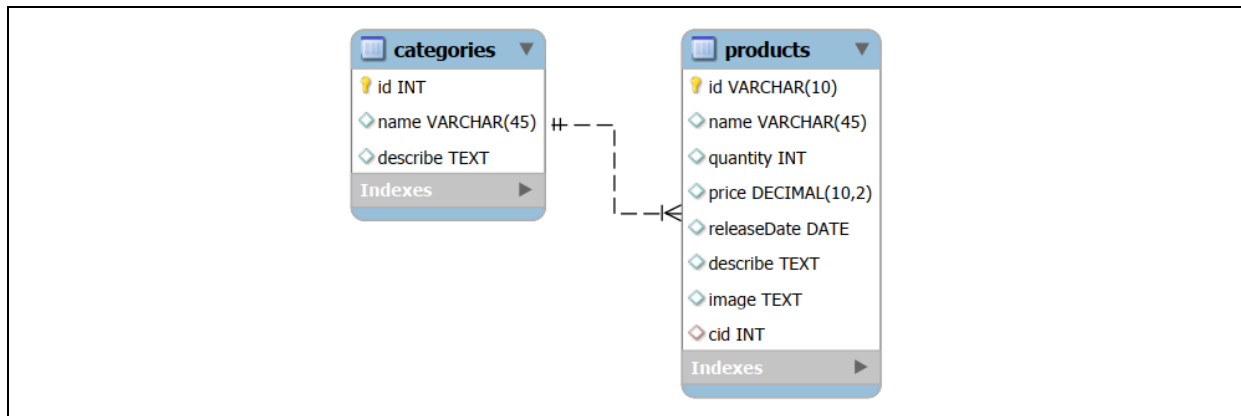





## BÀI THỰC HÀNH TUẦN 7


**Bài 1:** Xây dựng ứng dụng sau (Sử dụng 2 bảng categories và products)



Giao diện ứng dụng:

All iPhone9 Samsung Oppo Vsmart SS7 Iphone LG 8				
ID	Name	Price	Image	Category Name
ip1	iphone 12 series	19000.0		iPhone9
ip2	iphone 11 series	16000.0		iPhone9
ip3	iphone X series	13000.0		iPhone9

Khi click vào **All**: Liệt kê tất cả các sản phẩm

All iPhone9 Samsung Oppo Vsmart SS7 Iphone LG 8				
ID	Name	Price	Image	Category Name
vsm1	Vsmart Joy 4 3GB-64GB	13000.0		Vsmart

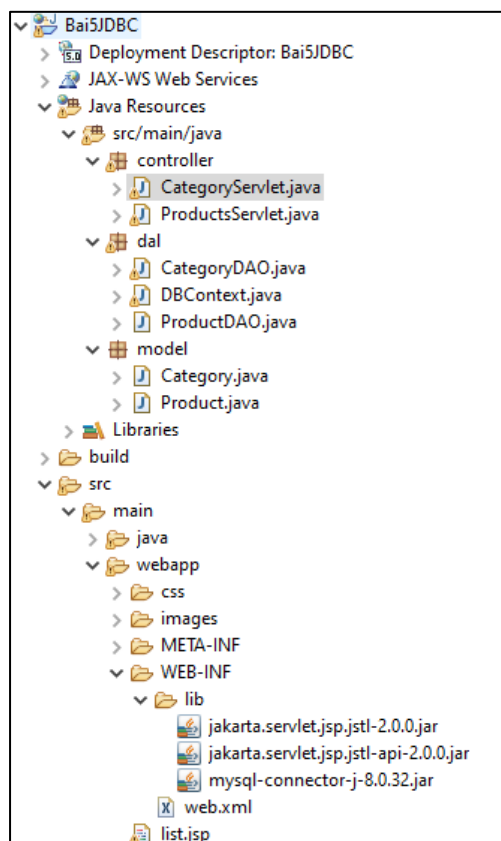
Khi click vào **Vsmart**: Liệt kê tất cả các sản phẩm của hãng Vsmart như hình

All iPhone9 Samsung Oppo Vsmart SS7 Iphone LG 8

**No Products**

Khi click vào hãng không có sản phẩm hiển thị **No Products**

Cấu trúc của Project như sau:



Tạo class **DBContext.java** trong package **dal**

```
public class DBContext {  
    protected Connection connection;  
  
    public DBContext() {  
        String name = "root";  
        String pass = "1234567890";  
        String url = "jdbc:mysql://localhost:3306/trading2023";  
  
        try {  
            Class.forName("com.mysql.jdbc.Driver");  
            connection = DriverManager.getConnection(url, name, pass);  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```

Tạo class **Category.java** trong **model**

```
public class Category {
    private int id;
    private String name;
    private String describe;

    public Category() {
    }

    public Category(int id, String name, String de
        this.id = id;
        this.name = name;
        this.describe = describe;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getDescribe() {
        return describe;
    }

    public void setDescribe(String describe) {
        this.describe = describe;
    }

    @Override
    public String toString() {
        return "Category [id=" + id + ", name="
        + name + ", describe=" + describe + "]";
    }
}
```

Tạo class **Product.java** trong **model** bao gồm các phương thức khởi tạo, phương thức get/set/ toString

```
import java.sql.Date;

public class Product {
    private String id;
    private String name;
    private int quantity;
    private double price;
    private Date releaseDate;
    private String describe;
    private String image;
    private Category category;

    public Product() {
    }

    public Product(String id, String name, int quantity, double price,
        Date releaseDate, String describe, String image, Category category) {
        this.id = id;
        this.name = name;
        this.quantity = quantity;
        this.price = price;
        this.releaseDate = releaseDate;
        this.describe = describe;
        this.image = image;
        this.category = category;
    }

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }
}
```

Tạo class **CategoryDAO.java** trong **dal**

```
import java.sql.PreparedStatement;

public class CategoryDAO extends DBContext {

    public List<Category> getAll(){
        List<Category> list = new ArrayList<>();
        String sql = "select * from `categories`";

        try {
            PreparedStatement st = connection.prepareStatement(sql);
            ResultSet rs = st.executeQuery();

            while(rs.next()) {
                Category c = new Category(rs.getInt("id"),
                    rs.getString("name"),
                    rs.getString("describe"));

                list.add(c);
            }
            st.close();
        } catch (Exception e) {
            System.out.println(e);
        }

        return list;
    }
}
```

```
public Category getCategoryById(int id){

    String sql = "select * from categories where id=?";

    try {
        PreparedStatement st = connection.prepareStatement(sql);
        st.setInt(1, id);
        ResultSet rs = st.executeQuery();

        if(rs.next()) {
            Category c = new Category(rs.getInt("id"),
                rs.getString("name"),
                rs.getString("describe"));

            return c;
        }

    } catch (Exception e) {
        System.out.println(e);
    }

    return null;
}
```

## Tạo class **ProductDAO.java** trong dal

```
public class ProductDAO extends DBContext {
    public List<Product> getProductByCid(int cid){
        List<Product> list = new ArrayList<>();
        CategoryDAO cd = new CategoryDAO();
        String sql = "SELECT `products`.`id`,\r\n"
            + "    `products`.`name`,\r\n"
            + "    `products`.`quantity`,\r\n"
            + "    `products`.`price`,\r\n"
            + "    `products`.`releaseDate`,\r\n"
            + "    `products`.`describe`,\r\n"
            + "    `products`.`image`,\r\n"
            + "    `products`.`cid`\r\n"
            + "FROM `trading2023`.`products`\r\n"
            + "where 1=1 ";
        if(cid!=0) {
            sql = sql + " and cid="+cid;
        }
        try {
            PreparedStatement ps = connection.prepareStatement(sql);
            //ps.setInt(1, cid);
            ResultSet rs = ps.executeQuery();

            String id, name, describe, image;
            Date releaseDate;
            int quantity;
            double price;
            while(rs.next()) {
                id = rs.getString("id");
                name = rs.getString("name");
                quantity = rs.getInt("quantity");
                price = rs.getDouble("price");
                releaseDate = rs.getDate("releaseDate");
                describe = rs.getString("describe");
                image = rs.getString("image");
                cid = rs.getInt("cid");
                Category c = cd.getCategoryById(cid);
                Product p = new Product(id,name,quantity,price,releaseDate,describe,image,c);
                list.add(p);
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
        return list;
    }
}
```

## Tạo servlet **CategoryServlet.java** trong controller

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    CategoryDAO cd = new CategoryDAO();
    List<Category> list = cd.getAll();
    request.setAttribute("data", list);
    request.getRequestDispatcher("list.jsp").forward(request, response);
}
```

## Tạo servlet **ProductsServlet.java** trong controller

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    String cid_s = request.getParameter("cid");
    int cid = Integer.parseInt(cid_s);
    ProductDAO pd = new ProductDAO();
    List<Product> list = pd.getProductByCid(cid);
    request.setAttribute("products", list);
    request.getRequestDispatcher("CategoryServlet").forward(request, response);
}
```

## Tạo list.jsp trong webapp

```
2 <%@ page language="java" contentType="text/html; charset=UTF-8"
3   pageEncoding="UTF-8"%>
4 <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
5 <!DOCTYPE html>
6 <html>
7 <head>
8   <meta charset="ISO-8859-1">
9   <title>Home Page</title>
10  <link rel="stylesheet" href="./css/style.css" />
11 </head>
12 <body>
13   <div class="container">
14     <div class="navbar">
15       <ul class="menu">
16         <li><a href="ProducServlet?cid=0">All</a></li>
17         <c:forEach items="${data}" var="c">
18           <li><a href="ProducServlet?cid=${c.id}">${c.name }</a></li>
19         </c:forEach>
20       </ul>
21     </div>
22     <div class="content">
23       <c:set var="list" value="${requestScope.products}"/>
24       <c:if test="${empty list }">
25         <h3>No Product</h3>
26       </c:if>
27       <c:if test="${!empty list }">
28         <table border="1px">
29           <tr>
30             <th>ID</th>
31             <th>NAME</th>
32             <th>PRICE</th>
33             <th>IMAGE</th>
34             <th>CATEGORY NAME</th>
35           </tr>
36           <c:forEach items="${list}" var="p">
37             <tr>
38               <td>${p.id}</td>
39               <td>${p.name}</td>
40               <td>${p.price}</td>
41               <td></td>
42               <td>${p.category.name}</td>
43             </tr>
44           </c:forEach>
45         </table>
46       </c:if>
47     </div>
48   </div>
49 </body>
50 </html>
```

Insert jstl core

Link css

