

데이터베이스 & SQL

SQL 응용

날짜 : 2022/ 10/ 25 (화)

이름 : 김재준

목차

› 프로젝트 생성 및 구성

› 화면 구현

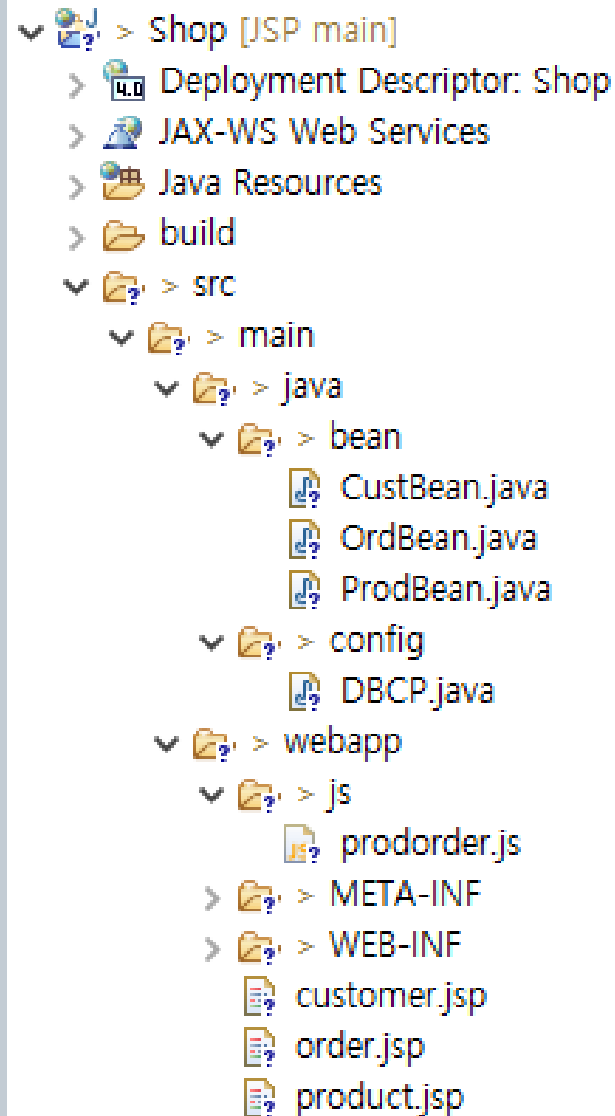
› 기능 구현

› 실행

과정 수행

1. 프로젝트 생성 및 구성

- 프로젝트명 : Shop
- WAS : Tomcat 9
- DB : java2_shop
- 개발도구 : Eclipse,
Workbench



2. 화면 구현

- customer(고객) 목록 화면 구현
- order(주문) 목록 화면 구현
- product(상품) 목록 화면 구현

● 화면 구현

● table resource 입력

```
<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="java2_shop"
password="1234"
maxTotal="16"
maxIdle="16"
maxWaitMillis="3000"
/>
```

● DBCP

```
1 package config;
2
3 import java.sql.Connection;
4
5
6
7
8
9
10 public class DBCP {
11
12     private static DataSource ds = null;
13
14     public static Connection getConnection() throws NamingException, SQLException {
15
16         if(ds == null) {
17
18             ds = (DataSource) new InitialContext().lookup("java:comp/env/dbcp_java2_shop");
19         }
20
21         return ds.getConnection();
22     }
23 }
24
25
```

● bean

```
public class CustBean {

    private String custid;
    private String name;
    private String hp;
    private String addr;
    private String rdate;

    public String getCustid() {
        return custid;
    }
    public void setCustid(String custid) {
        this.custid = custid;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getHp() {
        return hp;
    }
    public void setHp(String hp) {
        this.hp = hp;
    }
    public String getAddr() {
        return addr;
    }
    public void setAddr(String addr) {
        this.addr = addr;
    }
    public String getRdate() {
        return rdate;
    }
    public void setRdate(String rdate) {
```

```
public class OrdBean {

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

    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;
    }
    public String getOrderDate() {
        return orderDate;
    }
    public void setOrderDate(String orderDate) {
```

```
public class ProdBean {
    private int prodNo;
    private String prodName;
    private int stock;
    private int price;
    private String company;

    public int getProdNo() {
        return prodNo;
    }
    public void setProdNo(int prodNo) {
        this.prodNo = prodNo;
    }
    public String getProdName() {
        return prodName;
    }
    public void setProdName(String prodName) {
        this.prodName = prodName;
    }
    public int getStock() {
        return stock;
    }
    public void setStock(int 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;
    }
}
```

● 화면 구현

● 고객목록 구현

```
11 List<CustBean> custs = new ArrayList<>();
12
13 try{
14     Connection conn = DBCP.getConnection();
15     Statement stmt = conn.createStatement();
16     ResultSet rs = stmt.executeQuery("select * from `customer`");
17
18     while(rs.next()){
19         CustBean cb = new CustBean();
20         cb.setCustid(rs.getString(1));
21         cb.setName(rs.getString(2));
22         cb.setHp(rs.getString(3));
23         cb.setAddr(rs.getString(4));
24         cb.setRdate(rs.getString(5));
25
26         custs.add(cb);
27     }
28
29     rs.close();
30     stmt.close();
31     conn.close();
32
33 }catch(Exception e){
34     e.printStackTrace();
35 }
36 %>
37 <!DOCTYPE html>
38 <html>
39 <head>
40     <meta charset="UTF-8">
41     <title>customer</title>
42 </head>
43 <body>
44     <h3>고객목록</h3>
45
```

```
46
47 <a href="/Shop/customer.jsp">고객목록</a>
48 <a href="/Shop/order.jsp">주문목록</a>
49 <a href="/Shop/product.jsp">상품목록</a>
50
51 <table border="1">
52     <tr>
53         <th>아이디</th>
54         <th>이름</th>
55         <th>휴대폰</th>
56         <th>주소</th>
57         <th>가입일</th>
58     </tr>
59     <% for(CustBean cb : custs){ %>
60     <tr>
61         <td><%= cb.getCustid() %></td>
62         <td><%= cb.getName() %></td>
63         <td><%= cb.getHp() %></td>
64         <td><%= cb.getAddr() %></td>
65         <td><%= cb.getRdate() %></td>
66     </tr>
67     <% } %>
68 </table>
69 </body>
</html>
```

● 주문목록 구현

```
11 List<OrdBean> ords = new ArrayList<>();
12
13 try{
14     Connection conn = DBCP.getConnection();
15     Statement stmt = conn.createStatement();
16     ResultSet rs = stmt.executeQuery("SELECT * FROM `order`");
17
18     while(rs.next()){
19         OrdBean ob = new OrdBean();
20         ob.setOrderNo(rs.getInt(1));
21         ob.setOrderId(rs.getString(2));
22         ob.setOrderProduct(rs.getString(3));
23         ob.setOrderCount(rs.getInt(4));
24         ob.setOrderDate(rs.getString(5));
25
26         ords.add(ob);
27     }
28
29     rs.close();
30     stmt.close();
31     conn.close();
32
33 }catch(Exception e){
34     e.printStackTrace();
35 }
36 %>
37 <!DOCTYPE html>
38 <html>
39 <head>
40     <meta charset="UTF-8">
41     <title>order</title>
42 </head>
43 <body>
44     <h3>주문목록</h3>
45
46     <a href="/Shop/customer.jsp">고객목록</a>
```

```
47
48 <a href="/Shop/order.jsp">주문목록</a>
49 <a href="/Shop/product.jsp">상품목록</a>
50
51 <table border="1">
52     <tr>
53         <th>주문번호</th>
54         <th>주문자</th>
55         <th>주문상품</th>
56         <th>주문수량</th>
57         <th>주문일</th>
58     </tr>
59     <% for(OrdBean ob : ords){ %>
60     <tr>
61         <td><%= ob.getOrderNo() %></td>
62         <td><%= ob.getOrderId() %></td>
63         <td><%= ob.getOrderProduct() %></td>
64         <td><%= ob.getOrderCount() %></td>
65         <td><%= ob.getOrderDate() %></td>
66     </tr>
67     <% } %>
68 </table>
69 </body>
</html>
```

● 화면 구현

● 상품목록 구현

```
12 List<ProdBean> prods = new ArrayList<>();
13
14 try{
15     Connection conn = DBCP.getConnection();
16     Statement stmt = conn.createStatement();
17     ResultSet rs = stmt.executeQuery("select * from `product`");
18
19     while(rs.next()){
20         ProdBean pb = new ProdBean();
21         pb.setProdNo(rs.getInt(1));
22         pb.setProdName(rs.getString(2));
23         pb.setStock(rs.getInt(3));
24         pb.setPrice(rs.getInt(4));
25         pb.setCompany(rs.getString(5));
26
27         prods.add(pb);
28     }
29
30     rs.close();
31     stmt.close();
32     conn.close();
33
34 }catch(Exception e){
35     e.printStackTrace();
36 }
37 %>
38 <!DOCTYPE html>
39 <html>
40 <head>
41     <meta charset="UTF-8">
42     <title>product</title>
43     <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.1/jquery.min.js"></script>
44     <script src="/Shop/js/prodorder.js"></script>
45     <script>
46     $(document).on('click', '#btnOrder', function(){
```

```
48     let prodno = $('input[name=prodno]').val();
49     let ordercount = $('input[name=ordercount]').val();
50     let orderid = $('input[name=orderid]').val();
51
52     let jsonData = {
53         "prodno":prodno,
54         "ordercount":ordercount,
55         "orderid":orderid
56     };
57
58     console.log(jsonData);
59
60     $.ajax({
61         url: './product.jsp',
62         method: 'post',
63         data: jsonData,
64         dataType: 'json',
65         success: function(data){
66
67             alert('주문완료!');
68         }
69     });
70 </script>
71 </head>
72 <body>
73     <h3>상품목록</h3>
74
75     <a href="/Shop/customer.jsp">고객목록</a>
76     <a href="/Shop/order.jsp">주문목록</a>
77     <a href="/Shop/product.jsp">상품목록</a>
78
79     <table border="1">
80     <tr>
81         <th>상품번호</th>
82         <th>상품명</th>
83
```

```
84         <th>재고량</th>
85         <th>가격</th>
86         <th>제조사</th>
87         <th>주문</th>
88     </tr>
89     <% for(ProdBean pb : prods){ %>
90     <tr>
91         <td><%= pb.getProdNo() %></td>
92         <td><%= pb.getProdName() %></td>
93         <td><%= pb.getStock() %></td>
94         <td><%= pb.getPrice() %></td>
95         <td><%= pb.getCompany() %></td>
96         <td><button type="button" id="btnOrder">주문</button></td>
97     </tr>
98     <% } %>
99 </table>
100 <h4>주문하기</h4>
101
102 <table border="1">
103     <tr>
104         <td>상품번호</td><td><input type="number" name="prodno"/></td>
105     </tr>
106     <tr>
107         <td>수량</td><td><input type="number" name="ordercount"/></td>
108     </tr>
109     <tr>
110         <td>주문자</td><td><input type="text" name="orderid"/></td>
111     </tr>
112     <tr>
113         <td colspan="2"><input type="submit" value="주문하기" id="btnOrder"/></td>
114     </tr>
115 </table>
116 </body>
117 </html>
```


3. 기능 구현

- customer(고객) 목록 화면 기능 구현
- order(주문) 목록 화면 기능 구현
- product(상품) 목록 화면 기능 구현
- product(상품) 주문하기 기능 구현
- DBCP 커넥션 사용
- 각 기능별 코드 화면 첨부 및 설명

● 기능 구현

● 주문하기 기능 (미구현)

```
5 function product(){
6
7     $(function(){
8
9         $('section').empty();
10        $('nav').empty().append("<h4>주문하기</h4>");
11
12        let table = "<table border='1'>";
13        table += "<tr>";
14        table += "<td>상품번호</td>";
15        table += "<td><input type='number' name='prodno' /></td>";
16        table += "</tr>";
17        table += "<tr>";
18        table += "<td>수량</td>";
19        table += "<td><input type='number' name='ordercount' /></td>";
20        table += "</tr>";
21        table += "<tr>";
22        table += "<td>주문자</td>";
23        table += "<td><input type='text' name='orderid' /></td>";
24        table += "</tr>";
25        table += "<tr>";
26        table += "<td colspan='2' align='right'><input type='submit' id='btnOrder' value='주문하기' /></td>";
27        table += "</tr>";
28        table += "</table>";
29
30        $('section').append(table);
31
32    });
33
34 }
```

- 기능 구현

- customer, order, product 화면 기능 구현

`고객목록`

`주문목록`

`상품목록`

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	쪄리풍	2200	0	크라운	주문
6	마가렛트	3500	3500	롯데	주문
7	뿌셔뿌셔	1650	1200	오뚜기	주문

- 주문기능

- 미구현

주문하기

상품번호	<input type="text"/>
수량	<input type="text"/>
주문자	<input type="text"/>
<input type="button" value="주문하기"/>	

html body

Console

Issues

▶

🔇

top ▼

👁

Filter

Default levels ▼

1 Issue: 1

⚙

▶ {prodno: '1', ordercount: '10',orderid: 'c103'} product.jsp:31

> |

상품번호	1
수량	10
주문자	c103
<input type="button" value="주문하기"/>	

- 미구현 기능 목록

- 상품목록에서 주문 버튼 클릭 시 하단 주문하기 화면 출력
- 상품번호 자동 입력
- 주문하기 버튼 클릭 시 Order 테이블로 컬럼 입력

4. 실행

- customer(고객) 목록 실행 화면 첨부
- order(주문) 목록 실행 화면 첨부
- product(상품) 목록 실행 화면 첨부
- product(상품) 주문 실행 화면 첨부

고객목록

[고객목록](#) [주문목록](#) [상품목록](#)

아이디	이름	휴대폰	주소	가입일
c101	김유신	010-1234-1001	김해시 봉황동	2022-01-01
c102	김준추	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
c105	이성계	null	null	2022-01-05
c106	정철	010-1234-1006	경기도 용인시	2022-01-06
c107	허준	null	null	2022-01-07
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	c102	3	2	2022-08-29 19:06:53
2	c101	4	1	2022-08-29 19:06:53
3	c108	1	1	2022-08-29 19:06:53
4	c109	6	5	2022-08-29 19:06:53
5	c102	2	1	2022-08-29 19:06:53
6	c101	7	3	2022-08-29 19:06:53
7	c110	1	2	2022-08-29 19:06:53
8	c104	2	4	2022-08-29 19:06:53
9	c102	1	3	2022-08-29 19:06:53
10	c107	6	1	2022-08-29 19:06:53

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	쪄리풍	2200	0	크라운	주문
6	마가렛트	3500	3500	롯데	주문
7	뿌셔뿌셔	1650	1200	오뚜기	주문

주문하기

상품번호	<input type="text"/>
수량	<input type="text"/>
주문자	<input type="text"/>
<input type="button" value="주문하기"/>	