

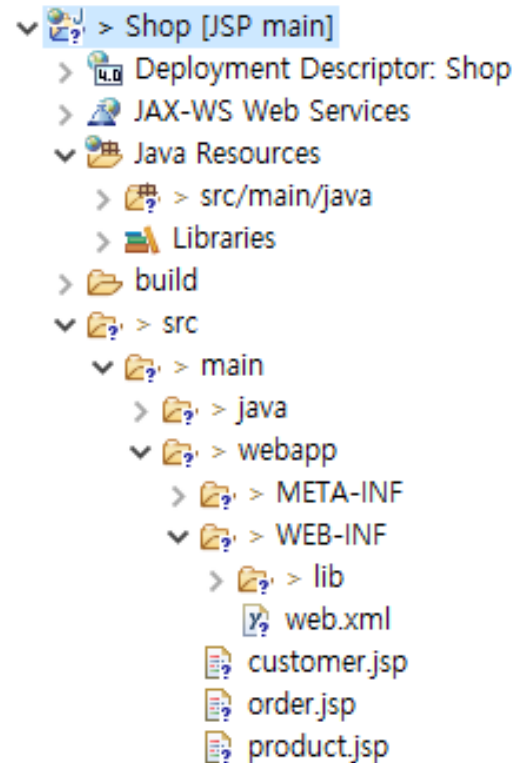
# SQL 응용

데이터베이스 구현

2022/10/25

라성준

# 문제1. 프로젝트 생성 및 구성



- 프로젝트명 : Shop
- WAS : Tomcat 9
- DB : java2\_shop
- 개발도구 : Eclipse, Workbench

## 문제2. 화면 구현

### 고객 목록

[고객목록](#) [주문목록](#) [상품목록](#)

아이디	이름	휴대폰	주소	가입일
c101	김유신	010-1234-1001	김해시 봉황동	2022-01-01
c102	김춘추	010-1234-1002	경주시 보문동	2022-01-02
c103	장보고	010-1234-1003	완도군 청산면	2022-01-03
c104	강감찬	010-1234-1004	서울시 마포구	2022-01-04
c106	정철	010-1234-1006	경기도 용인시	2022-01-06
c108	이순신	010-1234-1008	서울시 영등포구	2022-01-08
c109	송상현	010-1234-1009	부산시 동래구	2022-01-09
c110	정약용	010-1234-1010	경기도 광주시	2022-01-10

### 상품목록

[고객목록](#) [주문목록](#) [상품목록](#)

상품번호	상품명	재고량	가격	제조사	주문
1	새우깡	5000	1500	농심	주문
2	초코파이	2500	2500	오리온	주문
3	포카칩	3600	1700	오리온	주문
4	양파링	1250	1800	농심	주문
5	쪼리퐁	2200	0	크라운	주문
6	마카렛트	3500	3500	롯데	주문
7	뿌서뿌서	1650	1200	오뚜기	주문

### 주문목록

[고객목록](#) [주문목록](#) [상품목록](#)

주문번호	주문자	주문상품	주문수량	주문일
1	김춘추	포카칩	2	2022-07-01 13:15:10
2	김유신	양파링	1	2022-07-01 14:16:11
3	이순신	새우깡	1	2022-07-01 17:23:18
4	송상현	마카렛트	5	2022-07-02 10:46:36
5	김춘추	초코파이	1	2022-07-03 09:15:37
6	김유신	뿌서뿌서	3	2022-07-03 12:35:12
7	정약용	새우깡	2	2022-07-03 16:55:36
8	강감찬	초코파이	4	2022-07-04 14:23:23
9	김춘추	새우깡	3	2022-07-04 21:54:34

# 문제3. 기능구현

- ▶ JavaBean 생성
- ▶ orderBean 조건에 따라 추가 필드 생성

```
package bean;

public class OrderBean {

    private int orderNo;
    private String orderId;
    private int orderProduct;
    private int orderCount;
    private String orderDate;

    // 추가필드
    private String name;
    private String prodName;
    public int getOrderNo() {
        return orderNo;
    }
    public void setOrderNo(int orderNo) {
        this.orderNo = orderNo;
    }
    public String getOrderId() {
        return orderId;
    }
    public void setOrderId(String orderId) {
        this.orderId = orderId;
    }
    public int getOrderProduct() {
        return orderProduct;
    }
    public void setOrderProduct(int orderProduct) {
        this.orderProduct = orderProduct;
    }
    public int getOrderCount() {
        return orderCount;
    }
    public void setOrderCount(int orderCount) {
        this.orderCount = orderCount;
    }
}
```

```
package bean;

public class OrderBean {

    private int orderNo;
    private String orderId;
    private int orderProduct;
    private int orderCount;
    private String orderDate;

    // 추가필드
    private String name;
    private String prodName;
    public int getOrderNo() {
        return orderNo;
    }
    public void setOrderNo(int orderNo) {
        this.orderNo = orderNo;
    }
    public String getOrderId() {
        return orderId;
    }
    public void setOrderId(String orderId) {
        this.orderId = orderId;
    }
    public int getOrderProduct() {
        return orderProduct;
    }
    public void setOrderProduct(int orderProduct) {
        this.orderProduct = orderProduct;
    }
    public int getOrderCount() {
        return orderCount;
    }
    public void setOrderCount(int orderCount) {
        this.orderCount = orderCount;
    }
}
```

```
package bean;

public class ProductBean {

    private String prodNo;
    private String prodName;
    private String stock;
    private int price;
    private String company;
    public String getProdNo() {
        return prodNo;
    }
    public void setProdNo(String prodNo) {
        this.prodNo = prodNo;
    }
    public String getProdName() {
        return prodName;
    }
    public void setProdName(String prodName) {
        this.prodName = prodName;
    }
    public String getStock() {
        return stock;
    }
    public void setStock(String stock) {
        this.stock = stock;
    }
    public int getPrice() {
        return price;
    }
    public void setPrice(int price) {
        this.price = price;
    }
    public String getCompany() {
        return company;
    }
    public void setCompany(String company) {
        this.company = company;
    }
}
```

# 문제3 기능구현

```
package config;

import java.sql.Connection;

public class DBCP {

    private static DataSource ds = null;

    public static Connection getConnection() throws NamingException, SQLException {

        if(ds == null) {
            ds = (DataSource) new InitialContext().lookup("java:comp/env/dbcp_java2_shop");
        }

        return ds.getConnection();
    }
}
```

```
<Resource
    name="dbcp_java2_shop"
    auth="Container"
    type="javax.sql.DataSource"
    driverClassName="com.mysql.cj.jdbc.Driver"
    url="jdbc:mysql://127.0.0.1:3306/java2_shop"
    username="root"
    password="1234"
    maxTotal="16"
    maxIdle="16"
    maxWaitMillis="3000"
/>
```

```
</Context>
```

- DBCP 생성
- 커넥션 풀 생성

# 문제3. 기능구현

```
<%page import="java.sql.ResultSet"%>
<%page import="java.sql.Statement"%>
<%page import="config.DBCP"%>
<%page import="java.sql.Connection"%>
<% page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%

List<CustomerBean> customers = null;

try{
    Connection conn = DBCP.getConnection();

    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM `customer`");

    customers = new ArrayList<>();

    while(rs.next()){
        CustomerBean cb = new CustomerBean();
        cb.setCustId(rs.getString(1));
        cb.setName(rs.getString(2));
        cb.setHp(rs.getString(3));
        cb.setAddr(rs.getString(4));
        cb.setDate(rs.getString(5));

        customers.add(cb);
    }

    rs.close();
    stmt.close();
    conn.close();

}
```

```
}catch(Exception e){
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>customer</title>
</head>
<body>
    <h3>고객 목록</h3>
    <a href="#">고객목록</a>
    <a href="/order.jsp">주문목록</a>
    <a href="/product.jsp">상품목록</a>

    <table border="1">
        <tr>
            <th>아이디</th>
            <th>이름</th>
            <th>휴대폰</th>
            <th>주소</th>
            <th>가입일</th>
        </tr>
        <%
        for(CustomerBean cb: customers){
        %>
            <tr>
                <td><%= cb.getCustId() %></td>
                <td><%= cb.getName() %></td>
                <td><%= cb.getHp() %></td>
                <td><%= cb.getAddr() %></td>
                <td><%= cb.getDate() %></td>
            </tr>
        <% } %>
    </table>

}
```

```
<%page import="java.sql.PreparedStatement"%>
<%page import="bean.OrderBean"%>
<%page import="java.util.List"%>
<%page import="java.util.ArrayList"%>
<%page import="java.sql.ResultSet"%>
<%page import="java.sql.Statement"%>
<%page import="org.apache.tomcat.dbcp.dbcp2.PoolingConnection.StatementType"%>
<%page import="config.DBCP"%>
<%page import="java.sql.Connection"%>
<% page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%

List<OrderBean> orders = null;

try{
    Connection conn = DBCP.getConnection();

    String sql = "SELECT a.`orderNo`, b.`name`, c.`prodName`, a.`orderCount`, a.`orderDate` "
        + "from `order` AS a "
        + "join `customer` AS b ON a.orderId = b.custId "
        + "join `product` AS c ON a.orderProduct = c.prodNo";

    PreparedStatement pstmt = conn.prepareStatement(sql);
    ResultSet rs = pstmt.executeQuery();

    orders = new ArrayList<>();

    while(rs.next()){
        OrderBean ob = new OrderBean();
        ob.setOrderNo(rs.getInt(1));
        ob.setName(rs.getString(2));
        ob.setProdName(rs.getString(3));
        ob.setOrderCount(rs.getInt(4));
        ob.setOrderDate(rs.getString(5));

        orders.add(ob);
    }

}

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>order</title>
</head>
<body>
    <h3>주문목록</h3>
    <a href="#">고객목록</a>
    <a href="#">주문목록</a>
    <a href="#">상품목록</a>

    <table border="1">
        <tr>
            <th>주문번호</th>
            <th>주문자</th>
            <th>주문상품</th>
            <th>주문수량</th>
            <th>주문일</th>
        </tr>
        <%
        for(OrderBean ob : orders){
        %>
            <tr>
                <td><%= ob.getOrderNo() %></td>
                <td><%= ob.getName() %></td>
                <td><%= ob.getProdName() %></td>
                <td><%= ob.getOrderCount() %></td>
                <td><%= ob.getOrderDate() %></td>
            </tr>
        <% } %>
    </table>

</body>
</html>
```

```
<%page import="bean.ProductBean"%>
<%page import="java.util.List"%>
<%page import="java.util.ArrayList"%>
<%page import="java.sql.ResultSet"%>
<%page import="java.sql.Statement"%>
<%page import="config.DBCP"%>
<%page import="java.sql.Connection"%>
<% page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%

List<ProductBean> products = null;

try{
    Connection conn = DBCP.getConnection();

    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM `product`");

    products = new ArrayList<>();

    while(rs.next()){
        ProductBean pb = new ProductBean();
        pb.setProdNo(rs.getString(1));
        pb.setProdName(rs.getString(2));
        pb.setStock(rs.getString(3));
        pb.setPrice(rs.getInt(4));
        pb.setCompany(rs.getString(5));

        products.add(pb);
    }

    rs.close();
    stmt.close();
    conn.close();

}catch(Exception e){

<!--
<body>
    <h3>상품목록</h3>
    <a href="#">고객목록</a>
    <a href="#">주문목록</a>
    <a href="#">상품목록</a>

    <table border="1">
        <tr>
            <th>상품번호</th>
            <th>상품명</th>
            <th>재고량</th>
            <th>가격</th>
            <th>제조사</th>
            <th>주문</th>
        </tr>
        <%
        for(ProductBean pb : products){
        %>
            <tr>
                <td><%= pb.getProdNo() %></td>
                <td><%= pb.getProdName() %></td>
                <td><%= pb.getStock() %></td>
                <td><%= pb.getPrice() %></td>
                <td><%= pb.getCompany() %></td>
                <td>
                    <form>
                        <input type="submit" value="구매">
                    </form>
                </td>
            </tr>
        <% } %>
    </table>

</body>
</html>
```

- ▶ customer(고객) 목록 화면 기능 구현
- ▶ Order(주문) 목록 화면 기능 구현
- ▶ Product(상품) 주문하기 기능 구현

## 문제4. 실행

### 고객 목록

[고객목록](#) [주문목록](#) [상품목록](#)

아이디	이름	휴대폰	주소	가입일
c101	김유신	010-1234-1001	김해시 봉황동	2022-01-01
c102	김춘추	010-1234-1002	경주시 보문동	2022-01-02
c103	장보고	010-1234-1003	완도군 청산면	2022-01-03
c104	강감찬	010-1234-1004	서울시 마포구	2022-01-04
c106	정철	010-1234-1006	경기도 용인시	2022-01-06
c108	이순신	010-1234-1008	서울시 영등포구	2022-01-08
c109	송상현	010-1234-1009	부산시 동래구	2022-01-09
c110	정약용	010-1234-1010	경기도 광주시	2022-01-10

### 주문목록

[고객목록](#) [주문목록](#) [상품목록](#)

주문번호	주문자	주문상품	주문수량	주문일
1	김춘추	포카칩	2	2022-07-01 13:15:10
2	김유신	양파링	1	2022-07-01 14:16:11
3	이순신	새우깡	1	2022-07-01 17:23:18
4	송상현	마카렛트	5	2022-07-02 10:46:36
5	김춘추	초코파이	1	2022-07-03 09:15:37
6	김유신	뿌셔뿌셔	3	2022-07-03 12:35:12
7	정약용	새우깡	2	2022-07-03 16:55:36
8	강감찬	초코파이	4	2022-07-04 14:23:23
9	김춘추	새우깡	3	2022-07-04 21:54:34

### 상품목록

[고객목록](#) [주문목록](#) [상품목록](#)

상품번호	상품명	재고량	가격	제조사	주문
1	새우깡	5000	1500	농심	<a href="#">주문</a>
2	초코파이	2500	2500	오리온	<a href="#">주문</a>
3	포카칩	3600	1700	오리온	<a href="#">주문</a>
4	양파링	1250	1800	농심	<a href="#">주문</a>
5	쥬리풍	2200	0	크라운	<a href="#">주문</a>
6	마카렛트	3500	3500	롯데	<a href="#">주문</a>
7	뿌셔뿌셔	1650	1200	오뚜기	<a href="#">주문</a>

- ▶ customer(고객) 목록 실행 화면 첨부
- ▶ Order(주문) 목록 실행 화면 첨부
- ▶ Product(상품) 목록 실행 화면 첨부

감사합니다