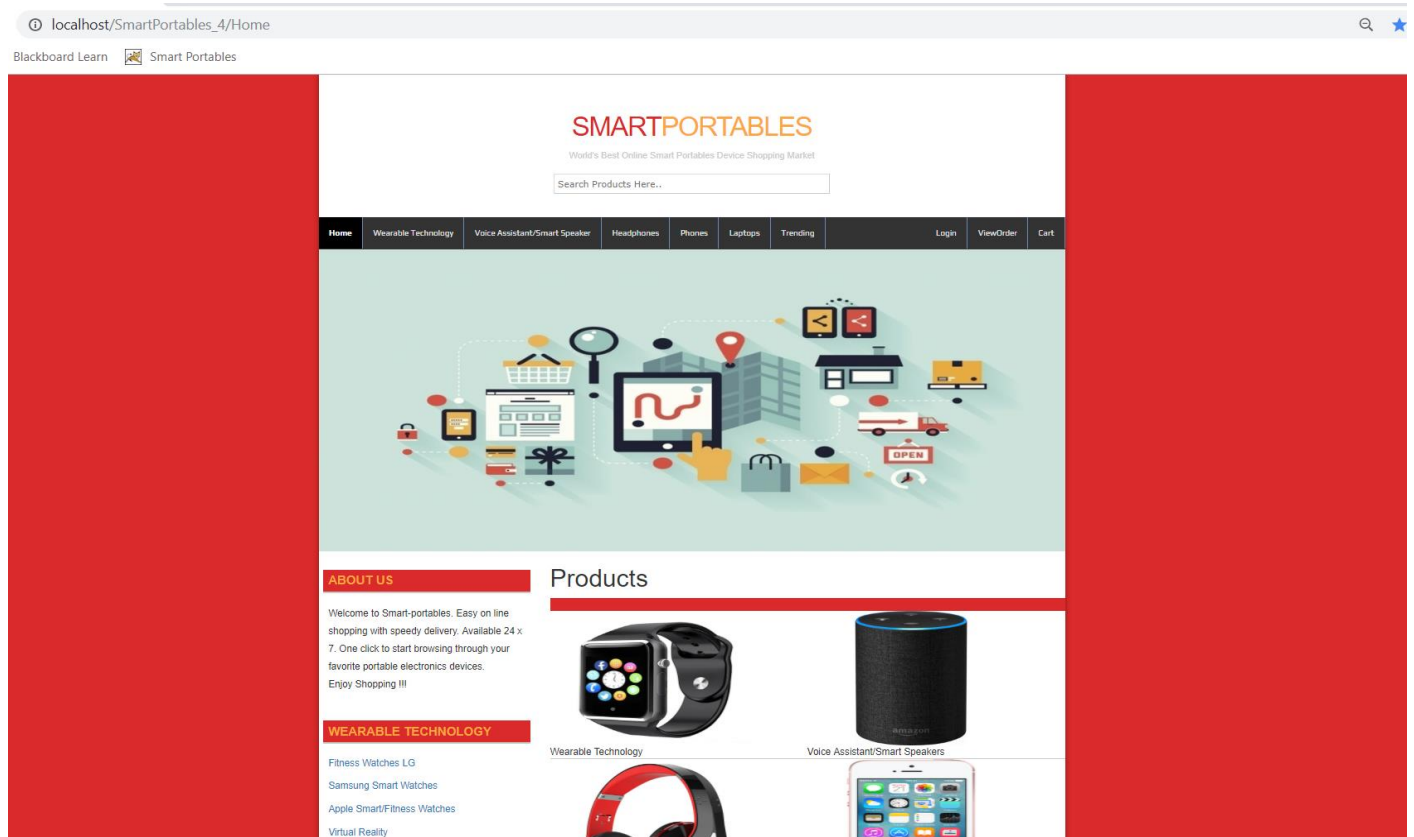
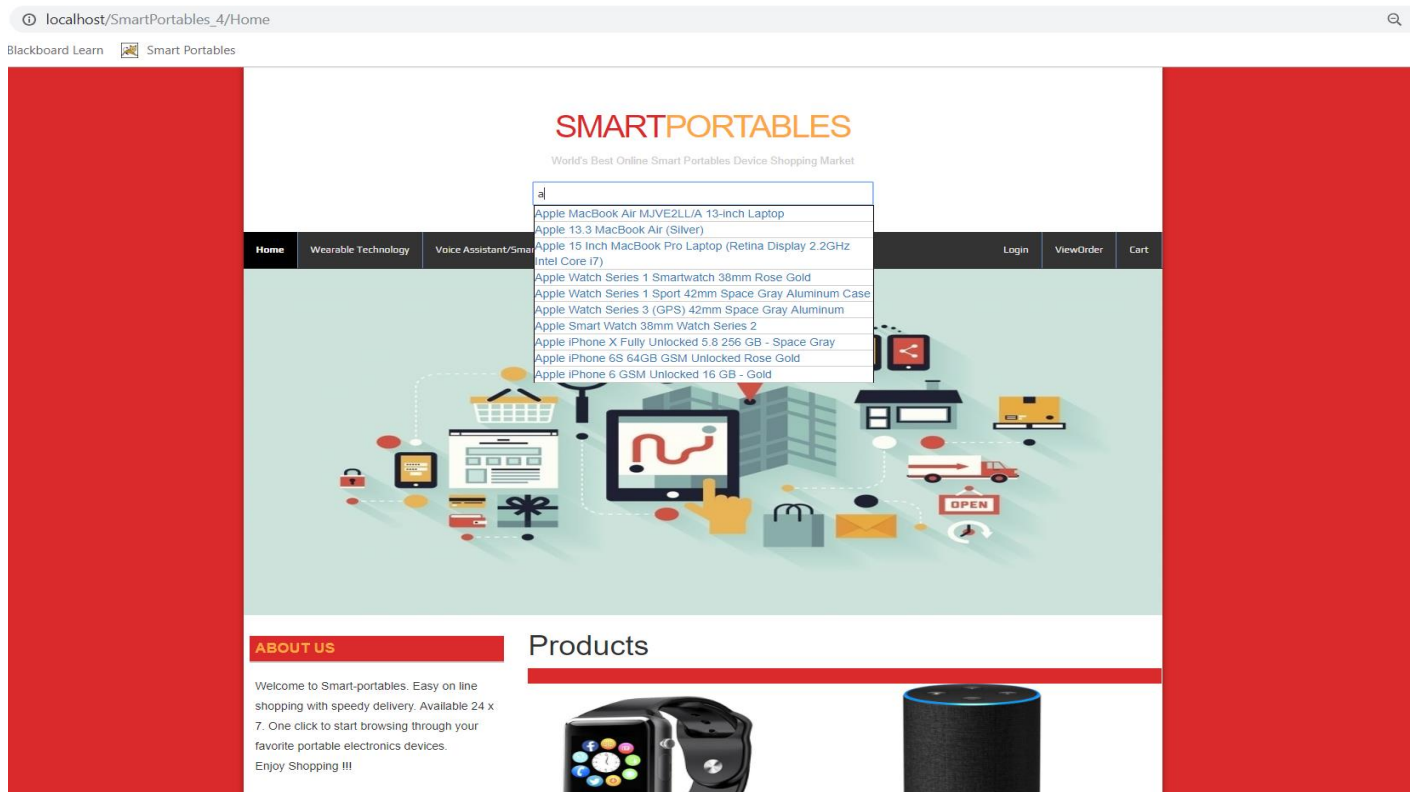


1. Home page with search:



2. Search Product with a:



3. Searched Product:

localhost/SmartPortables_4/AjaxUtilities?action=lookup&id=54

Blackboard Learn Smart Portables

SMARTPORTABLES

World's Best Online Smart Portables Device Shopping Market

[Home](#) [Wearable Technology](#) [Voice Assistant/Smart Speaker](#) [Headphones](#) [Phones](#) [Laptops](#) [Trending](#) [Login](#) [ViewOrder](#) [Cart](#)



ABOUT US

Welcome to Smart-portables. Easy on line shopping with speedy delivery. Available 24 x 7. One click to start browsing through your favorite portable electronics devices. Enjoy Shopping !!!

WEARABLE TECHNOLOGY

[Fitness Watches LG](#)
[Samsung Smart Watches](#)

Travel



- Name: Apple MacBook Air MJVE2LL/A 13-inch Laptop
- Price: \$1049
- Condition: New
- Manufacturer: Apple

[ADD TO CART](#) [WRITE REVIEW](#) [READ REVIEW](#)

4. Searched Product for Manager:

localhost/SmartPortables_4/AjaxUtilities?action=lookup&id=54

Blackboard Learn Smart Portables

SMARTPORTABLES

World's Best Online Smart Portables Device Shopping Market

[Home](#) [Add Product](#) [Update Product](#) [Delete Product](#) [Inventory](#) [Sales Report](#) [Trending](#) [Data Analytics](#) [Hi ashish, Logout](#) [ViewOrder](#)



ABOUT US

Welcome to Smart-portables. Easy on line shopping with speedy delivery. Available 24 x 7. One click to start browsing through your favorite portable electronics devices. Enjoy Shopping !!!

MANAGE PRODUCTS

[Add Product](#)
[Update Product](#)

Travel



- Name: Apple MacBook Air MJVE2LL/A 13-inch Laptop
- Price: \$1049
- Condition: New
- Manufacturer: Apple

[UPDATE PRODUCT](#) [DELETE PRODUCT](#)

5. Ajax Utilities Code:

```
4 import java.util.*;
5
6
7 public class AjaxUtilities extends HttpServlet {
8     String searchKeyword;
9     String action;
10    HashMap<String, Product> products ;
11    public void doGet(HttpServletRequest request,
12                      HttpServletResponse response)
13        throws ServletException, IOException {
14        StringBuffer sb = new StringBuffer();
15
16        boolean prod = false;
17        PrintWriter out= response.getWriter();
18        Utilities utility = new Utilities(request,out);
19        products = MySQLDataStoreUtilities.getProducts();
20        action = request.getParameter("action");
21        searchKeyword = request.getParameter("id");
22
23        if(searchKeyword!= "" || !searchKeyword.equals(""))
24        {
25
26
27            if(action.equals("complete"))
28            {
29                searchKeyword = searchKeyword.trim().toLowerCase();
30                for(Map.Entry<String,Product> entry : products.entrySet())
31                {
32                    Product product=(Product)entry.getValue();
33
34                    if(product.getName().toLowerCase().startsWith(searchKeyword))
35                    {
36                        sb.append("<product>");
37                        sb.append("<id>" + product.getId() + "</id>");
38                        sb.append("<name>" + product.getName() + "</name>");
39                        sb.append("</product>");
40                        prod = true;
41                    }
42                }
43                if(prod)
44                {
45                    response.setContentType("text/xml");
46                    response.setHeader("Cache-Control","no-cache");
47                    out.write("<products>" + sb.toString() + "</products>");
48                }
49                else
50                {
51                    response.setStatus(HttpServletResponse.SC_NO_CONTENT);
52                }
53            }
54            if(action.equals("lookup"))
55            {
56                request.setAttribute("productSearched",products.get(searchKeyword));
57                request.getRequestDispatcher("/ViewSearchProduct").forward(request, response);
58            }
59        }
60    }
61 }
```

6. Insert Product in the database and HashMap code:

```
public static void insertNewProduct(String productname,String producttype,String productcondition,String productmaker,
String productimage,String productquantity,String productprice,String productonsale,String productrebate) {
try {
getConnection();
int maxId=0;
String selectMaxIdQuery = new String();

selectMaxIdQuery ="select max(Id) as MaxId from Product;" ;

PreparedStatement pst1 = conn.prepareStatement(selectMaxIdQuery);
ResultSet rs = pst1.executeQuery();
if(rs.next())
{
    maxId = rs.getInt("MaxId")+1;
}
PreparedStatement pst = conn.prepareStatement("INSERT INTO Product(ProductName,Category,ProductCondition,Manufacturer,"
+ "ProductImage,Quantity,Price,OnSale,Rebate,Id) VALUES(?,?,?,?,?,?,?,?,?,?)");

pst.setString(1, productname);
pst.setString(2, producttype);
pst.setString(3, productcondition);
pst.setString(4, productmaker);
pst.setString(5,productimage );
pst.setInt(6,Integer.parseInt(productquantity));
pst.setDouble(7, new Double(productprice));
pst.setString(8,productonsale);
pst.setString(9,productrebate);
pst.setInt(10,maxId);

pst.execute();
conn.close();
Saxpaser.hm.put(maxId+"", new Product(productname,producttype,productcondition,productmaker,productimage,Integer.parseInt(productquantity)
,Integer.parseInt(productprice),productonsale,productrebate,maxId));
} catch (Exception e) {
e.printStackTrace();
}
}
```

7. Update Product in the database and HashMap code:

```
public static void updateProduct(String productname, String producttype, String productcondition,
    String productmaker, String productimage, String productquantity, String productprice, String productonsale,
    String productrebate,String id) {
    int rs=0;
    try
    {
        getConnection();
        String selectOrderQuery = new String();
        selectOrderQuery ="Update Product set ProductName=\""+productname+"\" "
            + ",Category=\""+producttype+"\" "
            + ",ProductCondition=\""+productcondition+"\" "
            + ",Manufacturer=\""+productmaker+"\" "
            + ",ProductImage=\""+productimage+"\" "
            + ",Quantity=\""+Integer.parseInt(productquantity)+"\" "
            + ",Price=\""+Double.parseDouble(productprice)+"\" "
            + ",OnSale=\""+productonsale+"\" "
            + ",Rebate=\""+productrebate+"\" "
            + "where Id = "+Integer.parseInt(id)+ " ";
        PreparedStatement pst = conn.prepareStatement(selectOrderQuery);
        rs = pst.executeUpdate();
        conn.close();
        Saxpaser.hm.put(id,new Product( productname, producttype, productcondition,
            productmaker, productimage, Integer.parseInt(productquantity), Integer.parseInt(productprice), productonsale,
            productrebate,Integer.parseInt(id)));
    }catch (Exception e) {
        e.printStackTrace();
    }
}
```

8. Delete Product in the database and HashMap code:

```
public static int deleteProduct(String productid) {

    int rs= 0;
    try
    {
        getConnection();
        String selectOrderQuery = new String();
        selectOrderQuery ="Delete from Product where Id = "+productid+ " ";
        PreparedStatement pst = conn.prepareStatement(selectOrderQuery);
        rs = pst.executeUpdate();
        conn.close();
    }catch (Exception e) {
        e.printStackTrace();
    }
    Saxpaser.hm.remove(productid);
    return rs;
}
```