



#2022.10.21 최아영

수행평가



SQL 활용



목차

UI 설계 확인하기
UI 구현하기

#1, 프로젝트 생성 및 구성

#2, 화면 구현

#3, 기능 구현

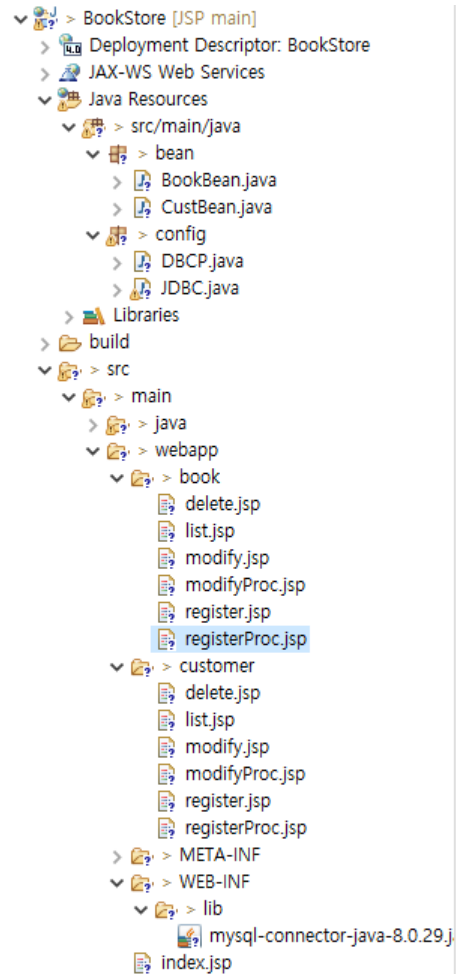
#4, 실행



Part 1,

프로젝트 생성 및 구성





프로젝트 구조

프로젝트명 : BookStore

WAS : Tomcat 9

DB : java1_bookstore

개발도구 : Eclipse, Workbench



Part 2,

화면 구현

화면 구현

BookStore

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champion	Pearson	13000	수정 삭제

도서수정

처음으로 도서목록

도서번호	10
도서명	Olympic Champion
출판사	Pearson
가격	13000
<input type="button" value="수정"/>	

도서목록 고객목록

고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	null	수정 삭제

고객수정

처음으로 고객목록

고객번호	1
고객명	박지성
주소	영국 맨체스타
휴대폰	000-5000-0001
<input type="button" value="수정"/>	

Part 3,

기능 구현

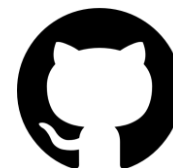


Part 3 기능 구현

[index]

```
<body>
  <h3>BookStore</h3>
  <a href="/book/list.jsp">도서목록</a>
  <a href="/customer/list.jsp">고객목록</a>
</body>
```

<코드 GitHub참조>



click

[SQL연결]

```
▼ WEB-INF
  ▼ lib
    mysql-connector-java-8.0.29.jar
  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_java1_bookstore");
      }
      return ds.getConnection();
    }
  }

  <!-- 커넥션 풀 설정 -->
  <Resource
    name="dbcp_java1_bookstore"
    auth="Container"
    type="javax.sql.DataSource"
    driverClassName="com.mysql.cj.jdbc.Driver"
    url="jdbc:mysql://127.0.0.1:3306/java1_bookstore"
    username="root"
    password="1234"
    maxTotal="8"
    maxIdle="8"
    maxWaitMillis="3000"
  />
```

```
12 List<BookBean> books = null;
13
14 try{
15     Connection conn = DBCP.getConnection();
16
17     Statement stmt = conn.createStatement();
18     ResultSet rs = stmt.executeQuery("SELECT * FROM `book`");
19
20     books = new ArrayList<>();
21
22     while(rs.next()){
23         BookBean ub = new BookBean();
24         ub.setBookId(rs.getInt(1));
25         ub.setBookName(rs.getString(2));
26         ub.setPublisher(rs.getString(3));
27         ub.setPrice(rs.getString(4));
28
29         books.add(ub);
30     }
31
32     rs.close();
33     stmt.close();
34     conn.close();
35
36 }catch(Exception e){
37     e.printStackTrace();
38 }
39
40 %>
```

```
41 <!DOCTYPE html>
42<html>
43<head>
44     <meta charset="UTF-8">
45     <title>book::list</title>
46 </head>
47<body>
48     <h3>도서목록</h3>
49
50     <a href="../index.jsp">처음오르</a>
51     <a href="../register.jsp">도서등록</a>
52
53     <table border="1">
54         <tr>
55             <th>도서번호</th>
56             <th>도서명</th>
57             <th>출판사</th>
58             <th>가격</th>
59             <th>관리</th>
60         </tr>
61
62         <% for(BookBean ub : books){ %>
63             <tr>
64                 <td><%= ub.getBookId() %></td>
65                 <td><%= ub.getBookName() %></td>
66                 <td><%= ub.getPublisher() %></td>
67                 <td><%= ub.getPrice() %></td>
68                 <td>
69                     <a href="../modify.jsp?bookId=<%= ub.getBookId() %>">수정</a>
70                     <a href="../delete.jsp?bookId=<%= ub.getBookId() %>">삭제</a>
71                 </td>
72             </tr>
73             <% } %>
74         </table>
75
76     </body>
77 </html>
```

Part 3 기능 구현 [book-modify]

```
7  <%
8  request.setCharacterEncoding("utf-8");
9  String bookId = request.getParameter("bookId");
10
11  BookBean bb = null;
12
13  try{
14      Connection conn = DBCP.getConnection();
15      String sql = "select * from `book` where `bookId`=?";
16      PreparedStatement psmt = conn.prepareStatement(sql);
17      psmt.setString(1, bookId);
18
19      ResultSet rs = psmt.executeQuery();
20
21      if(rs.next()){
22          bb = new BookBean();
23          bb.setBookId(rs.getInt(1));
24          bb.setBookName(rs.getString(2));
25          bb.setPublisher(rs.getString(3));
26          bb.setPrice(rs.getString(4));
27      }
28
29      rs.close();
30      psmt.close();
31      conn.close();
32
33  }catch(Exception e){
34      e.printStackTrace();
35  }
36  }
37  %>
```

```
38  <!DOCTYPE html>
39  <html>
40  <head>
41      <meta charset="UTF-8">
42      <title>book::modify</title>
43  </head>
44  <body>
45      <h3>도서수정</h3>
46
47      <a href=" ../index.jsp">처음으로</a>
48      <a href=" ../List.jsp">도서목록</a>
49
50      <form action=" ../modifyProc.jsp" method="post">
51          <table border="1">
52              <tr>
53                  <td>도서번호</td>
54                  <td><input type="number" name="bookId" value="<%= bb.getBookId() %%" readonly="readonly"/></td>
55              </tr>
56              <tr>
57                  <td>도서명</td>
58                  <td><input type="text" name="bookName" value="<%= bb.getBookName() %%" /></td>
59              </tr>
60              <tr>
61                  <td>출판사</td>
62                  <td><input type="text" name="publisher" value="<%= bb.getPublisher() %%" /></td>
63              </tr>
64              <tr>
65                  <td>가격</td>
66                  <td><input type="text" name="price" value="<%= bb.getPrice() %%" /></td>
67              </tr>
68              <tr>
69                  <td colspan="2" align="right"><input type="submit" value="수정"/></td>
70              </tr>
71          </table>
72      </form>
73  </body>
74  </html>
```

```
5  <%
6  //전송 데이터 수신
7  request.setCharacterEncoding("utf-8");
8  String bookId = request.getParameter("bookId");
9  String bookName = request.getParameter("bookName");
10 String publisher = request.getParameter("publisher");
11 String price = request.getParameter("price");
12
13 try{
14     Connection conn = DBCP.getConnection();
15
16     String sql = "update `book` set `bookName`=?, `publisher`=?, `price`=? ";
17     sql += "where `bookId`=?";
18
19     PreparedStatement psmt = conn.prepareStatement(sql);
20     psmt.setString(1, bookName);
21     psmt.setString(2, publisher);
22     psmt.setString(3, price);
23     psmt.setString(4, bookId);
24
25     psmt.executeUpdate();
26
27     psmt.close();
28     conn.close();
29
30 }catch(Exception e){
31     e.printStackTrace();
32 }
33
34 response.sendRedirect(" ../list.jsp");
35
36
37 %>
```

```

2 <!DOCTYPE html>
3 <html>
4   <head>
5     <meta charset="UTF-8">
6     <title>book::register</title>
7   </head>
8   <body>
9     <h3>도서등록</h3>
10
11     <a href="../index.jsp">처음으로</a>
12     <a href="../list.jsp">도서목록</a>
13
14     <form action="../registerProc.jsp" method="post">
15       <table border="1">
16         <tr>
17           <td>도서번호</td>
18           <td><input type="number" name="bookId" placeholder="도서번호 입력"/></td>
19         </tr>
20         <tr>
21           <td>도서명</td>
22           <td><input type="text" name="bookName" placeholder="도서명 입력"/></td>
23         </tr>
24         <tr>
25           <td>출판사</td>
26           <td><input type="text" name="publisher" placeholder="출판사 입력"/></td>
27         </tr>
28         <tr>
29           <td>가격</td>
30           <td><input type="text" name="price" placeholder="가격 입력"/></td>
31         </tr>
32         <tr>
33           <td colspan="2" align="right"><input type="submit" value="등록"/></td>
34         </tr>
35       </table>
36     </form>
37   </body>
38 </html>

```

```

//전송 데이터 수신
request.setCharacterEncoding("utf-8");
String bookId = request.getParameter("bookId");
String bookName = request.getParameter("bookName");
String publisher = request.getParameter("publisher");
String price = request.getParameter("price");

// 데이터베이스 작업
try{
  // 1단계 - JNDI 서비스 객체생성
  Context initCtx = new InitialContext();
  Context ctx = (Context)initCtx.lookup("java:comp/env"); // JNDI 서비스를 사용하기 위한 기본 이름

  // 2단계 - 커넥션 풀에서 커넥션 가져오기
  DataSource ds = (DataSource) ctx.lookup("dbcp_java1_bookstore"); // 커넥션 풀 가져오기
  Connection conn = ds.getConnection(); // 커넥션 가져오기

  // 3단계
  String sql = "INSERT INTO `book` VALUES (?, ?, ?, ?)";
  PreparedStatement psmt = conn.prepareStatement(sql);
  psmt.setString(1, bookId);
  psmt.setString(2, bookName);
  psmt.setString(3, publisher);
  psmt.setString(4, price);

  // 4단계
  psmt.executeUpdate();
  // 5단계
  // 6단계
  psmt.close();
  conn.close(); // 반납

}catch(Exception e){
  e.printStackTrace();
}

// 리다이렉트
response.sendRedirect("../list.jsp");

```



```
//전송 데이터 수신
request.setCharacterEncoding("utf-8");
String bookId    = request.getParameter("bookId");
String bookName  = request.getParameter("bookName");
String publisher = request.getParameter("publisher");
String price     = request.getParameter("price");

// 데이터베이스 작업
try{
    // 1단계 - JNDI 서비스 객체생성
    Context initCtx = new InitialContext();
    Context ctx = (Context)initCtx.lookup("java:comp/env"); // JNDI 서비스를 사용하기 위한 기본 이름

    // 2단계 - 커넥션 풀에서 커넥션 가져오기
    DataSource ds = (DataSource) ctx.lookup("dbcp_java1_bookstore"); // 커넥션 풀 가져오기
    Connection conn = ds.getConnection(); // 커넥션 가져오기

    // 3단계
    String sql = "DELETE FROM `book` WHERE `bookId`=?";
    PreparedStatement psmt = conn.prepareStatement(sql);
    psmt.setString(1, bookId);

    // 4단계
    psmt.executeUpdate();
    // 5단계
    // 6단계
    psmt.close();
    conn.close(); // 반납

}catch(Exception e){
    e.printStackTrace();
}

// 리다이렉트
response.sendRedirect("./list.jsp");
```

```

List<CustBean> custs = null;

try{
    Connection conn = DBCP.getConnection();

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

    custs = new ArrayList<>();

    while(rs.next()){
        CustBean cb = new CustBean();
        cb.setCustId(rs.getInt(1));
        cb.setName(rs.getString(2));
        cb.setAddress(rs.getString(3));
        cb.setPhone(rs.getString(4));

        custs.add(cb);
    }

    rs.close();
    stmt.close();
    conn.close();
}catch(Exception e){
    e.printStackTrace();
}

```

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>customer::list</title>
</head>
<body>
    <h3>고객목록</h3>

    <a href="../index.jsp">처음으로</a>
    <a href="../register.jsp">고객등록</a>

    <table border="1">
        <tr>
            <th>고객번호</th>
            <th>고객명</th>
            <th>주소</th>
            <th>휴대폰</th>
            <th>관리</th>
        </tr>

        <% for(CustBean cb : custs){ %>
        <tr>
            <td><%= cb.getCustId() %></td>
            <td><%= cb.getName() %></td>
            <td><%= cb.getAddress() %></td>
            <td><%= cb.getPhone() %></td>
            <td>
                <a href="../modify.jsp?custId=<%= cb.getCustId() %>">수정</a>
                <a href="../delete.jsp?custId=<%= cb.getCustId() %>">삭제</a>
            </td>
        </tr>
        <% } %>
    </table>

</body>
</html>

```

Part 3 기능 구현 [customer-modify]

```
<%
request.setCharacterEncoding("utf-8");
String custId = request.getParameter("custId");

CustBean cb = null;

try{
    Connection conn = DBCP.getConnection();
    String sql = "select * from `customer` where `custId`=?";
    PreparedStatement psmt = conn.prepareStatement(sql);
    psmt.setString(1, custId);

    ResultSet rs = psmt.executeQuery();

    if(rs.next()){
        cb = new CustBean();
        cb.setCustId(rs.getInt(1));
        cb.setName(rs.getString(2));
        cb.setAddress(rs.getString(3));
        cb.setPhone(rs.getString(4));
    }

    rs.close();
    psmt.close();
    conn.close();
}catch(Exception e){
    e.printStackTrace();
}
%>
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>customer::modify</title>
  </head>
  <body>
    <h3>고객수정</h3>

    <a href=".." />뒤로</a>
    <a href=".." />고객목록</a>

    <form action=".." method="post">
      <table border="1">
        <tr>
          <td>고객번호</td>
          <td><input type="text" name="custId" value="<%= cb.getCustId() %>" readonly="readonly" /></td>
        </tr>
        <tr>
          <td>고객명</td>
          <td><input type="text" name="name" value="<%= cb.getName() %>" /></td>
        </tr>
        <tr>
          <td>주소</td>
          <td><input type="text" name="address" value="<%= cb.getAddress() %>" /></td>
        </tr>
        <tr>
          <td>휴대폰</td>
          <td><input type="text" name="phone" value="<%= cb.getPhone() %>" /></td>
        </tr>
        <tr>
          <td colspan="2" align="right"><input type="submit" value="수정" /></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

```
<%
//전송 데이터 수신
request.setCharacterEncoding("utf-8");
String custId = request.getParameter("custId");
String name = request.getParameter("name");
String address = request.getParameter("address");
String phone = request.getParameter("phone");

try{
    Connection conn = DBCP.getConnection();

    String sql = "update `customer` set `name`=?, `address`=?, `phone`=? ";
    sql += "where `custId`=?";

    PreparedStatement psmt = conn.prepareStatement(sql);
    psmt.setString(1, name);
    psmt.setString(2, address);
    psmt.setString(3, phone);
    psmt.setString(4, custId);

    psmt.executeUpdate();

    psmt.close();
    conn.close();

}catch(Exception e){
    e.printStackTrace();
}

response.sendRedirect("../list.jsp");
%>
```

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>book::register</title>
  </head>
  <body>
    <h3>고객등록</h3>

    <a href="../index.jsp">처음으로</a>
    <a href="../list.jsp">고객목록</a>

    <form action="../registerProc.jsp" method="post">
      <table border="1">
        <tr>
          <td>고객번호</td>
          <td><input type="number" name="custId" placeholder="고객번호 입력"/></td>
        </tr>
        <tr>
          <td>고객명</td>
          <td><input type="text" name="name" placeholder="고객명 입력"/></td>
        </tr>
        <tr>
          <td>주소</td>
          <td><input type="text" name="address" placeholder="주소 입력"/></td>
        </tr>
        <tr>
          <td>휴대폰</td>
          <td><input type="text" name="phone" placeholder="휴대폰 입력"/></td>
        </tr>
        <tr>
          <td colspan="2" align="right"><input type="submit" value="등록"/></td>
        </tr>
      </table>
    </form>
  </body>
</html>

```

```

<%
//전송 데이터 수신
request.setCharacterEncoding("utf-8");
String custId = request.getParameter("custId");
String name = request.getParameter("name");
String address = request.getParameter("address");
String phone = request.getParameter("phone");

// 데이터베이스 작업
try{
  // 1단계 - JNDI 서비스 객체생성
  Context initCtx = new InitialContext();
  Context ctx = (Context)initCtx.lookup("java:comp/env"); // JNDI 서비스를 사용하기 위한 기본 이름

  // 2단계 - 커넥션 풀에서 커넥션 가져오기
  DataSource ds = (DataSource) ctx.lookup("dbcp_java1_bookstore"); // 커넥션 풀 가져오기
  Connection conn = ds.getConnection(); // 커넥션 가져오기

  // 3단계
  String sql = "DELETE FROM `customer` WHERE `custId`=?";
  PreparedStatement psmt = conn.prepareStatement(sql);
  psmt.setString(1, custId);

  // 4단계
  psmt.executeUpdate();
  // 5단계
  // 6단계
  psmt.close();
  conn.close(); // 반납

}catch(Exception e){
  e.printStackTrace();
}

// 리다이렉트
response.sendRedirect("../list.jsp");
%>

```



```
<%  
//전송 데이터 수신  
request.setCharacterEncoding("utf-8");  
String custId = request.getParameter("custId");  
String name = request.getParameter("name");  
String address = request.getParameter("address");  
String phone = request.getParameter("phone");  
  
// 데이터베이스 작업  
try{  
    // 1단계 - JNDI 서비스 객체생성  
    Context initCtx = new InitialContext();  
    Context ctx = (Context)initCtx.lookup("java:comp/env"); // JNDI 서비스를 사용하기 위한 기본 이름  
  
    // 2단계 - 커넥션 풀에서 커넥션 가져오기  
    DataSource ds = (DataSource) ctx.lookup("dbcp_java1_bookstore"); // 커넥션 풀 가져오기  
    Connection conn = ds.getConnection(); // 커넥션 가져오기  
  
    // 3단계  
    String sql = "DELETE FROM `customer` WHERE `custId`=?";  
    PreparedStatement psmt = conn.prepareStatement(sql);  
    psmt.setString(1, custId);  
  
    // 4단계  
    psmt.executeUpdate();  
    // 5단계  
    // 6단계  
    psmt.close();  
    conn.close(); // 반납  
  
}catch(Exception e){  
    e.printStackTrace();  
}  
  
// 리다이렉트  
response.sendRedirect("../list.jsp");  
%>
```

A close-up photograph of a brown leather bag with a strap, resting on a dark blue textured surface. Inside the bag, a black smartphone, a brown notebook with a red string, a red and silver pen, and a pair of black-rimmed glasses are visible.

Part 4,

실행

Part 4 **실행** [book]

BookStore

도서목록 [고객목록](#)

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champion	Pearson	13000	수정 삭제

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠1	7100	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champion	Pearson	13000	수정 삭제

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champion	Pearson	13000	수정 삭제

도서수정

[처음으로 도서목록](#)

도서번호	1
도서명	축구의 역사
출판사	굿스포츠
가격	7000
<input type="button" value="수정"/>	

도서수정

[처음으로 도서목록](#)

도서번호	1
도서명	축구의 역사
출판사	굿스포츠1
가격	7100
<input type="button" value="수정"/>	

Part 4

실행 [customer]

BookStore

[도서목록](#) [고객목록](#)

고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	null	수정 삭제

고객수정

처음으로 고객목록

고객번호	1
고객명	박지성
주소	영국 맨체스타
휴대폰	000-5000-0001
<input type="button" value="수정"/>	

고객수정

처음으로 고객목록

고객번호	1
고객명	박지성
주소	영국 런던
휴대폰	000-5000-0002
<input type="button" value="수정"/>	

고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 런던	000-5000-0002	수정 삭제
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	null	수정 삭제

고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	null	수정 삭제

A night sky with a starry background and a blue rectangular box in the center. The box contains the Korean text '감사합니다.' in white.

감사합니다.