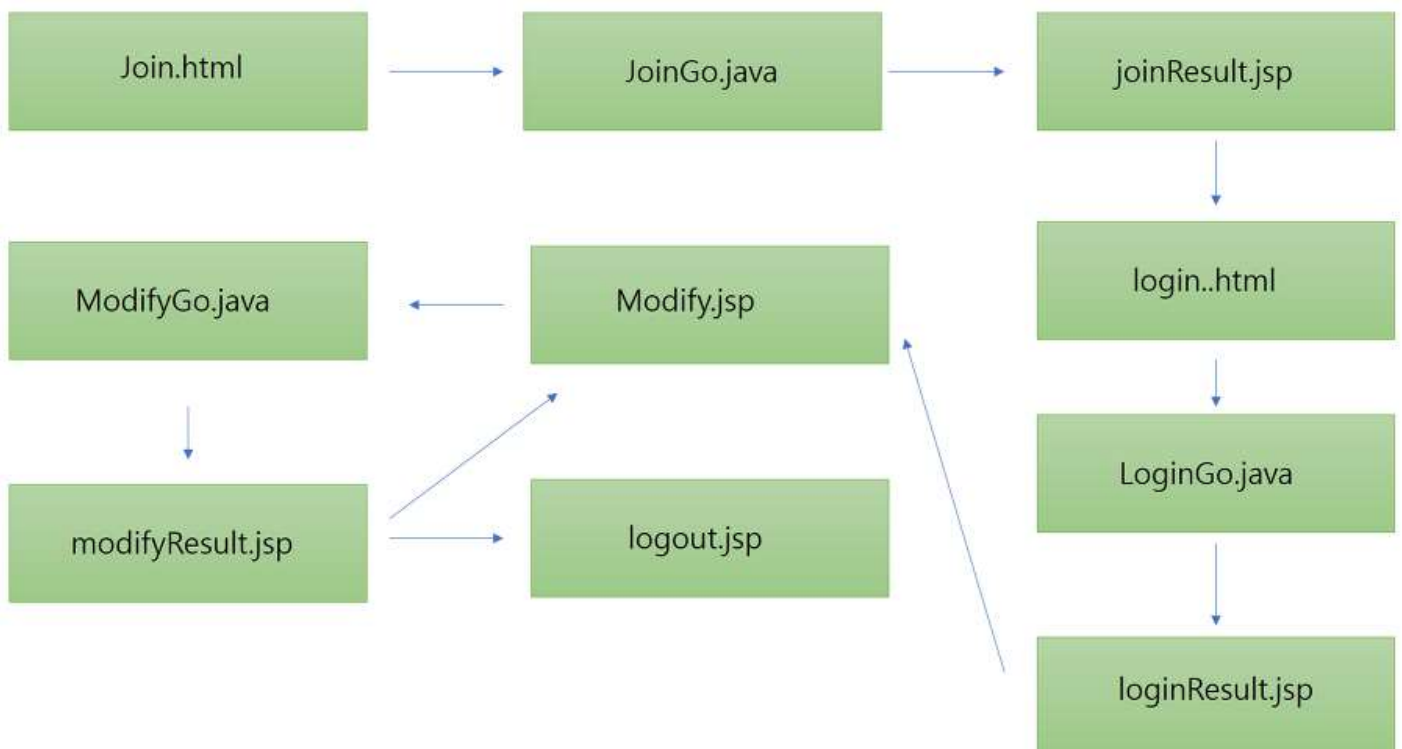


JSP Servlet 기초(12) 회원가입 및 회원정보 수정하기

JSP 2018. 1. 31. 16:41

이번엔 JSP Servlet 기초(11) 에서 했던것들을
이용해 회원가입 및 회원정보 수정하기를
만들어 보겠습니다

먼저 회원가입 및 회원정보 수정 하기가 가능한
기능을 만들기 위해서는 설계를 먼저 해야합니다



먼저 첫화면부터 값을 어디로 넘겨 또 그곳에서 데이터
처리후 어디로 보낼것인지를 설계 해야합니다.

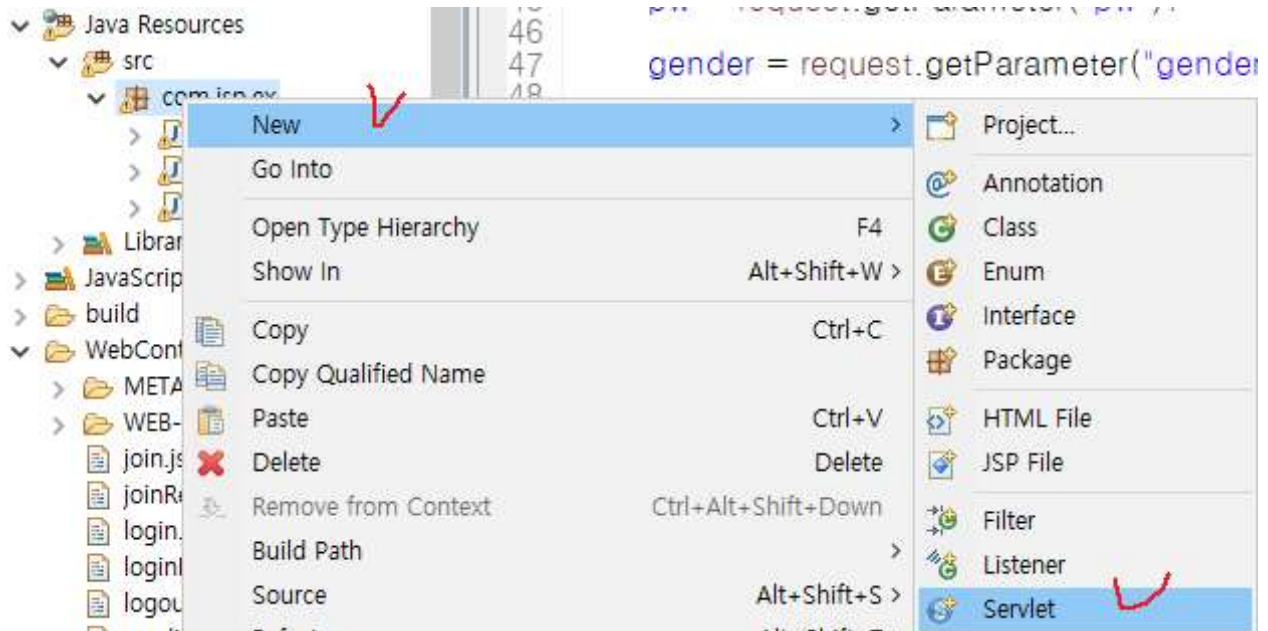
그전에 먼저 DB에 테이블먼저 생성해보겠습니다

```
create table member(  
    name varchar2(20),  
    id varchar2(20),  
    pw varchar2(20),  
    gender varchar2(20));
```

The screenshot shows a SQL IDE window with a toolbar at the top. The main area displays the SQL code for creating a table named 'member' with columns 'name', 'id', 'pw', and 'gender', all of type 'varchar2(20)'. The code is highlighted in yellow.

디벨로퍼에서 scott 계정에 들어가 위와 같이
입력해주시고 스크립트 실행 해주시면됩니다

다음 이클립스에 패키지를 만들고
그안에 서블릿파일 3개를 만들겠습니다



회원가입란에서 입력한 정보를 db에 넣어줄 파일과

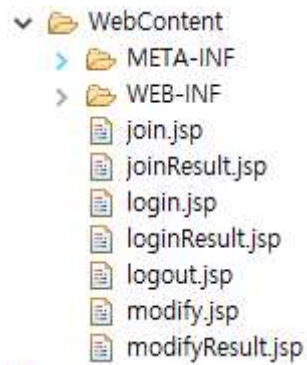
로그인 화면에서 입력한 정보를 db에서 뽑아와
세션에 넣어주는 파일,

마지막으로 회원정보를 수정하도록 도와주는
java 파일을 만들겠습니다

먼저 이름은 JoinGo.java, LoginGo.java, ModifyGo.java
로 설정하겠습니다



다음 WebContent에
7개의 jsp파일을 만들어줍니다



먼저 join 화면 먼저 보겠습니다

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10   <form action="JoinGo" method="post">
11     이름 : <input type="text" name="name" size="10"><br />
12     아이디 : <input type="text" name="id" size="10"><br />
13     비밀번호 : <input type="text" name="pw" size="10"><br />
14     성별구분 : <input type="radio" name="gender" value="man">남 <input t
15     <input type="submit" value="회원가입">
16   </form>
17 </body>
18 </html>
```

join 화면에서는 간단하게 이름, 아이디, 비밀번호, 성별을
입력할수 있도록 하겠습니다.
다음 action은 JoinGo로 하겠습니다.

이제 JoinGo.java 보겠습니다

```
1 package com.jsp.ex;
2
3 import java.io.IOException;
4 import java.io.PrintWriter;
5 import java.sql.Connection;
6 import java.sql.Driver;
7 import java.sql.DriverManager;
8 import java.sql.PreparedStatement;
9 import java.sql.ResultSet;
10 import java.sql.SQLException;
11 import java.sql.Statement;
12 import java.util.Arrays;
13
```

```

14 import javax.servlet.ServletException;
15 import javax.servlet.annotation.WebServlet;
16 import javax.servlet.http.HttpServlet;
17 import javax.servlet.http.HttpServletRequest;
18 import javax.servlet.http.HttpServletResponse;
19
20
21 @WebServlet("/JoinGo")
22 public class JoinGo extends HttpServlet {
23     private static final long serialVersionUID = 1L;
24     private Connection connection;
25     private Statement stmt;
26
27     private String name, id, pw, gender;
28
29     public JoinGo() {
30         super();
31     }
32
33     protected void doGet(HttpServletRequest request, HttpServletResponse response) {
34         actionDo(request, response);
35     }
36
37     protected void doPost(HttpServletRequest request, HttpServletResponse response) {
38         actionDo(request, response);
39     }
40
41     private void actionDo(HttpServletRequest request, HttpServletResponse response) {
42
43         request.setCharacterEncoding("UTF-8");
44
45         name = request.getParameter("name");
46         id = request.getParameter("id");
47         pw = request.getParameter("pw");
48         gender = request.getParameter("gender");
49
50         String query = "insert into member values('" + name + "', '" + id + "';
51
52         try {
53             Class.forName("oracle.jdbc.driver.OracleDriver");
54             connection = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe");
55             stmt = connection.createStatement();
56             int i = stmt.executeUpdate(query);
57             if(i == 1){
58                 System.out.println("insert success");
59                 response.sendRedirect("joinResult.jsp");
60             } else {
61                 System.out.println("insert fail");
62                 response.sendRedirect("join.jsp");
63             }
64
65         } catch (Exception e) {

```

```

67         e.printStackTrace();
68     } finally {
69         try {
70             if(stmt != null) stmt.close();
71             if(connection != null) connection.close();
72         } catch (Exception e) {}
73     }
74 }
75 }
76 }
77 }
78 }
79 }

```

먼저 24,25line에서 커넥션과 스테이트먼트 객체를 생성합니다.

다음45~48line 에서는 join.jsp 에서 입력한 데이터들을 request.getParameter 하여 요청합니다 .

50line에서는 데이터를 insert 해줄 쿼리문을 입력합니다
다음 try문안에 데이터베이스 연결을위한 로드를 합니다

56line 은 stmt.executeUpdate(query);를 사용하여 위에서 50line 에서 입력한 쿼리문을 update 해줍니다 executeUpdate 문 의 값 반환형은 int 형이고 몇개의 레코드가 수정됐는지에 따라 값이 반환됩니다
1개의 레코드가 수정되었기 때문에 1이 반환 될것입니다.
1이 반환된것은 정상적으로 update 된것이기 때문에 if 조건문을 걸어 다음화면으로 sendRedirect 시켜줍니다

다음 JoinResult.jsp 화면입니다

```

1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2      pageEncoding="UTF-8"%>
3  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
4  <html>
5  <head>
6  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7  <title>Insert title here</title>
8  </head>
9  <body>
10
11  회원가입이 완료 되었습니다.<br />
12  <a href="login.jsp">로그인</a>
13
14  </body>

```

15 </html>

회원가입이 완료되었다는 메시지와
로그인 화면으로 넘어갈 수 있도록
하이퍼링크 달아줍니다

로그인 화면 보겠습니다

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10
11 <form action="LoginGo" method="post">
12   아이디 : <input type="text" name="id"><br />
13   비밀번호 : <input type="text" name="pw"><br />
14   <input type="submit" value="로그인">
15 </form>
16
17 </body>
18 </html>
```

로그인 화면에서는 회원가입한 아이디와 비밀번호를
입력할 수 있도록 form 태그를 달아주고
LoginGo로 action 해줍니다

```
1 package com.jsp.ex;
2
3 import java.io.IOException;
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.ResultSet;
7 import java.sql.Statement;
8 import java.util.Arrays;
9
10 import javax.servlet.ServletException;
11 import javax.servlet.annotation.WebServlet;
12 import javax.servlet.http.HttpServlet;
13 import javax.servlet.http.HttpServletRequest;
14 import javax.servlet.http.HttpServletResponse;
15 import javax.servlet.http.HttpSession;
16
17 @WebServlet("/LoginGo")
18 public class LoginGo extends HttpServlet {
19   private static final long serialVersionUID = 1L;
```

```

20
21     private Connection connection;
22     private Statement stmt;
23     private ResultSet resultSet;
24
25     private String name, id, pw, gender;
26
27     public LoginGo() {
28         super();
29     }
30
31     protected void doGet(HttpServletRequest request, HttpServletResponse response) {
32         actionDo(request, response);
33     }
34
35     protected void doPost(HttpServletRequest request, HttpServletResponse response) {
36         actionDo(request, response);
37     }
38
39     private void actionDo(HttpServletRequest request, HttpServletResponse response) {
40
41         request.setCharacterEncoding("UTF-8");
42
43         id = request.getParameter("id");
44         pw = request.getParameter("pw");
45
46         String query = "select * from member where id = '" + id + "' and pw = '" + pw + "'";
47
48         try {
49             Class.forName("oracle.jdbc.driver.OracleDriver");
50             connection = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe");
51             stmt = connection.createStatement();
52             resultSet = stmt.executeQuery(query);
53
54             while (resultSet.next()) {
55                 name = resultSet.getString("name");
56                 id = resultSet.getString("id");
57                 pw = resultSet.getString("pw");
58                 gender = resultSet.getString("gender");
59             }
60
61             HttpSession httpSession = request.getSession();
62             httpSession.setAttribute("name", name);
63             httpSession.setAttribute("id", id);
64             httpSession.setAttribute("pw", pw);
65
66             response.sendRedirect("loginResult.jsp");
67
68         } catch (Exception e) {
69             e.printStackTrace();
70         } finally {
71             try {
72                 if(resultSet != null) resultSet.close();

```

```

73         if(stmt != null) stmt.close();
74         if(connection != null) connection.close();
75     } catch (Exception e2) {
76         e2.printStackTrace();
77     }
78 }
79
80 }
81
82 }
83

```

LoginGo에서는
login.jsp 에서 입력한 아이디와 비밀번호를
request.getParameter 하여 값을 요청합니다

46line 에서는 db에 login.jsp 에서 가져온
아이디와 비밀번호 와 일치하는 데이터를 db에서
뽑아오기 위해 쿼리문을 작성합니다
데이터를 검색하기위해 select 문을 사용하고
where 조건문에 login.jsp 에서 가져온
아이디와 비밀번호를 사용합니다
즉 login.jsp 에서 가져온 아이디와
db에 저장된 아이디 그리고 login.jsp 에서 가져온
비밀번호와 db에 저장되어있는 비밀번호가 일치하는
데이터만 select 하는겁니다

다음 try문에서 데이터베이스 로드를하고
(52line db에서 뽑혀온 데이터들은resultSet에 담겨집니다)
resultSet에저장되어 있는
name, id, pw, gender 다시 저장해줍니다(55~58line)

61line에서는 세션을 생성한다음
세션에 이름, 아이디, 비밀번호 값을 담습니다
다음 loginResult 화면으로 리다이렉트 시켜줍니다

```

1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2      pageEncoding="UTF-8"%>
3
4      <%!
5          String name, id, pw;
6      %>
7
8  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
9  <html>
10 <head>
11 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
12 <title>Insert title here</title>
13 </head>
14 <body>
15

```



```

16     <%
17         name = (String)session.getAttribute("name");
18         id = (String)session.getAttribute("id");
19         pw = (String)session.getAttribute("pw");
20     %>
21
22     <%= name%>님 안녕하세요 <br />
23     <a href="modify.jsp">회원정보 수정</a>
24
25 </body>
26 </html>

```

loginResult 에서는 세션에 저장되어있는 이름, 아이디, 비밀번호를 name, id, pw 에 저장합니다
(String) 은 session.getAttribute로 값을 가져오면 Object 형으로 반환되기 때문에 사용시 실제 할당된 객체 타입으로 형변화(casting) 해줍니다.

다음 로그인이 정상적으로 되었다는 메시지 출력해주고 회원정보 수정으로 이동 할 수 있도록 해줍니다.

```

1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2      pageEncoding="UTF-8"%>
3
4  <%@page import="java.sql.DriverManager"%>
5  <%@page import="java.sql.ResultSet"%>
6  <%@page import="java.sql.Statement"%>
7  <%@page import="java.sql.Connection"%>
8
9      <%!
10          Connection connection;
11          Statement statement;
12          ResultSet resultSet;
13
14          String name, id, pw, gender;
15      %>
16
17  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
18  <html>
19  <head>
20  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
21  <title>Insert title here</title>
22  </head>
23  <body>
24
25      <%
26          id = (String)session.getAttribute("id");
27
28          String query = "select * from member where id = '" + id + "'";
29

```

```

30     Class.forName("oracle.jdbc.driver.OracleDriver");
31     connection = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe");
32     statement = connection.createStatement();
33     resultSet = statement.executeQuery(query);
34
35     while(resultSet.next()) {
36         name = resultSet.getString("name");
37         pw = resultSet.getString("pw");
38         gender = resultSet.getString("gender");
39     }
40
41     %>
42
43     <form action="ModifyGo" method="post">
44         이름 : <input type="text" name="name" size="10" value=<%=name%>><br />
45         아이디 : <%=id%><br />
46         비밀번호 : <input type="text" name="pw" size="10"><br />
47
48         <%
49             if(gender.equals("man")) {
50 %>
51         성별구분 : <input type="radio" name="gender" value="man" checked="checked" />
52         <%
53             } else {
54 %>
55         성별구분 : <input type="radio" name="gender" value="man" /> 남 <input type="radio" name="gender" value="woman" /> 여
56         <%
57             }
58 %>
59         <input type="submit" value="정보수정">
60     </form>
61
62
63 </body>
64 </html>

```

회원정보 수정화면에서는
 회원정보수정을 할수있도록 form 태그를 달아줍니다
 이때 아이디는 변경할수 없도록 세션에 저장되어
 있는 id값으로 지정해줍니다<%=id%>
 26line (String) 은 session.getAttribute로
 값을 가져오면 Object 형으로 반환되기 때문에
 사용시 실제 할당된 객체 타입으로 형변화(casting)
 해준겁니다.

28line 에서 세션에서 가져온 id 값과 일치하는
 db정보를 가져오도록 쿼리문 작성합니다
 그 값들을 35line 에서 다시 저장해줍니다
 49line 에서는 db에서 뽑아온 성별을 man이라면
 성별구분 form에 남자 를 미리 체크되어 있게끔
 해준겁니다 else 문은 여자가 미리 체크되어
 있게 만들어졌습니다

다음 submit 으로 **ModifyGo** 로 action 해줍니다.

```
1 package com.jsp.ex;
2
3 import java.io.IOException;
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.Statement;
7
8 import javax.servlet.ServletException;
9 import javax.servlet.annotation.WebServlet;
10 import javax.servlet.http.HttpServlet;
11 import javax.servlet.http.HttpServletRequest;
12 import javax.servlet.http.HttpServletResponse;
13 import javax.servlet.http.HttpSession;
14
15 @WebServlet("/ModifyGo")
16 public class ModifyGo extends HttpServlet {
17     private static final long serialVersionUID = 1L;
18
19     private Connection connection;
20     private Statement stmt;
21
22     private String name, id, pw, gender;
23
24     HttpSession httpSession;
25
26     public ModifyGo() {
27         super();
28     }
29
30     protected void doGet(HttpServletRequest request, HttpServletResponse response) {
31         actionDo(request, response);
32     }
33
34     protected void doPost(HttpServletRequest request, HttpServletResponse response) {
35         actionDo(request, response);
36     }
37
38     private void actionDo(HttpServletRequest request, HttpServletResponse response) {
39
40         request.setCharacterEncoding("UTF-8");
41         httpSession = request.getSession();
42
43         name = request.getParameter("name");
44         id = request.getParameter("id");
45         pw = request.getParameter("pw");
46
47         gender = request.getParameter("gender");
48
49         if(pwConfirm()){
50             System.out.println("OK");
51         }
52     }
53 }
```

```

52         String query = "update member set name='" + name + "', gender =
53         try {
54             Class.forName("oracle.jdbc.driver.OracleDriver");
55             connection = DriverManager.getConnection("jdbc:oracle:thin:@
56             stmt = connection.createStatement();
57             int i = stmt.executeUpdate(query);
58             if(i ==1 ){
59                 System.out.println("update success");
60                 HttpSession.setAttribute("name", name);
61                 response.sendRedirect("modifyResult.jsp");
62             } else {
63                 System.out.println("update fail");
64                 response.sendRedirect("modify.jsp");
65             }
66         } catch(Exception e) {
67             e.printStackTrace();
68         } finally {
69             try {
70                 if(stmt != null) stmt.close();
71                 if(connection != null) connection.close();
72             } catch (Exception e) {}
73         }
74     } else {
75         System.out.println("NG");
76     }
77 }
78
79 }
80
81 private boolean pwConfirm() {
82     boolean rs = false;
83
84     String sessionPw = (String)HttpSession.getAttribute("pw");
85
86     if(sessionPw.equals(pw)) {
87         rs = true;
88     } else {
89         rs = false;
90     }
91
92     return rs;
93 }
94
95 }
96

```

ModifyGo에서는 회원정보 수정 화면에서
 입력한 회원정보를 db에 update 시켜줍니다
 먼저 49line 에서 if 조건문에 pwConfirm 메소드를 넣어줍니다
 (자세한건 밑에서 설명하겠습니다)
 다음 52line 에서 update 쿼리문을 작성해줍니다

회원정보 수정란에서 이름과 성별만 수정할수 있게
만들었기때문에 이름과 성별만 update 해줍니다

57line에서 executeUpdate 를 사용하여 쿼리문을
update시켜주고 1개의 레코드를 update 해줬기
때문에 반환값 1을 i 에 저장해줍니다.
update 가 정상적으로 되었다면 i의 값은 1이므로
58line에 조건을 i==1 이라는 조건을
걸어주고 확인 메세지 출력 시켜줍니다
다음 세션에 name 값을 저장하고 modifyResult
화면으로 리다이렉트 시켜줍니다
그것이 아니라면 업데이트 실패이기 때문에
else문에 update fail 메세지 출력합니다

다음 81line 에 pwConfirm 메소드는
회원정보 수정란에서 입력한 비밀번호가 세션에
저장되어있는 비밀번호와 일치한지 비교 하는
메소드입니다 pwConfirm() 메소드는
boolean 으로 선언되었기 때문에
true, false 값만 반환됩니다.그 반환값을 rs에
저장하고 rs값을 return 시켜줍니다.
true 가 반환된다면 49라인데 if조건문이
if(true) 가 되어 정상적으로 조건문안의
로직들을 수행하게 됩니다.

마지막으로 회원정보수정 결과하면입니다

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10
11 <%= session.getAttribute("name")%>님의 회원정보 수정이 정상 처리 되었습니다.<br
12 <a href="logout.jsp">로그아웃</a>      <a href="modify.jsp">정보수정</a>
13
14 </body>
15 </html>
```

세션에 저장되어있는 이름값을 이용해
수정이 정상처리 되었다는 메시지를
출력시켜주고 로그아웃 화면으로 이동할수
있도록 해줍니다

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.v
```

```
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10
11     <%
12         session.invalidate();
13         response.sendRedirect("login.jsp");
14     %>
15
16 </body>
17 </html>
```

로그아웃 화면에서는
session.invalidate(); 사용하여
세션에 저장된 모든 데이터를 삭제합니다
다음 로그인 화면으로
sendRedirect 시켜줍니다