

# **JSP Board**

## **JSP Board CRUD**

# servlet control

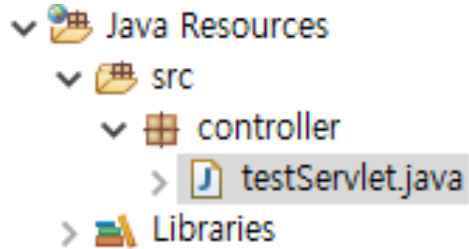
## Servlet Control

jsp를 해석하여 실행하는  
서블릿을 이용하여  
데이터를 원하는 대로 변경이 가능하다

get과 post에 따라  
같은 도메인에서 별도의 동작을  
수행할 수 있다.

**도메인을 원하는 방식으로 변경하여 사용이 가능**

# 서블릿 제작



```
import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
*testServlet.java
1 package controller;
2
3 import java.io.IOException;
4
11
12 @WebServlet("/testServlet")
13 public class testServlet extends HttpServlet {
14     private static final long serialVersionUID = 1L;
15
16     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
17         System.out.println("GET방식으로 서블릿이 실행되었습니다.");
18         RequestDispatcher dis=request.getRequestDispatcher("/get.jsp");
19         request.setAttribute("data", "get방식의 데이터");
20         dis.forward(request, response);
21     }
22
23     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
24         System.out.println("POST방식으로 서블릿이 실행되었습니다.");
25         RequestDispatcher dis=request.getRequestDispatcher("/post.jsp");
26         request.setAttribute("data", "post방식의 데이터");
27         dis.forward(request, response);
28     }
29 }
```

서블릿은 servlet으로 제작한다.

# index

index.jsp

```
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>테스트 페이지 입니다.</title>
8   </head>
9   <body>
10    <a href="testServlet">get으로 서블렛 호출</a>
11    <form method="post" action="testServlet">
12      <input type="submit" value="post로 서블렛 호출">
13    </form>
14  </body>
15 </html>
```

← → ↻ ⓘ 127.0.0.1:8080/servletTest/

[get으로 서블렛 호출](#)

post로 서블렛 호출

**get과 post를 같은 경로로 동작시킨다.**

# get.jsp

```
get.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>get으로 호출한 경우</title>
8   </head>
9   <body>
10    <h1>get방식으로 호출된 경우 이 페이지가 실행됩니다.</h1>
11    <h1>현재 파일은 get.jsp입니다.</h1>
12    ${requestScope.data }
13  </body>
14 </html>
```

← → ↻ ⓘ 127.0.0.1:8080/servletTest/testServlet

get방식으로 호출된 경우 이 페이지가 실행됩니다.

현재 파일은 get.jsp입니다.

get방식의 데이터

**a태그 또는 도메인으로 직접 접근시 get.jsp실행**

# post.jsp

```
post.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>post로 호출한 경우</title>
8   </head>
9   <body>
10    <h1>post방식으로 호출된 경우 이 페이지가 실행됩니다.</h1>
11    <h1>현재 파일은 post.jsp입니다.</h1>
12    ${requestScope.data }
13  </body>
14 </html>
```

← → ↻ ⓘ 127.0.0.1:8080/servletTest/testServlet

**post방식으로 호출된 경우 이 페이지가 실행됩니다.**

**현재 파일은 post.jsp입니다.**

post방식의 데이터

**form의 submit은 post로 본다.**

# 도메인 은닉효과

```
sample.java
1 package controller;
2
3 import java.io.IOException;
4
5 import javax.servlet.RequestDispatcher;
6 import javax.servlet.ServletException;
7 import javax.servlet.annotation.WebServlet;
8 import javax.servlet.http.HttpServlet;
9 import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11
12 @WebServlet("/a.do")
13 public class sample extends HttpServlet {
14     private static final long serialVersionUID = 1L;
15     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
16         RequestDispatcher dis=request.getRequestDispatcher("/test.jsp");
17         dis.forward(request, response);
18     }
19     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {}
20 }
```

```
test.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5     <head>
6         <meta charset="utf-8">
7         <title>테스트 페이지 입니다.</title>
8     </head>
9     <body>
10         <h1>이곳은 test.jsp입니다.</h1>
11         <h1>/test.jsp와 /a.do 모두 이곳으로 이동할 수 있습니다.</h1>
12     </body>
13 </html>
```

← → ↺ 127.0.0.1:8080/servletTest/test.jsp

이곳은 test.jsp입니다.

/test.jsp와 /a.do 모두 이곳으로 이동할 수 있습니다.

← → ↺ 127.0.0.1:8080/servletTest/a.do

이곳은 test.jsp입니다.

/test.jsp와 /a.do 모두 이곳으로 이동할 수 있습니다.

도메인의 은닉과 더불어 같은 도메인으로  
다른 동작을 수행할 수 있다는 장점이 있다.

# 게시판

## Board CRUD

게시판은  
글 목록과 상세내용을 연람하고(select)  
글을 올리고(insert)  
글을 수정하고(update)  
글을 삭제하는(delete)  
기능이 모두 들어가 있는 좋은 연습대상이다.

**데이터를 다양하게 다루어 볼 수 있다.**



# 메인페이지

index.jsp

```
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>메인페이지</title>
8   </head>
9   <body>
10    <h1>여기는 메인페이지 입니다.</h1>
11    <a href="member_list.jsp">전체멤버 조회</a>
12    <a href="register.jsp">회원가입</a>
13  </body>
14 </html>
```

여기는 메인페이지 입니다.

[전체멤버 조회](#) [회원가입](#)

로그인을 구현하기 위해서는 회원가입이 필요하다.

# 데이터베이스 준비

```
mysql> desc member;
```

| Field  | Type | Null | Key | Default | Extra          |
|--------|------|------|-----|---------|----------------|
| num    | int  | NO   | PRI | NULL    | auto_increment |
| name   | text | YES  |     | NULL    |                |
| userid | text | YES  |     | NULL    |                |
| pwd    | text | YES  |     | NULL    |                |
| email  | text | YES  |     | NULL    |                |
| phone  | text | YES  |     | NULL    |                |
| admin  | text | YES  |     | NULL    |                |

```
7 rows in set (0.00 sec)
```

```
mysql> select * from member;
```

| num | name | userid | pwd  | email         | phone         | admin |
|-----|------|--------|------|---------------|---------------|-------|
| 1   | 일이삼  | one    | 1111 | test1@iot.com | 010-1234-5678 | 0     |
| 2   | 사오육  | admin  | 1111 | test2@iot.com | 010-9999-9999 | 1     |
| 3   | 칠팔구  | two    | 1111 | test3@iot.com | 010-8765-4321 | 0     |

```
3 rows in set (0.00 sec)
```

## 미리 값이 입력된 데이터베이스 사용

# 전체조회 페이지

```
member_list.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <%@ page import="java.sql.*" %>
4 <!DOCTYPE html>
5 <html>
6   <head>
7     <meta charset="utf-8">
8     <title>전체회원 조회</title>
9     <link rel="stylesheet" href="style.css">
10  </head>
11  <body>
12    <%
13      Connection conn=null;
14      Statement stmt=null;
15      ResultSet rs=null;
16
17      String url="jdbc:mysql://127.0.0.1:3306/sakila2";
18      String id="root";
19      String pass="iotiot";
20
21      String sql="select * from member;";
22
23      try{
24        Class.forName("com.mysql.cj.jdbc.Driver");
25        conn=DriverManager.getConnection(url, id, pass);
26        stmt=conn.createStatement();
27        rs=stmt.executeQuery(sql);
28
29
30        out.print("<table>");
31        out.print("<tr>");
32        out.print("<th>회원번호</th>");
33        out.print("<th>이름</th>");
34        out.print("<th>아이디</th>");
35        out.print("<th>비밀번호</th>");
36        out.print("<th>이메일</th>");
37        out.print("<th>전화번호</th>");
38        out.print("<th>관리자여부</th>");
39        out.print("</tr>");
40        while(rs.next()){
41          out.print("<tr>");
42          out.print("<td>" + rs.getString("num") + "</td>");
43          out.print("<td>" + rs.getString("name") + "</td>");
44          out.print("<td>" + rs.getString("userid") + "</td>");
45          out.print("<td>" + rs.getString("pwd") + "</td>");
46          out.print("<td>" + rs.getString("email") + "</td>");
47          out.print("<td>" + rs.getString("phone") + "</td>");
48          out.print("<td>" + rs.getString("admin") + "</td>");
49          out.print("</tr>");
50        }
51        out.print("</table>");
52      }catch(Exception e){
53        System.out.println(e);
54      }finally{
55        try{
56          if(rs!=null){rs.close();}
57          if(stmt!=null){stmt.close();}
58          if(conn!=null){conn.close();}
59        }catch(Exception e){
60          System.out.println(e);
61        }
62      }
63    }
64  </body>
65 </html>
```

```
style.css
1 @charset "utf-8";
2
3 table, td{
4   border:1px solid black;
5 }
```

| 회원번호 | 이름  | 아이디   | 비밀번호 | 이메일           | 전화번호          | 관리자여부 |
|------|-----|-------|------|---------------|---------------|-------|
| 1    | 일이삼 | one   | 1111 | test1@iot.com | 010-1234-5678 | 0     |
| 2    | 사오육 | admin | 1111 | test2@iot.com | 010-9999-9999 | 1     |
| 3    | 칠팔구 | two   | 1111 | test3@iot.com | 010-8765-4321 | 0     |

ResultSet, Statement,  
Connection은 순차적으로  
제거해 준다.

# 회원가입 페이지

```
register.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>회원가입 페이지</title>
8     <link rel="stylesheet" href="style.css">
9   </head>
10  <body>
11    <h1>여기는 회원가입페이지 입니다.</h1>
12    <form method="post" action="member_insert.jsp">
13      <table>
14        <tr>
15          <td>이름</td>
16          <td><input type="text" name="name"></td>
17        </tr>
18        <tr>
19          <td>아이디</td>
20          <td><input type="text" name="userid"></td>
21        </tr>
22        <tr>
23          <td>비밀번호</td>
24          <td><input type="password" name="pwd"></td>
25        </tr>
26        <tr>
27          <td>이메일</td>
28          <td><input type="text" name="email"></td>
29        </tr>
30        <tr>
31          <td>전화번호</td>
32          <td><input type="text" name="phone"></td>
33        </tr>
34        <tr>
35          <td>등급</td>
36          <td>
37            <input type="radio" name="admin" value="1">관리자<br>
38            <input type="radio" name="admin" value="0" checked="">일반회원
39          </td>
40        </tr>
41        <tr>
42          <td>입력</td>
43          <td><input type="submit" value="회원가입"><input type="reset" value="초기화"></td>
44        </tr>
45      </table>
46    </form>
47  </body>
48 </html>
```

여기는 회원가입페이지 입니다.

|      |   |
|------|---|
| 이름   | <input type="text"/>  |
| 아이디  | <input type="text"/>  |
| 비밀번호 | <input type="password"/>  |
| 이메일  | <input type="text"/>  |
| 전화번호 | <input type="text"/>  |
| 등급   | <input type="radio"/> 관리자<br><input checked="" type="radio"/> 일반회원    |
| 입력   | <input type="submit" value="회원가입"/> <input type="reset" value="초기화"/> |

form을 이용하여 데이터를 전송

# 입력 처리

member\_insert.jsp

```
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <%@ page import="java.sql.*"%>
4 <!DOCTYPE html>
5 <html>
6 <head>
7   <meta charset="utf-8">
8   <title>회원데이터 입력 페이지</title>
9 </head>
10 <body>
11 <%
12   Connection conn=null;
13   Statement stmt=null;
14   ResultSet rs=null;
15
16   String url="jdbc:mysql://127.0.0.1:3306/sakila2";
17   String id="root";
18   String pass="iotiot";
19
20   //기존의 Statement방식의 처리는 실행법
21   String sql="insert into member (name, userid, pwd, email, phone, admin) values (";
22
23
24   request.setCharacterEncoding("utf-8");
25   String name=request.getParameter("name");
26   String userid=request.getParameter("userid");
27   String pwd=request.getParameter("pwd");
28   String email=request.getParameter("email");
29   String phone=request.getParameter("phone");
30   String admin=request.getParameter("admin");
31
32   try{
33     //기존의 Statement방식의 처리는 실행법
34     sql+="name='"+name+"',userid='"+userid+"',pwd='"+pwd+"',email='"+email+"',phone='"+phone+"',admin='"+admin+"';";
35     Class.forName("com.mysql.cj.jdbc.Driver");
36     conn=DriverManager.getConnection(url, id, pass);
37     stmt=conn.createStatement();
38     stmt.executeUpdate(sql);
39
40   }catch(Exception e){
41     System.out.println(e);
42   }finally{
43     try{
44       if(rs!=null){rs.close();}
45       if(stmt!=null){stmt.close();}
46       if(conn!=null){conn.close();}
47     }catch(Exception e){
48       System.out.println(e);
49     }
50   }
51 %>
52
53 </body>
54 </html>
```

여기는 회원가입페이지 입니다.

|      |  |
|------|--|
| 이름   | 십십일  |
| 아이디  | three  |
| 비밀번호 | ....   |
| 이메일  | test4@iot.com  |
| 전화번호 | 010-0000-0000  |
| 등급   | <input type="radio"/> 관리자<br><input checked="" type="radio"/> 일반회원     |
| 입력   | <input type="button" value="회원가입"/> <input type="button" value="초기화"/> |

| 회원번호 | 이름  | 아이디   | 비밀번호 | 이메일           | 전화번호          | 관리자여부 |
|------|-----|-------|------|---------------|---------------|-------|
| 1    | 일이삼 | one   | 1111 | test1@iot.com | 010-1234-5678 | 0     |
| 2    | 사오육 | admin | 1111 | test2@iot.com | 010-9999-9999 | 1     |
| 3    | 칠팔구 | two   | 1111 | test3@iot.com | 010-8765-4321 | 0     |
| 4    | 십십일 | three | 3333 | test4@iot.com | 010-0000-0000 | 0     |

form을 이용하여 데이터를 전송

# prepared Statement의 사용

```
Connection conn=null;
//Statement stmt=null;
PreparedStatement pstmt=null;
ResultSet rs=null;

String url="jdbc:mysql://127.0.0.1:3306/sakila2";
String id="root";
String pass="iotiot";

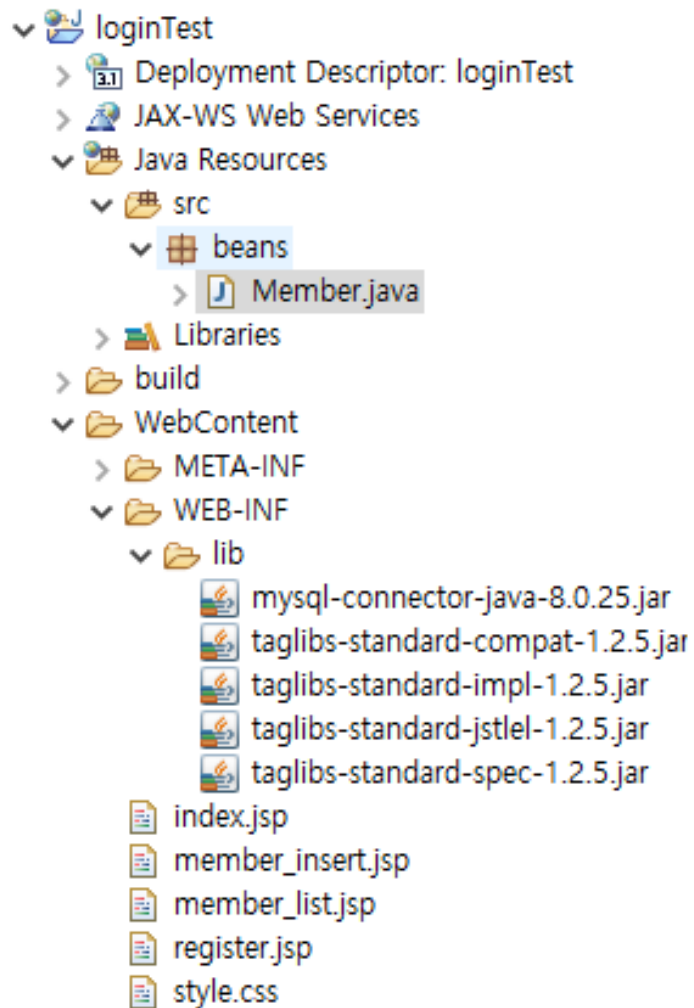
//기존의 Statement방식의 쿼리문 실행법
//String sql="insert into member (name, userid, pwd, email, phone, admin) values (";
//PreparedStatement방식의 쿼리문 실행법
String sql="insert into member (name, userid, pwd, email, phone, admin) values (?, ?, ?, ?, ?, ?)";

request.setCharacterEncoding("utf-8");
String name=request.getParameter("name");
String userid=request.getParameter("userid");
String pwd=request.getParameter("pwd");
String email=request.getParameter("email");
String phone=request.getParameter("phone");
String admin=request.getParameter("admin");

//기존의 Statement방식의 쿼리문 실행법
//sql+=" '"+name+"', '"+userid+"', '"+pwd+"', '"+email+"', '"+phone+"', '"+admin+"'";";
Class.forName("com.mysql.cj.jdbc.Driver");
conn=DriverManager.getConnection(url, id, pass);
//stmt=conn.createStatement();
pstmt=conn.prepareStatement(sql);
pstmt.setString(1, name);
pstmt.setString(2, userid);
pstmt.setString(3, pwd);
pstmt.setString(4, email);
pstmt.setString(5, phone);
pstmt.setString(6, admin);
//stmt.executeUpdate(sql);
pstmt.executeUpdate();
```

**?의 수와 위치에 주의하며 사용해야 한다.**

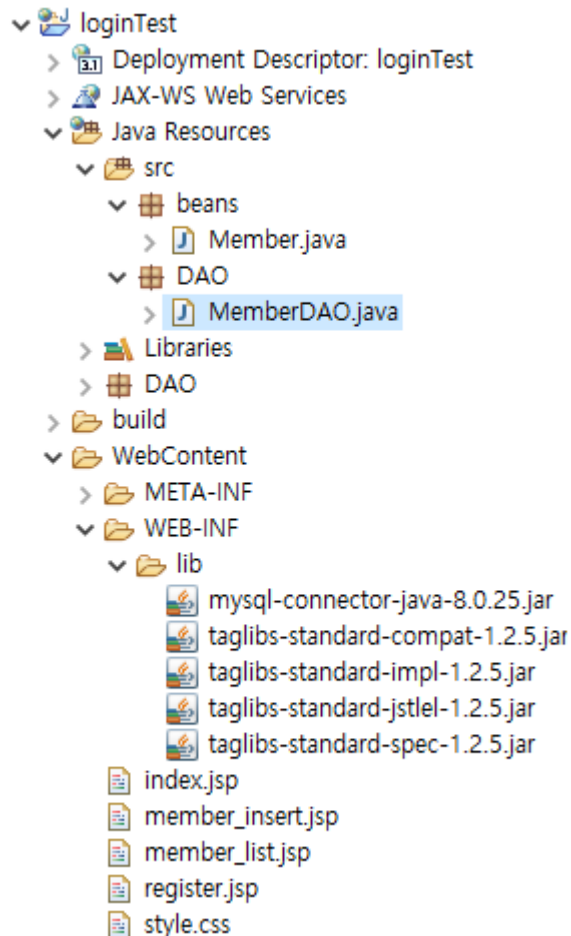
# beans 제작



```
Member.java
1 package beans;
2
3 public class Member {
4     private String num;
5     private String name;
6     private String userid;
7     private String pwd;
8     private String email;
9     private String phone;
10    private String admin;
11    public String getNum() {
12        return num;
13    }
14    public void setNum(String num) {
15        this.num = num;
16    }
17    public String getName() {
18        return name;
19    }
20    public void setName(String name) {
21        this.name = name;
22    }
23    public String getUserid() {
24        return userid;
25    }
26    public void setUserid(String userid) {
27        this.userid = userid;
28    }
29    public String getPwd() {
30        return pwd;
31    }
32    public void setPwd(String pwd) {
33        this.pwd = pwd;
34    }
35    public String getEmail() {
36        return email;
37    }
38    public void setEmail(String email) {
39        this.email = email;
40    }
41    public String getPhone() {
42        return phone;
43    }
44    public void setPhone(String phone) {
45        this.phone = phone;
46    }
47    public String getAdmin() {
48        return admin;
49    }
50    public void setAdmin(String admin) {
51        this.admin = admin;
52    }
53 }
```

## 데이터를 전송하기 위한 DTO를 제작

# DAO 제작



MemberDAO.java

```
1 package DAO;
2
3 public class MemberDAO {
4     private MemberDAO() {
5
6     }
7
8     private static MemberDAO instance=new MemberDAO();
9
10    public static MemberDAO getInstance() {
11        return instance;
12    }
13 }

//커넥션 연결 처리(이전에는 jsp마다 개별적으로 진행했던 연결을 통합)
public Connection getConnection() throws Exception{
    Connection conn=null;
    String url="jdbc:mysql://127.0.0.1:3306/board";
    String id="root";
    String pass="iotiot";

    Class.forName("com.mysql.cj.jdbc.Driver");
    conn=DriverManager.getConnection(url, id, pass);
    return conn;
}
```

## 싱글톤 패턴을 구현하기 위한 형태 제작



# 도메인 컨트롤

```
index.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>메인페이지</title>
8   </head>
9   <body>
10    <h1>여기는 메인페이지 입니다.</h1>
11    <a href="login.do">로그인하러 가기</a>
12    <a href="member_list.jsp">전체멤버 조회</a>
13    <a href="register.jsp">회원가입</a>
14  </body>
15 </html>
```

```
login.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta charset="utf-8">
7     <title>로그인 페이지</title>
8   </head>
9   <body>
10    <h1>여기는 로그인페이지 입니다.</h1>
11  </body>
12 </html>
```

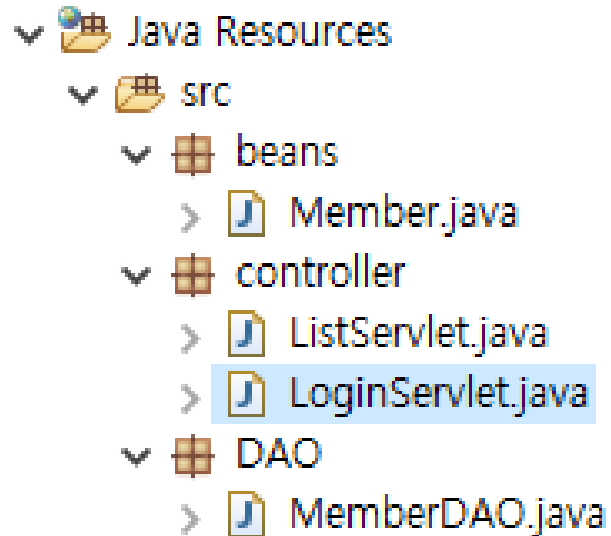
```
LoginServlet.java
1 package controller;
2
3 import java.io.IOException;
4
5 @WebServlet("/login.do")
6 public class LoginServlet extends HttpServlet {
7   private static final long serialVersionUID = 1L;
8
9   protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
10     RequestDispatcher dispatcher=request.getRequestDispatcher("login.jsp");
11     dispatcher.forward(request, response);
12   }
13
14   protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
15   }
16 }
17
18
19
20 }
```

← → ↻ ⓘ 127.0.0.1:8080/loginTest/login.do

여기는 로그인페이지 입니다.

서블릿을 이용하면 도메인 컨트롤이 가능하다.  
~.jsp로 직접 경로를 지정하지 않아도 접근 가능하다.

# 도메인 컨트롤



```
LoginServlet.java
1 package controller;
2
3 import java.io.IOException;
4
5
6
7
8 @WebServlet("/login.do")
9 public class LoginServlet extends HttpServlet {
10     private static final long serialVersionUID = 1L;
11
12     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
13         RequestDispatcher dispatcher=request.getRequestDispatcher("login.jsp");
14         dispatcher.forward(request, response);
15     }
16
17     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
18
19     }
20 }
```

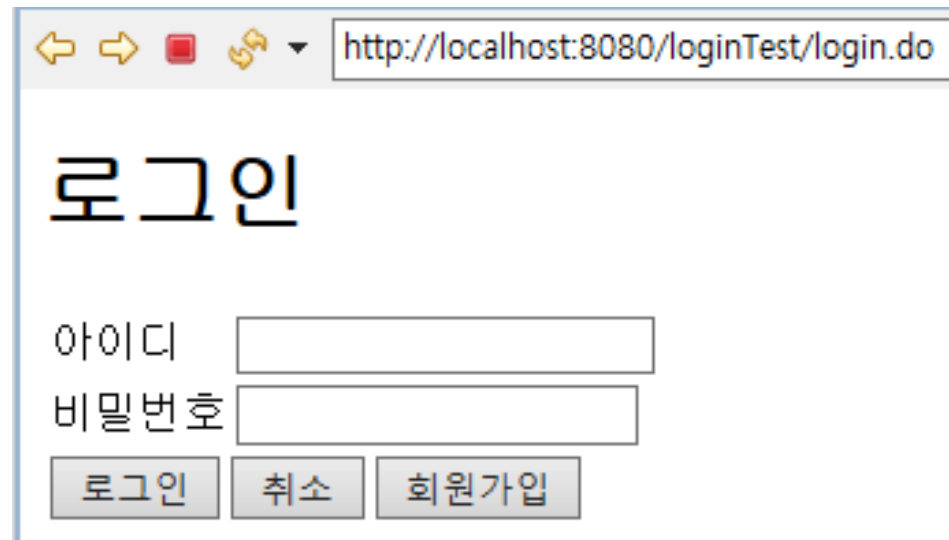
← → ↻ ⓘ 127.0.0.1:8080/loginTest/list.do

| 회원번호 | 이름  | 아이디   | 비밀번호 | 이메일           | 전화번호          | 관리자여부 |
|------|-----|-------|------|---------------|---------------|-------|
| 1    | 일이삼 | one   | 1111 | test1@iot.com | 010-1234-5678 | 0     |
| 2    | 사오육 | admin | 1111 | test2@iot.com | 010-9999-9999 | 1     |
| 3    | 칠팔구 | two   | 1111 | test3@iot.com | 010-8765-4321 | 0     |
| 4    | 십십일 | three | 3333 | test4@iot.com | 010-0000-0000 | 0     |

jsp의 파일명을 보이지 않도록 할 수 있다.

# 로그인화면 제작

```
<body>
  <h1>로그인</h1>
  <form method="post" action="login.do">
    <table>
      <tr>
        <td>아이디</td>
        <td><input type="text" name="userid"></td>
      </tr>
      <tr>
        <td>비밀번호</td>
        <td><input type="password" name="pwd"></td>
      </tr>
      <tr>
        <td colspan="3">
          <input type="submit" value="로그인">
          <input type="reset" value="취소">
          <input type="button" value="회원가입" onclick="reg()">
        </td>
      </tr>
    </table>
  </form>
  <script>
    function reg(){
      location.href("register.jsp");
    }
  </script>
</body>
```



http://localhost:8080/loginTest/login.do

## 로그인

아이디

비밀번호

## login.do로 서블릿을 실행

# 입력체크 제작

```
<input type="submit" value="로그인" onclick="return check()">
<input type="reset" value="취소">
<input type="button" value="회원가입" onclick="reg()">
</td>
</tr>
</table>
</form>
<script>
function reg(){
    location.href("register.jsp");
}
function check(){
    if(document.frm.userid.value.length==0){
        alert("아이디를 작성하셔야 합니다.");
        frm.userid.focus();
        return false;
    }
    if(document.frm.pwd.value==""){
        alert("비밀번호를 입력해 주세요");
        frm.pwd.focus();
        return false;
    }
    return true;
}
</script>
```

## 로그인

아이디

비밀번호

로그인

취소

회원가입

localhost:8080 내용:

아이디를 작성하셔야 합니다.

확인

## 로그인

아이디

비밀번호

로그인

취소

회원가입

localhost:8080 내용:

비밀번호를 입력해 주세요

확인

# 자바스크립트로 편리하게 제작 가능

# DAO 로그인 처리

```
//사용자 인증
public int userCheck(String userid, String pwd) {
    int result=-1;
    String sql="select pwd from member where userid=?";
    Connection conn=null;
    PreparedStatement pstmt=null;
    ResultSet rs=null;
    try {
        conn=getConnection();
        pstmt=conn.prepareStatement(sql);
        pstmt.setString(1, userid);
        rs=pstmt.executeQuery();
        if(rs.next()) {
            if(rs.getString("pwd").equals(pwd) && rs.getString("pwd")!=null) {
                result=1;//비밀번호가 null이 아니고 입력값과 동일한 경우
            }else {
                result=0;//null이거나 입력값과 다른 값을 입력한 경우
            }
        }else {
            result=-1;//아이디가 존재하지 않는 경우
        }
    }catch(Exception e) {
        System.out.println("로그인 인증 중 오류발생 : "+e);
    }finally{
        try{
            if(rs!=null){rs.close();}
            if(pstmt!=null){pstmt.close();}
            if(conn!=null){conn.close();}
        }catch(Exception e){
            System.out.println(e);
        }
    }
    return result;//계산된 결과를 return
}
```

```
//사용자 정보 조회
public Member getMember(String userid) {
    Member m=null;
    String sql="select * from member where userid=?";
    Connection conn=null;
    PreparedStatement pstmt=null;
    ResultSet rs=null;
    try {
        conn=getConnection();
        pstmt=conn.prepareStatement(sql);
        pstmt.setString(1, userid);
        rs=pstmt.executeQuery();
        if(rs.next()) {
            m=new Member();
            m.setNum(rs.getString("num"));
            m.setName(rs.getString("name"));
            m.setUserid(rs.getString("userid"));
            m.setPwd(rs.getString("pwd"));
            m.setEmail(rs.getString("email"));
            m.setPhone(rs.getString("phone"));
            m.setAdmin(rs.getString("admin"));
        }
    }catch(Exception e) {
        System.out.println("정보조회 중 오류발생 : "+e);
    }finally{
        try{
            if(rs!=null){rs.close();}
            if(pstmt!=null){pstmt.close();}
            if(conn!=null){conn.close();}
        }catch(Exception e){
            System.out.println(e);
        }
    }
    return m;
}
```

## memberDAO 부분에 입력할 것

# 서블릿 수정

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    RequestDispatcher dispatcher=request.getRequestDispatcher("login.jsp");
    dispatcher.forward(request, response);
}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    //post처리를 진행하게 되는 경우 실행되는 부분
    String url="login.jsp";
    request.setCharacterEncoding("utf-8");

    String userid=request.getParameter("userid");
    String pwd=request.getParameter("pwd");

    MemberDAO mDAO=MemberDAO.getInstance();//싱글톤을 유지하기 위함
    int result=mDAO.userCheck(userid, pwd);//만들어 둔 유저체크를 실행함

    if(result==1) {
        Member m=mDAO.getMember(userid);//만들어둔 회원정보 연람실행
        HttpSession session=request.getSession();//서블릿 단위에서 세션을 제작
        session.setAttribute("loginUser", m);//데이터 객체를 세션에 주입
        request.setAttribute("message", "회원 가입에 성공했습니다.");
        url="main.jsp";//로그인 성공시 이동할 페이지
    }else if(result==0) {
        request.setAttribute("message", "비밀번호가 맞지 않습니다.");
    }else if(result==-1) {
        request.setAttribute("message", "존재하지 않는 회원입니다.");
    }
    RequestDispatcher dis=request.getRequestDispatcher(url);
    dis.forward(request, response);
}
```

## LoginServlet 부분에 작성할 것

# 로그인 완료페이지

```
main.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
4 <c:if test="${empty loginUser }">
5   <jsp:forward page="login.do"></jsp:forward>
6 </c:if>
7 <!DOCTYPE html>
8 <html>
9   <head>
10    <meta charset="utf-8">
11    <title>메인 페이지</title>
12  </head>
13  <body>
14    <h1>안녕하세요 ${loginUser.name }, ${loginUser.userid }님</h1>
15    <a href="logout.do">로그아웃</a><br>
16    <a href="memberUpdate.do?userid=${loginUser.userid }">회원정보 수정</a>
17  </body>
18 </html>
```

localhost:8080/loginTest/login.do

안녕하세요 일이삼, one님

로그인

아이디

비밀번호

로그인

취소

회원가입

post시와 get시의 동작이 다름에 유의할 것

# 회원가입 서블릿

```
JoinServlet.java
1 package controller;
2
3 import java.io.IOException;
4 import javax.servlet.*;
5 import javax.servlet.annotation.*;
6 import javax.servlet.http.*;
7
8 import beans.Member;
9 import DAO.MemberDAO;
10
11 @WebServlet("/join.do")
12 public class JoinServlet extends HttpServlet {
13     private static final long serialVersionUID = 1L;
14
15     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
16         RequestDispatcher dis=request.getRequestDispatcher("/register.jsp");
17         dis.forward(request, response);
18     }
19
20     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
21
22     }
23 }
```

회원가입 처리를 진행해줄 join.do



# 회원가입 페이지

```
register.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="utf-8">
7   <title>회원가입 페이지</title>
8   <link rel="stylesheet" href="style.css">
9 </head>
10 <body>
11   <h1>여기는 회원가입페이지 입니다.</h1>
12   **표시는 필수 입력 항목입니다.
13   <form method="post" action="join.do" name="frm">
14     <table>
15       <tr>
16         <td>이름</td>
17         <td><input type="text" name="name" size="20">*</td>
18       </tr>
19       <tr>
20         <td>아이디</td>
21         <td>
22           <input type="text" name="userid">*
23           <input type="hidden" name="reid">
24           <input type="button" value="중복확인" onclick="idCheck()">
25         </td>
26       </tr>
27       <tr>
28         <td>비밀번호</td>
29         <td><input type="password" name="pwd" size="20">*</td>
30       </tr>
31       <tr>
32         <td>비밀번호 재확인</td>
33         <td><input type="password" name="pwd_check" size="20">*</td>
34       </tr>
35       <tr>
36         <td>이메일</td>
37         <td><input type="text" name="email" size="20"></td>
38       </tr>
39       <tr>
40         <td>전화번호</td>
41         <td><input type="text" name="phone" size="20"></td>
42       </tr>
43       <tr>
44         <td>등급</td>
45         <td>
46           <input type="radio" name="admin" value="1">관리자<br>
47           <input type="radio" name="admin" value="0" checked="">일반회원
48         </td>
49       </tr>
50       <tr>
51         <td>입력</td>
52         <td>
53           <input type="submit" value="회원가입" onclick="return joinCheck()">
54           <input type="reset" value="초기화">
55         </td>
56       </tr>
57     </table>
58   </form>
```

```
<script>
function idCheck(){
  if(document.frm.userid.value==""){
    alert("아이디를 입력해 주십시오");
    document.frm.userid.focus();
    return;
  }
  var url="idCheck.do?userid="+document.frm.userid.value;
  var popupX = (window.screen.width/2) - (450 / 2);
  var popupY= (window.screen.height/2) - (200 / 2);
  window.open(url, "_blank_1", "toolbar=no, menubar=no, scrollbar=yes, resizable=no, width=450, height=200, left="+popupX+", top="+popupY);
}
</script>
</body>
</html>
```

여기는 회원가입페이지 입니다.

\*\*표시는 필수 입력 항목입니다.

|          |  |        |
|----------|--|--------|
| 이름       | <input name="name" size="20" type="text"/>   | *      |
| 아이디      | <input name="userid" type="text" value="four"/>  | * 중복확인 |
| 비밀번호     | <input name="pwd" size="20" type="password"/>  | *      |
| 비밀번호 재확인 | <input name="pwd_check" size="20" type="password"/>  | *      |
| 이메일      | <input name="email" size="20" type="text"/>  |        |
| 전화번호     | <input name="phone" size="20" type="text"/>  |        |
| 등급       | <input name="admin" type="radio" value="1"/> 관리자<br><input checked="" name="admin" type="radio" value="0"/> 일반회원 |        |
| 입력       | <input type="submit" value="회원가입"/> <input type="reset" value="초기화"/>  |        |



이전에 작성한 register.jsp를 수정한다.

# 아이디 중복체크

```
//아이디 중복체크 처리
public int confirmID(String userid) {
    int result=-1;
    String sql="select userid from member where userid=?";
    Connection conn=null;
    PreparedStatement pstmt=null;
    ResultSet rs=null;
    try {
        conn=getConnection();
        pstmt=conn.prepareStatement(sql);
        pstmt.setString(1, userid);
        rs=pstmt.executeQuery();
        if(rs.next()) {
            result=1;
        }else {
            result=-1;
        }
    }catch(Exception e) {
        System.out.println("정보조회 중 오류발생 : "+e);
    }finally{
        try{
            if(rs!=null){rs.close();}
            if(pstmt!=null){pstmt.close();}
            if(conn!=null){conn.close();}
        }catch(Exception e){
            System.out.println(e);
        }
    }
    return result;
}
```

MemberDAO에 다음의 메서드를 추가

# 중복체크용 서블릿

```
IdCheckServlet.java
1 package controller;
2
3 import java.io.IOException;
4 import javax.servlet.*;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.HttpServlet;
7 import javax.servlet.http.HttpServletRequest;
8 import javax.servlet.http.HttpServletResponse;
9
10 import DAO.MemberDAO;
11
12 @WebServlet("/idCheck.do")
13 public class IdCheckServlet extends HttpServlet {
14     private static final long serialVersionUID = 1L;
15
16     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
17         String userid=request.getParameter("userid");
18         MemberDAO m=MemberDAO.getInstance();
19         int result=m.confirmID(userid);
20         request.setAttribute("userid", userid);
21         request.setAttribute("result", result);
22         RequestDispatcher dispatcher=request.getRequestDispatcher("idCheck.jsp");
23         dispatcher.forward(request, response);
24     }
25     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
26
27     }
28 }
```

중복체크를 진행해줄 서블릿을 제작

# 중복체크 jsp

```
idCheck.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4 <!DOCTYPE html>
5 <html>
6   <head>
7     <meta charset="utf-8">
8     <title>아이디 중복확인</title>
9   </head>
10  <body>
11    <h1>아이디 중복확인</h1>
12    <form method="get" action="idCheck.do" name="frm">
13      아이디 <input type="text" name="userid" value="${userid}">
14      <input type="submit" value="중복체크">
15      <br>
16      <c:if test="${result==1 }">
17        <script>
18          //opener는 창을 켜게 만든 부모창을 의미한다.
19          opener.document.frm.userid.value="";
20        </script>
21        ${userid}는 이미 사용중인 아이디 입니다.
22      </c:if>
23      <c:if test="${result==-1 }">
24        ${userid}는 사용 가능한 아이디 입니다.
25        <input type="button" value="사용하기" onclick="idok()">
26      </c:if>
27    </form>
28  </body>
29  <script>
30    function idok(){
31      opener.document.frm.userid.value=document.frm.userid.value;
32      opener.frm.reid.value=document.frm.userid.value;
33      //self는 지금 켜져 있는 창을 의미한다.
34      self.close();
35    }
36  </script>
37 </html>
```

여기는 회원가입페이지 입니다.

\*표시는 필수 입력 항목입니다.

|          |  |        |
|----------|--|--------|
| 이름       | 이영준  | *      |
| 아이디      | four   | * 중복확인 |
| 비밀번호     |  | *      |
| 비밀번호 재확인 |  | *      |
| 이메일      |  |        |
| 전화번호     |  |        |
| 등급       | <input type="radio"/> 관리자<br><input checked="" type="radio"/> 일반회원     |        |
| 입력       | <input type="button" value="회원가입"/> <input type="button" value="초기화"/> |        |

아이디 중복확인 - Chrome

localhost:8080/loginTest/idCheck.do?userid=four

## 아이디 중복확인

아이디

four는 이미 사용중인 아이디 입니다.

여기는 회원가입페이지 입니다.

\*표시는 필수 입력 항목입니다.

|          |  |        |
|----------|--|--------|
| 이름       | 이영준  | *      |
| 아이디      | seven  | * 중복확인 |
| 비밀번호     |  | *      |
| 비밀번호 재확인 |  | *      |
| 이메일      |  |        |
| 전화번호     |  |        |
| 등급       | <input type="radio"/> 관리자<br><input checked="" type="radio"/> 일반회원     |        |
| 입력       | <input type="button" value="회원가입"/> <input type="button" value="초기화"/> |        |

아이디 중복확인 - Chrome

localhost:8080/loginTest/idCheck.do?userid=seven

## 아이디 중복확인

아이디

seven는 사용 가능한 아이디 입니다.

## 동작하는 중복체크 기능 구현

# 회원가입의 기능들

```
<script>
function idCheck(){
    if(document.frm.userid.value==""){
        alert("아이디를 입력해 주십시오");
        document.frm.userid.focus();
        return;
    }
    var url="idCheck.do?userid="+document.frm.userid.value;
    var popupX = (window.screen.width / 2)- (450 / 2);
    var popupY= (window.screen.height / 2)- (200 / 2);
    window.open(url, "_blank_1", "toolbar=no, menubar=no, scrollbar=yes, resizable=no, width=450, height=200, left="+popupX+", top="+popupY);
}
function joinCheck(){
    if(document.frm.name.value.length==0){
        alert("이름을 작성해 주세요");
        frm.name.focus();
        return false;
    }
    if(document.frm.userid.value.length==0){
        alert("아이디를 작성해 주세요");
        frm.userid.focus();
        return false;
    }
    if(document.frm.userid.value.length<4){
        alert("아이디는 4문자 이상이어야 합니다.");
        frm.userid.focus();
        return false;
    }
    if(document.frm.pwd.value==""){
        alert("암호는 반드시 입력해야 합니다.");
        frm.pwd.focus();
        return false;
    }
    if(document.frm.pwd.value!=document.frm.pwd_check.value){
        alert("암호가 일치하지 않습니다.");
        frm.pwd.focus();
        return false;
    }
    if(document.frm.reid.value.length==0){
        alert("아이디 중복체크가 이루어지지 않았습니다.");
        frm.userid.focus();
        return false;
    }
    return true;
}
}
</script>
```

## register.jsp form의 유효성을 체크하는 기능 구현

# 회원가입 메서드

```
//회원가입 처리
public int insertMember(Member m) {
    int result=-1;
    String sql="insert into member (name, userid, pwd, email, phone, admin) values (?, ?, ?, ?, ?, ?)";
    Connection conn=null;
    PreparedStatement pstmt=null;
    try {
        conn=getConnection();
        pstmt=conn.prepareStatement(sql);
        pstmt.setString(1, m.getName());
        pstmt.setString(2, m.getUserid());
        pstmt.setString(3, m.getPwd());
        pstmt.setString(4, m.getEmail());
        pstmt.setString(5, m.getPhone());
        pstmt.setString(6, m.getAdmin());
        result=pstmt.executeUpdate(); //변경된 row수가 return이므로 insert문은 1만을 반환한다.
    } catch (Exception e) {
        System.out.println("회원가입 중 오류발생 : "+e);
    } finally{
        try{
            if(pstmt!=null){pstmt.close();}
            if(conn!=null){conn.close();}
        } catch (Exception e){
            System.out.println(e);
        }
    }
    return result;
}
```

## MemberDAO 부분에 작성

# 회원가입 서블릿

```
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    request.setCharacterEncoding("utf-8");
    String name=request.getParameter("name");
    String userid=request.getParameter("userid");
    String pwd=request.getParameter("pwd");
    String email=request.getParameter("email");
    String phone=request.getParameter("phone");
    String admin=request.getParameter("admin");

    Member m=new Member();
    m.setName(name);
    m.setUserId(userid);
    m.setPwd(pwd);
    m.setEmail(email);
    m.setPhone(phone);
    m.setAdmin(admin);

    MemberDAO mDAO=MemberDAO.getInstance();
    int result=mDAO.insertMember(m);

    HttpSession session=request.getSession();
    if(result==1) {
        session.setAttribute("userid", m.getUserId());
        request.setAttribute("message", "회원가입에 성공했습니다.");
    }else {
        request.setAttribute("message", "회원가입에 실패했습니다.");
    }
    RequestDispatcher dispatcher=request.getRequestDispatcher("login.do");
    dispatcher.forward(request, response);
}
```

## JoinServlet에 작성

# 로그아웃 처리

```
LogoutServlet.java
1 package controller;
2
3 import java.io.IOException;
4 import javax.servlet.*;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.*;
7
8 @WebServlet("/logout.do")
9 public class LogoutServlet extends HttpServlet {
10     private static final long serialVersionUID = 1L;
11
12     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
13         HttpSession session=request.getSession();
14         session.invalidate();
15         RequestDispatcher dispatcher=request.getRequestDispatcher("login.jsp");
16         dispatcher.forward(request, response);
17     }
18
19     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
20         doGet(request, response);
21     }
22 }
```

로그인

아이디

비밀번호

안녕하세요 사오육, admin님

[로그아웃](#)  
[회원정보 수정](#)

로그인

아이디

비밀번호

## LogoutServlet 작성



# 회원정보 수정 서블릿

```
MemberUpdateServlet.java
1 package controller;
2
3 import java.io.IOException;
4 import javax.servlet.*;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.HttpServlet;
7 import javax.servlet.http.HttpServletRequest;
8 import javax.servlet.http.HttpServletResponse;
9
10 import DAO.MemberDAO;
11 import beans.Member;
12
13 @WebServlet("/memberUpdate.do")
14 public class MemberUpdateServlet extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
18         String userid=request.getParameter("userid");
19         MemberDAO mDAO=MemberDAO.getInstance();
20         Member m=mDAO.getMember(userid);
21         request.setAttribute("mem", m);
22         RequestDispatcher dispatcher=request.getRequestDispatcher("memberUpdate.jsp");
23         dispatcher.forward(request, response);
24     }
25     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
26
27     }
28 }
```

## MemberUpdateServlet 작성

# 회원정보 수정 페이지

```
memberUpdate.jsp
1 <%@ page language="java" contentType="text/html; charset=utf-8"
2   pageEncoding="utf-8"%>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
4 <!DOCTYPE html>
5 <html>
6   <head>
7     <meta charset="utf-8">
8     <title>회원 정보 수정</title>
9   </head>
10  <body>
11    <form method="post" action="memberUpdate.do" name="frm">
12      <table>
13        <tr>
14          <td>이름</td>
15          <td><input type="text" name="name" size="20" value="${mem.name}" readonly></td>
16        </tr>
17        <tr>
18          <td>아이디</td>
19          <td><input type="text" name="userid" size="20" value="${mem.userid}" readonly></td>
20        </tr>
21        <tr>
22          <td>비밀번호</td>
23          <td><input type="password" name="pwd" size="20" value=""></td>
24        </tr>
25        <tr>
26          <td>비밀번호 재확인</td>
27          <td><input type="password" name="pwd_check" size="20" value=""></td>
28        </tr>
29        <tr>
30          <td>이메일</td>
31          <td><input type="text" name="email" size="20" value="${mem.email}"></td>
32        </tr>
33        <tr>
34          <td>전화번호</td>
35          <td><input type="text" name="phone" size="20" value="${mem.phone}"></td>
36        </tr>
37        <tr>
38          <td>등급</td>
39          <td>
40            <c:choose>
41              <c:when test="${mem.admin==0}">
42                <input type="radio" name="admin" value="1">관리자<br>
43                <input type="radio" name="admin" value="0" checked="">일반회원
44              </c:when>
45              <c:otherwise>
46                <input type="radio" name="admin" value="1" checked="">관리자<br>
47                <input type="radio" name="admin" value="0">일반회원
48              </c:otherwise>
49            </c:choose>
50          </td>
51        </tr>
52      </table>
53      <td>입력</td>
54      <td>
55        <input type="submit" value="회원가입" onclick="return joinCheck()">
56        <input type="reset" value="초기화">
57      </td>
58    </tr>
59  </table>
60 </form>
61
```

```
<script>
function idCheck(){
  if(document.frm.userid.value==""){
    alert("아이디를 입력해 주세요.");
    document.frm.userid.focus();
    return;
  }
  var url="idCheck.do?userid="+document.frm.userid.value;
  var popupX = (window.screen.width / 2) - (450 / 2);
  var popupY = (window.screen.height / 2) - (200 / 2);
  window.open(url, "_blank_1", "toolbar=no, menubar=no, scrollbar=yes, resizable=no, width=450, height=200, left="+popupX+", top="+popupY);
}
function joinCheck(){
  if(document.frm.name.value.length==0){
    alert("이름을 작성해 주세요.");
    frm.name.focus();
    return false;
  }
  if(document.frm.userid.value.length==0){
    alert("아이디를 작성해 주세요.");
    frm.userid.focus();
    return false;
  }
  if(document.frm.userid.value.length<4){
    alert("아이디는 4문자 이상이어야 합니다.");
    frm.userid.focus();
    return false;
  }
  if(document.frm.pwd.value==""){
    alert("암호는 반드시 입력해야 합니다.");
    frm.pwd.focus();
    return false;
  }
  if(document.frm.pwd.value!=document.frm.pwd_check.value){
    alert("암호가 일치하지 않습니다.");
    frm.pwd.focus();
    return false;
  }
  if(document.frm.reid.value.length==0){
    alert("아이디 중복체크가 이루어지지 않았습니다.");
    frm.userid.focus();
    return false;
  }
  return true;
}
</script>
```

register.jsp의 내용을 수정하여 작성

# 회원정보 수정 메서드

```
//회원 정보 변경 기능
public int updateMember(Member m) {
    int result=-1;
    String sql="update member set pwd=?, email=?, phone=?, admin=? where userid=?";
    Connection conn=null;
    PreparedStatement pstmt=null;
    try {
        conn=getConnection();
        pstmt=conn.prepareStatement(sql);
        pstmt.setString(1,m.getPwd());
        pstmt.setString(2,m.getEmail());
        pstmt.setString(3,m.getPhone());
        pstmt.setString(4,m.getAdmin());
        pstmt.setString(5,m.getUserid());
        result=pstmt.executeUpdate();
    }catch(Exception e) {
        System.out.println("회원정보 수정 중 오류발생 : "+e);
    }finally{
        try{
            if(pstmt!=null){pstmt.close();}
            if(conn!=null){conn.close();}
        }catch(Exception e){
            System.out.println(e);
        }
    }
    return result;
}
```

memberDAO 내부에 작성

# 회원정보 수정 서블릿

```
MemberUpdateServlet.java
1 package controller;
2
3 import java.io.IOException;
4 import javax.servlet.*;
5 import javax.servlet.annotation.WebServlet;
6 import javax.servlet.http.HttpServlet;
7 import javax.servlet.http.HttpServletRequest;
8 import javax.servlet.http.HttpServletResponse;
9
10 import DAO.MemberDAO;
11 import beans.Member;
12
13 @WebServlet("/memberUpdate.do")
14 public class MemberUpdateServlet extends HttpServlet {
15     private static final long serialVersionUID = 1L;
16
17     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
18         String userid=request.getParameter("userid");
19         MemberDAO mDAO=MemberDAO.getInstance();
20         Member m=mDAO.getMember(userid);
21         request.setAttribute("mem", m);
22         RequestDispatcher dispatcher=request.getRequestDispatcher("memberUpdate.jsp");
23         dispatcher.forward(request, response);
24     }
25     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
26         request.setCharacterEncoding("utf-8");
27         String userid=request.getParameter("userid");
28         String pwd=request.getParameter("pwd");
29         String email=request.getParameter("email");
30         String phone=request.getParameter("phone");
31         String admin=request.getParameter("admin");
32
33         Member m=new Member();
34         m.setUserid(userid);
35         m.setPwd(pwd);
36         m.setEmail(email);
37         m.setPhone(phone);
38         m.setAdmin(admin);
39
40         MemberDAO mDAO=MemberDAO.getInstance();
41
42         mDAO.updateMember(m);
43         response.sendRedirect("login.do");
44     }
45 }
```

## MemberUpdateServlet.doPost()에 작성

# 로그인 서블릿 수정

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
    String url="login.jsp";  
    HttpSession session=request.getSession();  
    if(session.getAttribute("loginUser")!=null) {  
        url="main.jsp";  
    }  
    RequestDispatcher dispatcher=request.getRequestDispatcher(url);  
    dispatcher.forward(request, response);  
}
```

**LoginServlet.doGet()부분을 수정**  
**로그인이 된 회원의 경우 인증 없이 페이지 이동**