

Tutorial week2

Mục tiêu:

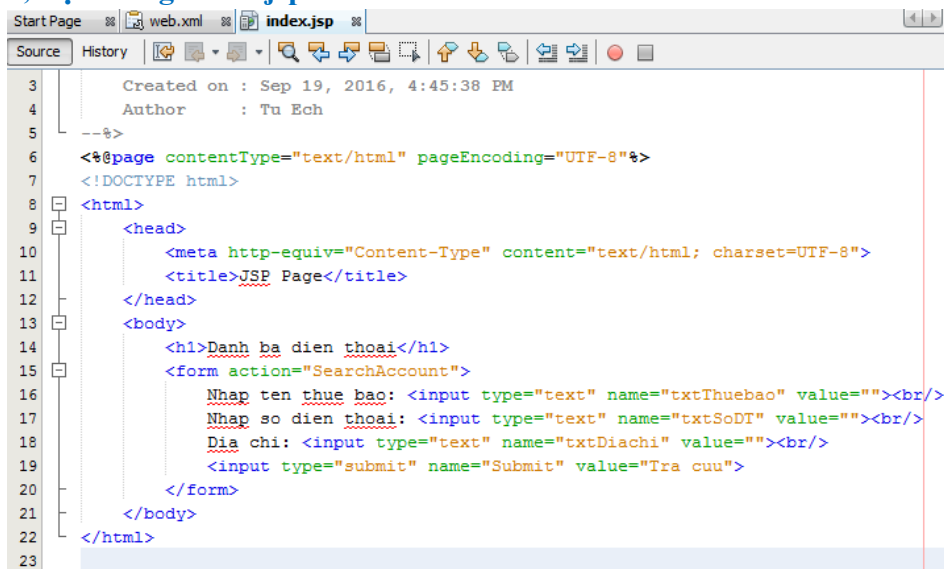
- ✓ Tạo web application tương tác với database
- ✓ Tạo web application có chức năng giỏ hàng

PHẦN I

Bài 1: Tạo ứng dụng tra cứu danh bạ điện thoại

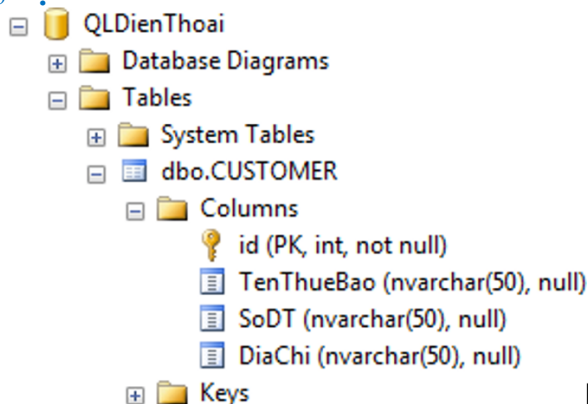
Tạo trang index.jsp dùng để tra cứu danh bạ điện thoại, viết một controller SearchAccount để tra cứu thông tin danh bạ .

1, Tạo trang index.jsp



```
3      Created on : Sep 19, 2016, 4:45:38 PM
4      Author    : Tu Ech
5      -->
6      <@page contentType="text/html" pageEncoding="UTF-8">
7      <!DOCTYPE html>
8      <html>
9      <head>
10         <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
11         <title>JSP Page</title>
12     </head>
13     <body>
14         <h1>Danh bạ dien thoai</h1>
15         <form action="SearchAccount">
16             Nhập tên thuê bao: <input type="text" name="txtThuebao" value=""><br/>
17             Nhập số điện thoại: <input type="text" name="txtSoDT" value=""><br/>
18             Địa chỉ: <input type="text" name="txtDiachi" value=""><br/>
19             <input type="submit" name="Submit" value="Tra cứu">
20         </form>
21     </body>
22 </html>
23
```

2, Tạo Database



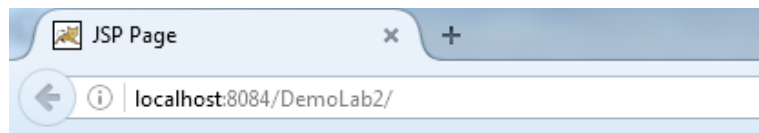
Results				
Messages				
	id	TenThueBao	SoDT	DiaChi
1	1	Le Van Phung	01683314129	311 No Trang Long, P.13
2	2	Nguyen Thi Tuong Vy	0903936117	Ninh Thuan
3	3	Le Quang Trung	09037829999	Phan Rang
4	4	Le Thi Hong Hanh	0167788889	Quan 12

3, Tạo controller (servlet) SearchAccount

```
Start Page  web.xml  index.jsp  SearchAccount.java
Source  History  [Icons]

22  /*
23  public class SearchAccount extends HttpServlet {
24
25      protected void processRequest(HttpServletRequest request, HttpServletResponse response)
26          throws ServletException, IOException {
27          response.setContentType("text/html;charset=UTF-8");
28          PrintWriter out = response.getWriter();
29          out.println("<html><head><title>SearchAccount</title></head>");
30          out.println("<body><h1>Ket qua tra dien thoai theo yeu cau cua ban :</h1>");
31          out.println("<table border=1 cellPadding=1 cellSpacing=1>");
32          String tentbao = request.getParameter("txtThuebao");
33          // xay dung lenh SQL
34          String newSQL = "SELECT * FROM CUSTOMER";
35          if (tentbao != null && tentbao.length() != 0) {
36              newSQL = newSQL + " where TenThueBao like '%" + tentbao + "%'";
37          }
38          String conStr = "jdbc:sqlserver://localhost:1433;databaseName=QLDienThoai";
39          Statement stmt = null;
40          ResultSet rs = null;
41          try {
42              Connection con=null;
43              Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
44              con = DriverManager.getConnection(conStr, "sa", "");
45              stmt = con.createStatement();
46              rs = stmt.executeQuery(newSQL);
47              out.println("<tr><th>So thu tu</th><th>Ten thue bao</th><th>So dien" +
48                  " thoai</th><th>Dia chi</th></tr>\n");
49              if (rs != null) {
50                  for ( int i = 1; rs.next(); ) {
51                      out.println("<tr>" + "<td>" + i + "</td>"
52                          + "<td>" + rs.getString(2) + "</td>"
53                          + "<td>" + rs.getString(3) + "</td>"
54                          + "<td>" + rs.getString(4) + "</td> </tr>\n");
55                  }
56              }
57              out.println("</table>");
58              rs.close();
59              stmt.close();
60              con.close();
61          } catch (Exception e) {
62              System.out.println("Error : " + e);
63          }
64      }
65
66      @Override
67      protected void doGet(HttpServletRequest request, HttpServletResponse response)
68          throws ServletException, IOException {
69          processRequest(request, response);
70      }
71
72      /** Handles the HTTP <code>POST</code> method ...8 lines */
73      @Override
74      protected void doPost(HttpServletRequest request, HttpServletResponse response)
75          throws ServletException, IOException {
76          processRequest(request, response);
77      }
78  }
```

Kết quả

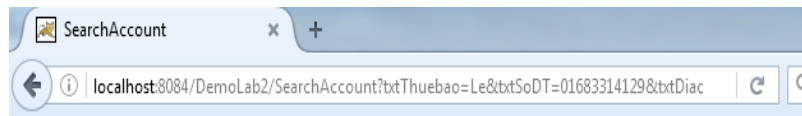


Danh ba dien thoai

Nhap ten thue bao:

Nhap so dien thoai:

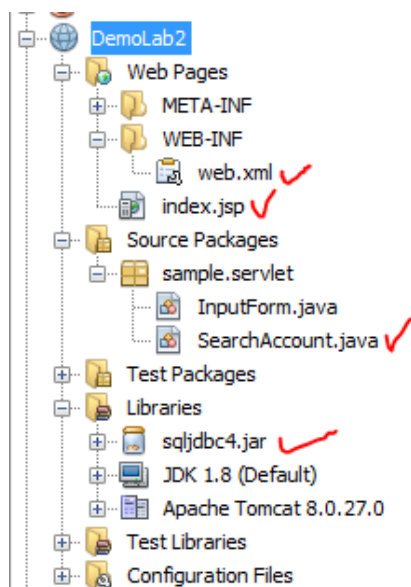
Dia chi:



Ket qua tra dien thoai theo yeu cau cua ban :

So thu tu	Ten thue bao	So dien thoai	Dia chi
1	Le Van Phung	01683314129	311 No Trang Long, P.13
2	Le Quang Trung	09037829999	Phan Rang
3	Le Thi Hong Hanh	0167788889	Quan 12

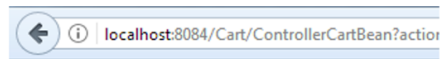
Cấu trúc chương trình



PHẦN II

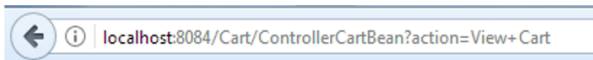
Bài 2: Tạo web app có chức năng cho người dùng có thể chọn những sản phẩm cần mua hoặc có thể remove, delete ra khỏi giỏ hàng những sản phẩm đã chọn.

Yêu cầu: Danh sách sản phẩm load từ database, viết theo kiến trúc MVC2



Products

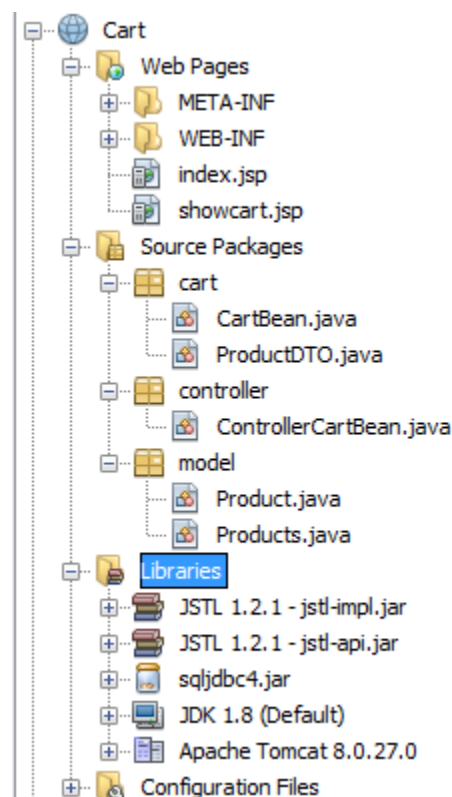
Ma SP	Ten SP	Gia	
SP001	Nokia 110	50.0	<input type="button" value="Add to Cart"/>
SP002	ASUS 2	300.0	<input type="button" value="Add to Cart"/>
SP003	ASUS 4	450.0	<input type="button" value="Add to Cart"/>
SP004	Iphone 3	20.0	<input type="button" value="Add to Cart"/>
SP005	Iphone 6	600.0	<input type="button" value="Add to Cart"/>
<input type="button" value="View Cart"/>			



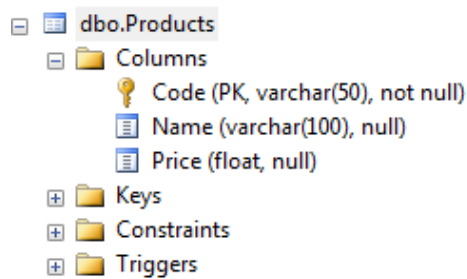
Your Shopping Cart

No.	Code	Name	Price	Quantity	Action
1	SP003	ASUS 4	450.0	<input type="text" value="1"/>	<input type="button" value="Remove"/>
2	SP002	ASUS 2	300.0	<input type="text" value="1"/>	<input type="button" value="Remove"/>
3	SP005	Iphone 6	600.0	<input type="text" value="2"/>	<input type="button" value="Remove"/>
Add more		<input type="button" value="Remove"/>			

Cấu trúc chương trình



Database



Results			
	Code	Name	Price
1	SP001	Nokia 110	50
2	SP002	ASUS 2	300
3	SP003	ASUS 4	450
4	SP004	Iphone 3	20
5	SP005	Iphone 6	600

View

Trang index.jsp

```
<body>
    <h1>Products </h1>
    <table border="1">
        <tr>
            <td>Ma SP</td>
            <td>Ten SP</td>
            <td>Gia</td>
        </tr>
    <%
        Products listSP = new Products();
        List<Product> list = listSP.showProduct("");
        for(Product sp:list){
            out.print("<form action=\"ControllerCartBean\">");
            out.print("<tr><td>" + sp.getCode() + "</td><td>" + sp.getName() + "</td>" +
                "<td>" + sp.getPrice() + "</td><td><input type=\"submit\" " +
                "value=\"Add to Cart\" name=\"action\" /></td>" +
                "<input type=\"hidden\" name=\"txtCode\" value=\"" + sp.getCode() + "\">" +
                "<input type=\"hidden\" name=\"txtName\" value=\"" + sp.getName() + "\">" +
                "<input type=\"hidden\" name=\"txtPrice\" value=\"" + sp.getPrice() +
                "\"></tr>");
            out.print("</form>");
        }
    %>
    </table>
    <form action="ControllerCartBean">
        <input type="submit" value="View Cart" name="action" />
    </form>
</body>
```

Trang showcart.jsp

```
18 <body>
19 <h1>Your Shopping Cart</h1>
20 <c:set var="shop" value="${sessionScope.SHOP}" />
21 <c:if test="${not empty shop}">
22 <table border="1">
23 <thead>
24 <tr>
25 <td>No.</td>
26 <td>Code</td>
27 <td>Name</td>
28 <td>Price</td>
29 <td>Quantity</td>
30 <td>Action</td>
31 </tr>
32 </thead>
33 <tbody>
34 <form action="ControllerCartBean">
35 <c:set var="count" value="0" />
36 <c:forEach var="rows" items="${shop}">
37 <c:set var="count" value="${count + 1}" />
38 <tr>
39 <td>${count}</td>
40 <td>${rows.value.sanpham.code}</td>
41 <td>${rows.value.sanpham.name}</td>
42 <td>${rows.value.sanpham.price}</td>
43 <td>${rows.value.quantity}</td>
44 <td><input type="checkbox" name="rmv"
45 <td>value="${rows.value.sanpham.code}" /></td>
46 </tr>
47 </c:forEach>
48 <tr>
49 <c:url var="add" value="ControllerCartBean">
50 <c:param name="action" value="AddMore"/>
51 </c:url>
52 <td colspan="2"><a href="${add}">Add more</a></td>
53 <td><input type="submit" value="Remove" name="action"/></td>
54 </tr>
55 </form>
56 </tbody>
57 </table>
58 </c:if>
59
60 </body>
```

```

20 public class Products {
21     public Products() {}
22     public void delete(String masp) {...15 lines }
37     public void insert(String masp, String tensp, float gia) {...17 lines }
54     public void update(String masp, String tensp, float gia) {...17 lines }
71     public List<Product> showProduct(String tensp){
72         try{
73             Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
74             String url="jdbc:sqlserver://localhost:1433;databaseName=AssignSOF301";
75             Connection con = DriverManager.getConnection(url, "sa", "");
76             String sql = "select * from Products";
77             if(tensp.length() > 0){
78                 sql += " where Name like '%" + tensp + "%'";
79             }
80             Statement stm = con.createStatement();
81             ResultSet rs = stm.executeQuery(sql);
82             List<Product> list = new ArrayList<Product>();
83             while(rs.next()){
84                 String code = rs.getString("Code");
85                 String name = rs.getString("Name");
86                 float price = rs.getFloat("Price");
87                 Product sp = new Product(code, name, price);
88                 list.add(sp);
89             }
90             return list;
91         } catch (Exception e) {
92             e.printStackTrace();
93         }
94         return null;
95     }
96 }

```

Model

```

12 public class Product {
13     private String code;
14     private String name;
15     private float price;
16     public Product(String code, String name, float price) {
17         this.code = code;
18         this.name = name;
19         this.price = price;
20     }
21     public Product() {
22     }
23     public String getCode() {
24         return code;
25     }
26     public void setCode(String code) {
27         this.code = code;
28     }
29     public String getName() {
30         return name;
31     }
32     public void setName(String name) {
33         this.name = name;
34     }
35     public float getPrice() {
36         return price;
37     }
38     public void setPrice(float price) {
39         this.price = price;
40     }
41 }

```

```

12  *
13  */
14 public class CartBean extends HashMap{
15     public void addSanPham(ProductDTO sp){
16         String key = sp.getSanpham().getCode();
17         if(this.containsKey(key)){
18             int oldQuantity = ((ProductDTO)this.get(key)).getQuantity();
19             ((ProductDTO)this.get(key)).setQuantity(oldQuantity+1);
20         }else{
21             this.put(sp.getSanpham().getCode(), sp);
22         }
23     }
24     public boolean removeSanPham(String code){
25         if(this.containsKey(code)){
26             this.remove(code);
27             return true;
28         }
29         return false;
30     }
31     public CartBean() {
32         super();
33     }
34 }

```



```

13 |      *
14 |      */
15 | public class ProductDTO implements Serializable{
16 |     private Product sanpham;
17 |     private int quantity;
18 |     public ProductDTO() {
19 |     }
20 |     public ProductDTO(Product sp) {
21 |         this.sanpham = sp;
22 |         this.quantity = 1;
23 |     }
24 |     public Product getSanpham() {
25 |         return sanpham;
26 |     }
27 |     public void setSanpham(Product sanpham) {
28 |         this.sanpham = sanpham;
29 |     }
30 |     public int getQuantity() {
31 |         return quantity;
32 |     }
33 |     public void setQuantity(int quantity) {
34 |         this.quantity = quantity;
35 |     }
36 | }

```

Controller

```

25 | public class ControllerCartBean extends HttpServlet {
26 |     protected void processRequest(HttpServletRequest request,
27 |     HttpServletResponse response) throws ServletException, IOException {
28 |         response.setContentType("text/html;charset=UTF-8");
29 |         PrintWriter out = response.getWriter();
30 |         try{
31 |             String action = request.getParameter("action");
32 |             if(action.equals("Add to Cart")){
33 |                 HttpSession session = request.getSession(true);
34 |                 CartBean shop = (CartBean)session.getAttribute("SHOP");
35 |                 if(shop==null){
36 |                     shop = new CartBean();
37 |                 }
38 |                 String code = request.getParameter("txtCode");
39 |                 String name = request.getParameter("txtName");
40 |                 String pri = request.getParameter("txtPrice");
41 |                 float fpri = Float.parseFloat(pri);
42 |                 Product s = new Product(code,name,fpri);
43 |                 ProductDTO sp = new ProductDTO(s);
44 |                 shop.addSanPham(sp);
45 |                 session.setAttribute("SHOP", shop);
46 |                 RequestDispatcher rd = request.getRequestDispatcher("index.jsp");
47 |                 rd.forward(request, response);

```

```

48         }else if(action.equals("View Cart")){
49             RequestDispatcher rd = request.getRequestDispatcher("showcart.jsp");
50             rd.forward(request, response);
51         }else if(action.equals("AddMore")){
52             RequestDispatcher rd = request.getRequestDispatcher("index.jsp");
53             rd.forward(request, response);
54         }else if(action.equals("Remove")){
55             String[] list = request.getParameterValues("rmv");
56             if(list !=null){
57                 HttpSession session = request.getSession();
58                 if(session !=null){
59                     CartBean shop = (CartBean)session.getAttribute("SHOP");
60                     if(shop !=null){
61                         for(int i=0;i<list.length;i++){
62                             shop.removeSanPham(list[i]);
63                         }
64                         session.setAttribute("SHOP", shop);
65                     }
66                 }
67             }
68             String url = "ControllerCartBean?action=View Cart";
69             RequestDispatcher rd = request.getRequestDispatcher(url);
70             rd.forward(request, response);
71         }
72     }catch(Exception e){
73         e.printStackTrace();
74     }finally{ out.close();}
75 }
76 + HttpServlet methods. Click on the + sign on the left to edit the code.
114 }

```