

Lab 02

Introduction to JAVA Servlet

Mục tiêu

Thực hiện các chức năng trong ứng dụng web

- Hiển thị tất cả sản phẩm
- Chi tiết sản phẩm
- Thêm mới sản phẩm
- Cập nhật sản phẩm
- Xóa sản phẩm

Phần I Bài tập step by step

Bài 1.1

Mục tiêu:

- Biết sử dụng dữ liệu giả định cho ứng dụng
- Điều khiển dữ liệu theo các luồng của ứng dụng:
 - Load all
 - Insert
 - Update
 - Delete
 - Find

STEP 1: Tạo class entity

Tạo một java class và đặt tên “Product”, package “entities”

```
package entities;  
  
/**  
 *  
 * @author HAITHANH  
 */
```

```
public class Product {
    private int proId;
    private String proName;
    private String producer;
    private int yearMaking;
    private float price;

    public Product() {
    }

    public Product(int proId, String proName, String producer, int yearMaking, float price) {
        this.proId = proId;
        this.proName = proName;
        this.producer = producer;
        this.yearMaking = yearMaking;
        this.price = price;
    }

    public int getProId() {
        return proId;
    }

    public void setProId(int proId) {
        this.proId = proId;
    }

    public String getProName() {
        return proName;
    }

    public void setProName(String proName) {
        this.proName = proName;
    }

    public String getProducer() {
        return producer;
    }

    public void setProducer(String producer) {
        this.producer = producer;
    }

    public int getYearMaking() {
        return yearMaking;
    }

    public void setYearMaking(int yearMaking) {
        this.yearMaking = yearMaking;
    }

    public float getPrice() {
```

```
        return price;
    }

    public void setPrice(float price) {
        this.price = price;
    }
}
```

STEP 2: Tạo dữ liệu cho project

- Tạo một java class và đặt tên “Data”, package “data”
- Khai báo một collection có phạm vi truy cập là public
- Khởi tạo dữ liệu ban đầu và lưu vào collection này.

```
package data;

import entities.Product;
import java.util.ArrayList;
import java.util.List;

/**
 *
 * @author HAITHANH
 */
public class Data {
    public static List<Product> listPro = new ArrayList<>();
}
```

STEP 3: Tạo Project web đặt tên “JspServlet_Lab2”

STEP 4: Tạo class DAO để cài đặt các hàm tương tác với dữ liệu.

- “New / Java Class”, đặt tên file là “ProductDAO”, package là “dao”. Lớp này có các hàm để tương tác với CSDL.

```
package dao;

import data.Data;
import entities.Product;
import java.util.List;
```

```
/**
 *
 * @author HAITHANH
 */
public class ProductDAO {
    public List<Product> getAllProducts(){
        return Data.listPro;
    }
    public Product getProductById(int proId){
        for (Product p : Data.listPro) {
            if(p.getProId()==proId){
                return p;
            }
        }
        return null;
    }

    public boolean addProduct(Product pro){
        return Data.listPro.add(pro);
    }
    public boolean updateProduct(Product pro){
        boolean bl = false;
        for (Product p : Data.listPro) {
            if(p.getProId()==pro.getProId()){
                Data.listPro.remove(p);
                Data.listPro.add(pro);
                bl = true;
                break;
            }
        }
        return bl;
    }
    public boolean deleteProduct(int proId){
        boolean bl = false;
        for (Product p : Data.listPro) {
            if(p.getProId()==proId){
                Data.listPro.remove(p);
                bl = true;
                break;
            }
        }
        return bl;
    }
}
```

STEP 5: Tạo một Servlet để khởi tạo dữ liệu ban đầu

- Right click Source Packages -> New -> Servlet, Đặt tên class "InitData", package "controller".
- Cài đặt lệnh trong hàm processRequest của lớp này như sau:

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    listPro.add(new Product(1, "Tivi 32 inch", "Toshiba", 2017, 4000000));
    listPro.add(new Product(2, "Tủ lạnh Sanyo", "Sanyo", 2017, 8000000));
    listPro.add(new Product(3, "Máy tính acer corei3", "Acer", 2017, 9000000));
    listPro.add(new Product(4, "Điện thoại iphone", "Apple", 2017, 14000000));
    listPro.add(new Product(5, "Truyện tranh", "Trương Bá Nam", 2017, 300000));
    listPro.add(new Product(6, "Truyện tiểu thuyết", "Nguyễn Thành Tâm", 2017,
400000));
    request.getRequestDispatcher("/LoadAllProducts").forward(request, response);
}
```

STEP 6: Tạo Servlet Load All dữ liệu để lấy về tất cả dữ liệu và gán thuộc tính cho đối tượng request

- Right click package "controller" -> New -> Servlet, Đặt tên class "LoadAllProducts", package "servlet".
- Cài đặt lệnh trong hàm processRequest của lớp này như sau:

```
protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    List<Product> allProducts = new ProductDAO().getAllProducts();
    request.setAttribute("listP", allProducts);
    request.getRequestDispatcher("index.jsp").forward(request, response);
}
```

STEP 7: Tạo trang "index.jsp" và trình bày nội dung cần hiển thị

- Xóa trang "index.xhtml" đi và click chuột phải vào "Web Pages" chọn "New / JSP...", đặt tên file là "index".
- Thêm nội dung sau vào giữa thẻ <body>

```
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <center>
      <h1>ALL PRODUCTS</h1>
      <table border="1" cellspacing="10px">
        <tr>
          <th>Product Id</th>
          <th>Product Name</th>
          <th>Producer</th>
          <th>Year Making</th>
          <th>Price</th>
        </tr>
        <c:forEach items="${listP}" var="p">
          <tr>
            <td>${p.proId}</td>
            <td>${p.proName}</td>
            <td>${p.producer}</td>
            <td>${p.yearMaking}</td>
            <td><fmt:formatNumber currencySymbol=""
value="${p.price}"></fmt:formatNumber> đ</td>
          </tr>
        </c:forEach>
      </table>
    </center>
  </body>
</html>
```

STEP 8: Cài đặt để Servlet chạy trước và chạy thử ứng dụng

- Để lấy được dữ liệu và đẩy lên trang "index.jsp" thì Servlet cần phải được chạy trước
- Click chuột phải vào project chọn properties. Chọn mục "Run", trong mục "Relative URL" điền giá trị sau vào: **"/InitData"** (đó chính là đường dẫn tương đối để gọi Servlet).
- Deploy project và chạy chương trình chúng ta sẽ được giao diện ở trang đầu tiên

Ta được giao diện trang "index.jsp" như sau:



ALL PRODUCTS

Product Id	Product Name	Producer	Year Making	Price	Details	Deletes
1	Tivi 32 inch	Toshiba	2017	4,000,000 đ	detail	delete
2	Tủ lạnh Sanyo	Sanyo	2017	8,000,000 đ	detail	delete
3	Máy tính acer corei3	Acer	2017	9,000,000 đ	detail	delete
4	Điện thoại iphone	Apple	2017	14,000,000 đ	detail	delete
5	Truyện tranh	Trương Bá Nam	2017	300,000 đ	detail	delete
6	Truyện tiểu thuyết	Nguyễn Thành Tâm	2017	400,000 đ	detail	delete

Ở đây chúng ta sử dụng thẻ

```
<fmt:formatNumber currencySymbol="" value="{p.price}"></fmt:formatNumber>
```

Để định dạng hiển thị dưới dạng tiền tệ, tuy nhiên chúng ta chưa sử dụng Locale để định nghĩa mã vùng hiển thị nên tạm thời thay thế kí tự tiền tệ của thẻ bằng rỗng và thêm vào kí tự 'đ' ở cuối để hiển thị là tiền tệ Việt Nam. Thẻ này cho phép chúng ta định dạng hiển thị các kiểu dữ liệu như: Số, tiền tệ, phần trăm.

Để sử dụng thẻ này, chúng ta phải thêm thư viện `jstl-1.2.jar` vào thư lục `libraries` của ứng dụng

Bài 1.2 Tạo chức năng chi tiết sản phẩm

STEP 1: Tạo liên kết để thực hiện chức năng detail

- Từ trang "index.jsp" thêm thẻ `<a>` để tạo một liên kết tới Servlet thực hiện detail

Trong thẻ tạo tiêu đề thêm thẻ tiêu đề sau:

```
<th>Details</th>
```

Trong thẻ `<forEach>` thêm thẻ sau:

```
<td><a href="DetailProduct?id={p.proId}">detail</a></td>
```

Khi chạy lên chúng ta sẽ có giao diện với liên kết như sau:

Product Id	Product Name	Producer	Year Making	Price	Details
1	Xe máy	Honda	2015	40,000,000 đ	Detail
2	Ô tô	Audi	2016	1,200,000,000 đ	Detail
3	Máy tính	Sony	2017	14,000,000 đ	Detail
4	Tivi	Toshiba	2015	5,000,000 đ	Detail
5	Tủ lạnh	Daewoo	2015	8,000,000 đ	Detail
6	Điều hòa	Funiki	2016	10,000,000 đ	Detail

STEP 2: Tạo Servlet và lấy chi tiết sản phẩm từ database

- Tạo một Servlet có tên “**DetailProduct**” trong cùng thư mục với servlet “**LoadAllProducts**”, và xóa khối lệnh try{...} trong hàm “**processRequest**” để thêm lệnh xử lý vào.

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    int proId = Integer.parseInt(request.getParameter("id"));
    Product productById = new ProductDAO().getProductById(proId);
    request.setAttribute("pro", productById);
    request.getRequestDispatcher("detailProduct.jsp").forward(request, response);
}
```

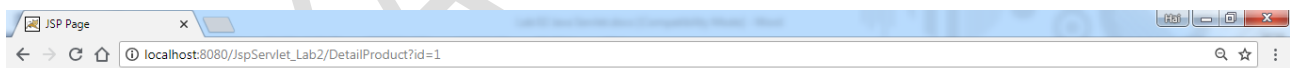
STEP 3: Tạo trang detailProduct.jsp và trình bày các nội dung chi tiết của sản phẩm.

```
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<center>
```



```
<h1>DETAIL PRODUCTS</h1>
<table border="0" cellspacing="10px">
  <tr>
    <td><b>Product Id:</b></td>
    <td>${pro.proId}</td>
  </tr>
  <tr>
    <td><b>Product Name:</b></td>
    <td>${pro.proName}</td>
  </tr>
  <tr>
    <td><b>Producer:</b></td>
    <td>${pro.producer}</td>
  </tr>
  <tr>
    <td><b>Year Making:</b></td>
    <td>${pro.yearMaking}</td>
  </tr>
  <tr>
    <td><b>Price:</b></td>
    <td><fmt:formatNumber currencySymbol=""
value="${pro.price}"></fmt:formatNumber> đ</td>
  </tr>
</table>
<a href="LoadAllProducts">Back</a>
</center>
</body>
</html>
```

Chúng ta được giao diện như sau:



DETAIL PRODUCTS

Product Id:	1
Product Name:	Tivi 32 inch
Producer:	Toshiba
Year Making:	2017
Price:	4,000,000 đ
	Update Back

STEP 4: Tạo liên kết để quay về trang index.jsp

- Thêm thẻ <a> vào trang detailProduct.jsp để quay về trang index.jsp như sau:

```
<a href="LoadAllProducts">Back</a>
```

- Thẻ này được thêm vào sau thẻ <table> khi hiển thị chi tiết các thông tin của sản phẩm.

Bài 1.3 Tạo chức năng thêm mới sản phẩm

STEP 1: Tạo liên kết để thực hiện chức năng thêm mới

- Từ trang "index.jsp" sau thẻ <table> thêm thẻ <a> vào sau thẻ <table> như sau để vào trang thêm mới

```
<a href="addProduct.jsp">Add New Product</a>
```

Product Id	Product Name	Producer	Year Making	Price	Details	Deletes
1	Tivi 32 inch	Toshiba	2017	4,000,000 đ	detail	delete
2	Tủ lạnh Sanyo	Sanyo	2017	8,000,000 đ	detail	delete
3	Máy tính acer corei3	Acer	2017	9,000,000 đ	detail	delete
4	Điện thoại iphone	Apple	2017	14,000,000 đ	detail	delete
5	Truyện tranh	Trương Bá Nam	2017	300,000 đ	detail	delete
6	Truyện tiểu thuyết	Nguyễn Thành Tâm	2017	400,000 đ	detail	delete

[Add New Product](#)

STEP 2: Tạo trang addProduct.jsp và tạo form để thêm mới một Vegetable

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<center>
<h1>ADD NEW PRODUCT</h1>
<form action="AddProduct">
<table border="1">
```

```
<tr>
  <td>Product Name: </td>
  <td>
    <input type="text" name="proName" required/>
  </td>
</tr>
<tr>
  <td>Producer: </td>
  <td>
    <input type="text" name="producer" required/>
  </td>
</tr>
<tr>
  <td>Year Making: </td>
  <td>
    <input type="text" name="yearMaking" required/>
  </td>
</tr>
<tr>
  <td>Price: </td>
  <td>
    <input type="text" name="price" required/>
  </td>
</tr>
<tr>
  <td></td>
  <td>
    <input type="submit" value="AddNew"/>
    <input type="reset" value="Clear"/>
  </td>
</tr>
</table>
</form>
</center>
</body>
</html>
```

Giao diện trang thêm mới:

ADD NEW PRODUCT

Product Name:	<input type="text"/>
Producer:	<input type="text"/>
Year Making:	<input type="text"/>
Price:	<input type="text"/>
	<input type="button" value="AddNew"/> <input type="button" value="Clear"/>

STEP 3: Tạo Servlet **AddProduct** và lấy dữ liệu từ form xuống

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    //Thiết lập kiểu kí tự để nhận được tiếng việt có dấu
    response.setContentType("text/html;charset=UTF-8");
    request.setCharacterEncoding("UTF-8");

    //Lấy dữ liệu từ form submit lên
    String proName = request.getParameter("proName");
    String producer = request.getParameter("producer");
    int yearMaking = Integer.parseInt(request.getParameter("yearMaking"));
    float price = Float.parseFloat(request.getParameter("Price"));

    //Gọi hàm xử lý trong DAO
    Product p = new Product();
    p.setProName(proName);
    p.setProducer(producer);
    p.setYearMaking(yearMaking);
    p.setPrice(price);

    boolean blAdd = new ProductDAO().addProduct(p);
    if(blAdd){
        //chuyển về trang index.jsp
        request.getRequestDispatcher("LoadAllProducts").forward(request, response);
    }else{
        //Có lỗi, gán biến lỗi và quay về trang addProduct.jsp
        request.setAttribute("status", "Add product failed!");
        request.getRequestDispatcher("addProduct.jsp").forward(request, response);
    }
}
```

- Thêm thẻ vào trang **"addProduct.jsp"** để hiển thị lỗi khi add new

```
<h3 style="color: red">${status}</h3>
```

Bài 1.4 Tạo chức năng cập nhật sản phẩm

STEP 1: Tạo liên kết để thực hiện chức năng detail

- Trong trang "detailProduct.jsp" thêm thẻ <a> gọi tới một Servlet để thực hiện hành động update

```
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<center>
<h1>DETAIL PRODUCTS</h1>
<table border="0" cellspacing="10px">
<tr>
<td><b>Product Id:</b></td>
<td>${pro.proId}</td>
</tr>
<tr>
<td><b>Product Name:</b></td>
<td>${pro.proName}</td>
</tr>
<tr>
<td><b>Producer:</b></td>
<td>${pro.producer}</td>
</tr>
<tr>
<td><b>Year Making:</b></td>
<td>${pro.yearMaking}</td>
</tr>
<tr>
<td><b>Price:</b></td>
<td><fmt:formatNumber currencySymbol=""
value="${pro.price}"></fmt:formatNumber> đ</td>
</tr>
<tr>
<td>
<a href="PreUpdateProduct?proId=${pro.proId}">Update</a>
</td>
<td>
<a href="LoadAllProducts">Back</a>
</td>
</tr>
</table>
</center>
</body>
</html>
```

```
</td>
</tr>
</table>
</center>
</body>
</html>
```

Giao diện sẽ như sau:

DETAIL PRODUCTS

Product Id: 1
Product Name: Xe máy
Producer: Honda
Year Making: 2015
Price: 40,000,000 đ
[Update](#) [Back](#)

STEP 2: Tạo Servlet và kết nối với database để lấy về chi tiết sản phẩm

- Tạo một Servlet có tên **"PreUpdateProduct"** trong package controller
- Thêm lệnh để xử lý trong hàm **"processRequest()"**:

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    int proId = Integer.parseInt(request.getParameter("proId"));
    Product product = new ProductDAO().getProductById(proId);
    request.setAttribute("pro", product);
    request.getRequestDispatcher("updateProduct.jsp").forward(request, response);
}
```

STEP 3: Tạo trang "updateProduct.jsp" và đẩy dữ liệu từ servlet lên để cập nhật

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
```

```
</head>
<body>
  <center>
    <h1>UPDATE PRODUCT</h1>
    <h3 style="color: red">${status}</h3>
    <form action="UpdateProduct">
      <table border="1">
        <tr>
          <td>Product Id:</td>
          <td>
            <input type="text" name="proId" value="${pro.proId}" disabled="true"/>
            <input type="hidden" name="proId" value="${pro.proId}"/>
          </td>
        </tr>
        <tr>
          <td>Product Name: </td>
          <td>
            <input type="text" name="proName" value="${pro.proName}" required/>
          </td>
        </tr>
        <tr>
          <td>Producer: </td>
          <td>
            <input type="text" name="producer" value="${pro.producer}" required/>
          </td>
        </tr>
        <tr>
          <td>Year Making: </td>
          <td>
            <input type="text" name="yearMaking" value="${pro.yearMaking}"
required/>
          </td>
        </tr>
        <tr>
          <td>Price: </td>
          <td>
            <input type="text" name="price" value="${pro.price}" required/>
          </td>
        </tr>
        <tr>
          <td></td>
          <td>
            <input type="submit" value="Update"/>
            <input type="reset" value="Clear"/>
          </td>
        </tr>
      </table>
    </form>
  </center>
```

```
</body>  
</html>
```

Giao diện trang cập nhật:

UPDATE PRODUCT

Product Id:	1
Product Name:	Xe máy
Producer:	Honda
Year Making:	2015
Price:	4.0E7
	<input type="button" value="Update"/> <input type="button" value="Clear"/>

STEP 4: Tạo Servlet “UpdateProduct” và thực hiện lấy dữ liệu từ form đẩy lên (giống như phần insert):

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException {  
    response.setContentType("text/html;charset=UTF-8");  
    request.setCharacterEncoding("UTF-8");  
  
    //Lấy dữ liệu từ form submit lên  
    int proId = Integer.parseInt(request.getParameter("proId"));  
    String proName = request.getParameter("proName");  
    String producer = request.getParameter("producer");  
    int yearMaking = Integer.parseInt(request.getParameter("yearMaking"));  
    float price = Float.parseFloat(request.getParameter("price"));  
  
    //Gọi hàm xử lý trong DAO  
    Product p = new Product();  
    p.setProId(proId);  
    p.setProName(proName);  
    p.setProducer(producer);  
    p.setYearMaking(yearMaking);  
    p.setPrice(price);  
  
    boolean blUpdate = new ProductDAO().updateProduct(p);  
    if(blUpdate){  
        //chuyển về trang index.jsp  
        request.getRequestDispatcher("LoadAllProducts").forward(request, response);  
    }else{
```



```
//Có lỗi, gán biến lỗi và quay về trang updateProduct.jsp
request.setAttribute("status", "Update product failed!");
request.getRequestDispatcher("updateProduct.jsp").forward(request, response);
}
}
```

Bài 1.5 Tạo chức năng xóa sản phẩm

STEP 1: Tạo liên kết để thực hiện hành động xóa

- Từ trang “index.jsp” thêm thẻ <a> vào để tạo liên kết xóa sản phẩm

Trong thẻ tạo tiêu đề thêm thẻ tiêu đề sau:

```
<th>Details</th>
```

Trong thẻ <forEach> thêm thẻ sau:

```
<td><a href="DetailProduct?id=${p.proId}">detail</a></td>
```

Giao diện khi chạy lên:



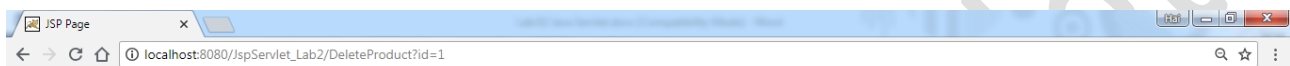
Product Id	Product Name	Producer	Year Making	Price	Details	Deletes
1	Tivi 32 inch	Toshiba	2017	4,000,000 đ	detail	delete
2	Tủ lạnh Sanyo	Sanyo	2017	8,000,000 đ	detail	delete
3	Máy tính acer corei3	Acer	2017	9,000,000 đ	detail	delete
4	Điện thoại iphone	Apple	2017	14,000,000 đ	detail	delete
5	Truyện tranh	Trương Bá Nam	2017	800,000 đ	detail	delete
6	Truyện tiểu thuyết	Nguyễn Thành Tâm	2017	400,000 đ	detail	delete

[Add New Product](#)

STEP 2: Tạo Servlet “DeleteProduct” và thêm lệnh vào để xử lý

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    //lấy về product id truyền từ liên kết đến
    int proId = Integer.parseInt(request.getParameter("id"));
    //Gọi hàm trong ProductDAO để thực hiện xóa
```

```
boolean blDelete = new ProductDAO().deleteProduct(proId);
if(blDelete){
    request.setAttribute("status", "Delete successfully for product with id: "+proId);
}else{
    request.setAttribute("status", "Delete failed for product with id: "+proId);
}
//Load lại dữ liệu và quay về trang index.jsp
List<Product> allProducts = new ProductDAO().getAllProducts();
request.setAttribute("listP", allProducts);
request.getRequestDispatcher("index.jsp").forward(request, response);
}
```



ALL PRODUCTS

Delete successfully for product with id: 1

Product Id	Product Name	Producer	Year Making	Price	Details	Deletes
2	Tủ lạnh Sanyo	Sanyo	2017	8,000,000 đ	detail	delete
3	Máy tính acer corei3	Acer	2017	9,000,000 đ	detail	delete
4	Điện thoại iphone	Apple	2017	14,000,000 đ	detail	delete
5	Truyện tranh	Trương Bá Nam	2017	300,000 đ	detail	delete
6	Truyện tiểu thuyết	Nguyễn Thành Tâm	2017	400,000 đ	detail	delete

[Add New Product](#)