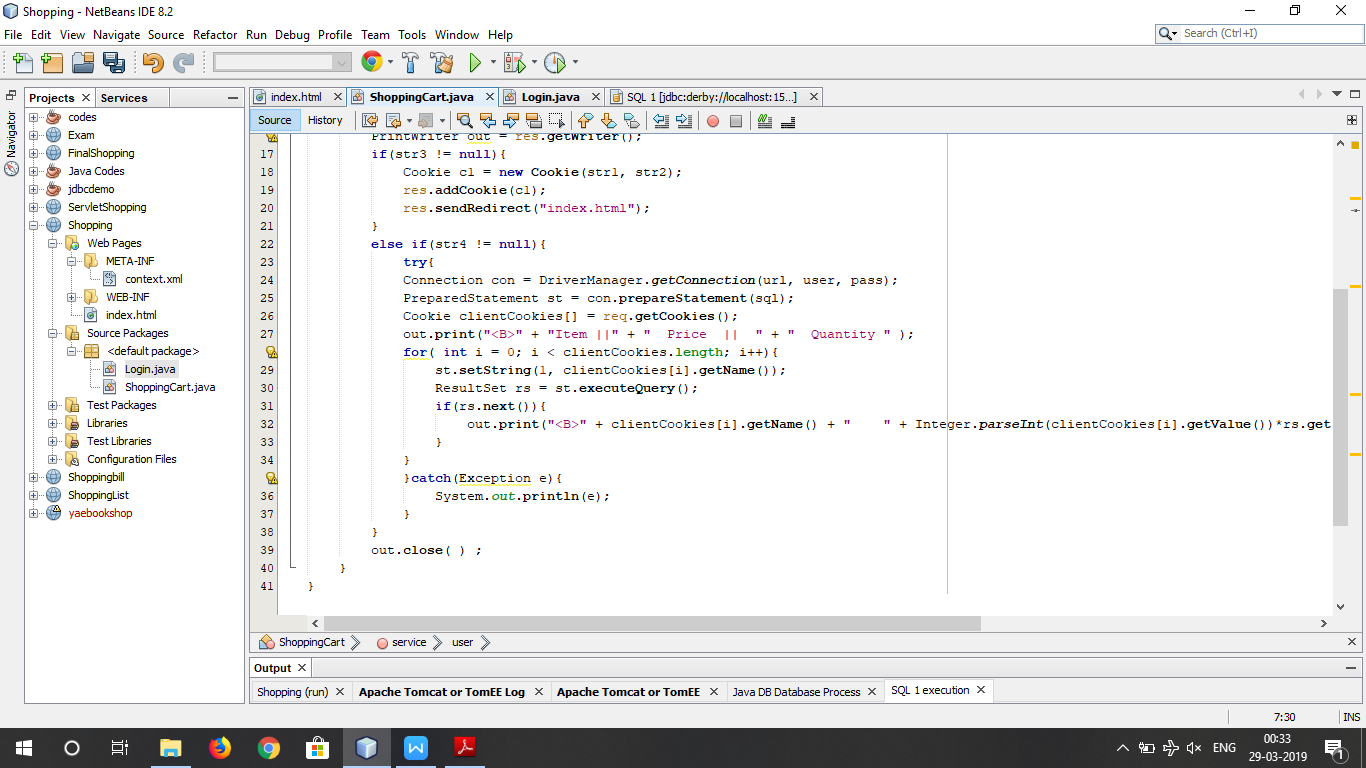


**Index.html**

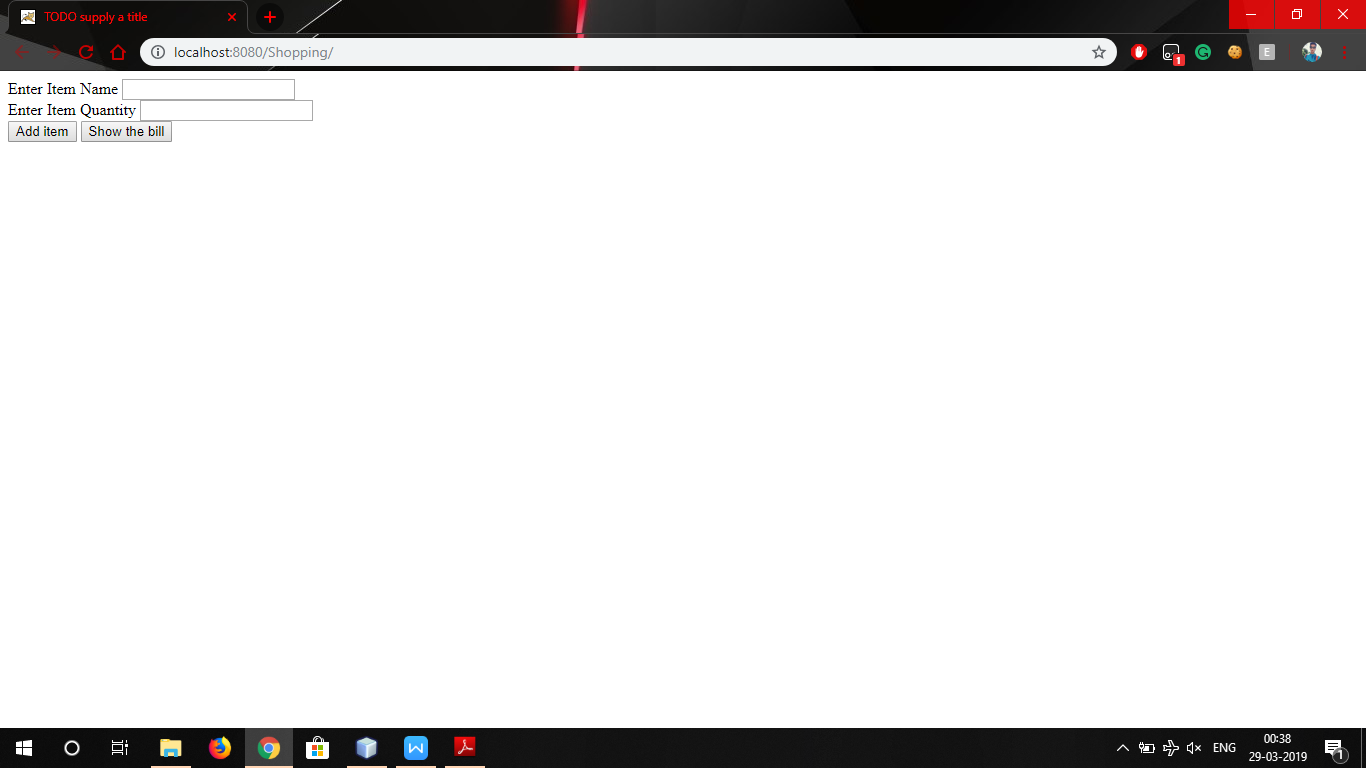


**ShoppingCart.java**

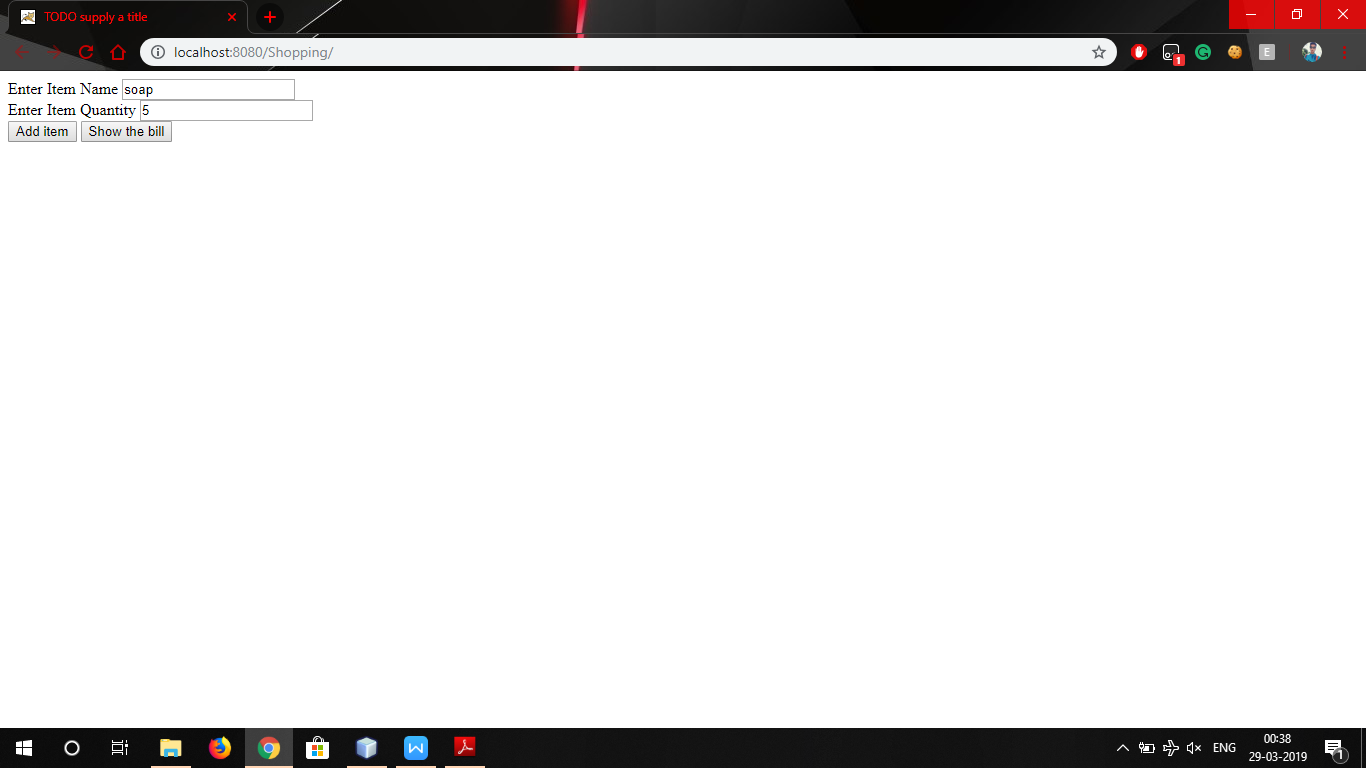




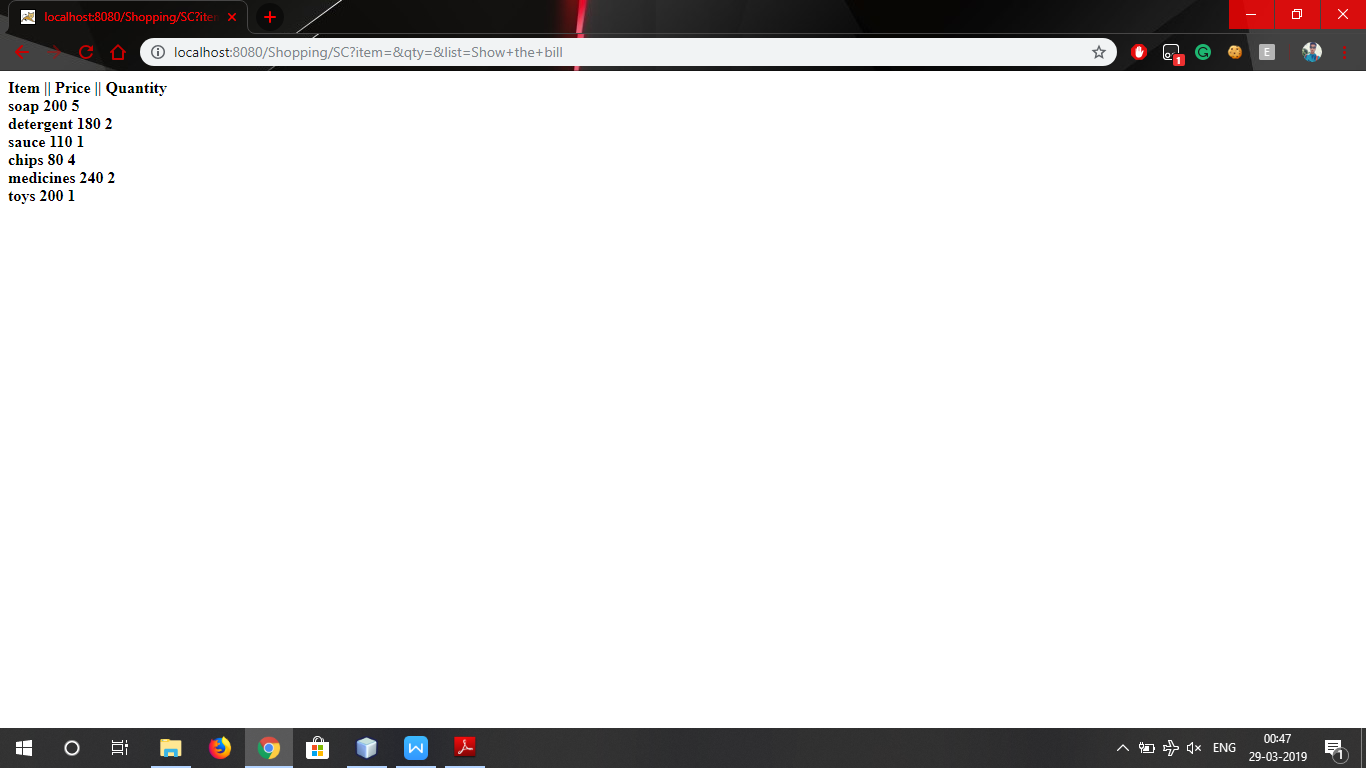
**Entry Screen**



**Entry**



**Total List**



**Inference**

It has been seen that all the items typed are compared with the database and are displayed on the final output screen.

The items displayed are framed in the classes as name of the item, price of the item and the quantity of the item.