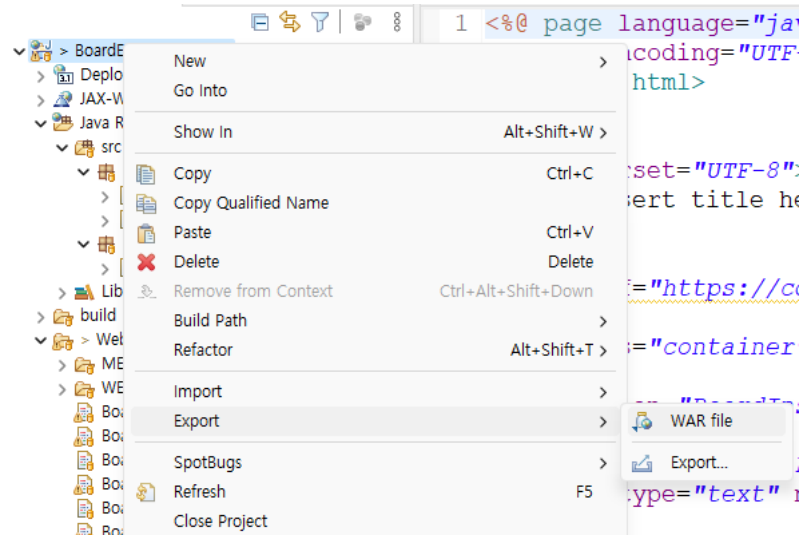


# 20230117 JDBC

## ■ Hour 1

cf) 프로젝트 export → WAR file의 형태로 export



## Summer Note

### • 주의사항

1. DB에 필드 사이즈를 크게 LONGTEXT (4GB 까지 가능)로 사용해야만 한다.  
(code 형태로 DB에 등록)
2. 전송 방식을 "post"로 설정해야만 한다.
3. Bootstrap 5 를 이용하려면 without Bootstrap을 사용한 뒤 Bootstrap 5 를 따로 적용해야됨

### • 모바일 화면 보기

```
<meta name="viewport" content="width=device-width, initial-scale=1" />
```

## StartBootstrap

사이트 전체 구성을 쉽고 빠르게 템플릿으로 이용하는 사이트

## ■ Hour 2

### file 전송

### form 형태로 제목, 내용, 파일 전송

```
<form action="FboardInsert.jsp" method="post" enctype="multipart/form-data">
<div class="mb-3">
  <label for="exampleFormControlInput1" class="form-label">제목</label>
  <input type="text" name = "fbtitle" class="form-control" id="exampleFormControlInput1">
</div>
<div class="mb-3">
  <label for="exampleFormControlTextarea1" class="form-label">내용</label>
  <textarea class="form-control" name = "fbcontent" id="exampleFormControlTextarea1" rows="3"></textarea>
</div>
<div class="input-group mb-3">
```

```
<label class="input-group-text" for="inputGroupFile01">Upload</label>
<input type="file" class="form-control" id="inputGroupFile01" name="fbfile">
</div>
<button type="submit" class="btn btn-primary">등록</button>
</form>
```

- 사진 전송을 위해서는 multipart/form-data 를 이용해야 한다.
- 이를 이용하면 key 값, value 값 형태로 데이터를 넘기는 기존의 방식과는 다르게 전체 data를 하나의 덩어리로 넘긴다.
- 따라서 하나의 덩어리에서 원하는 data를 각각 추출해서 DB에 넣어야한다.  
(사진 자체는 이름만 DB에 올라가고 사진 자체는 서버의 특정 위치로 전송된다.)

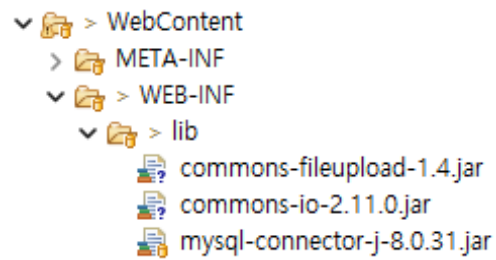
## 파일 추출을 위한 준비 단계

- ▼ commons-fileupload-1.4.jar (lib에 설치)

[https://commons.apache.org/proper/commons-fileupload/download\\_fileupload.cgi](https://commons.apache.org/proper/commons-fileupload/download_fileupload.cgi)

- ▼ commons-io-2.11.0.jar (lib에 설치)

[https://commons.apache.org/proper/commons-io/download\\_io.cgi](https://commons.apache.org/proper/commons-io/download_io.cgi)



```
<%@ page import="java.io.*" %>
<%@ page import="org.apache.commons.fileupload.*" %>
<%@ page import="org.apache.commons.fileupload.disk.*" %>
<%@ page import="org.apache.commons.fileupload.servlet.*" %>
```

## ■ Hour 3

### 추출에 필요한 기본 메서드 정리

#### isFormField()

- 요소가 key 값, value 값 쌍으로 구성된 필드를 찾아낸다.

#### getFieldName()

- key 값 속성을 추출

#### getString()

- value 값을 추출

#### getName()

- 사진(파일) 이름을 추출

#### get()

- 사진(파일) 자체를 추출

### form 형태로 전송한 데이터를 DB에 insert

```
<%
request.setCharacterEncoding("UTF-8"); // 한글 처리
```

```

String fbtitle = null;          // 게시물 제목
String fbwriter = "asd";       // 작성자
String fbcontent = null;       // 게시물 내용
String fbfile = null;          // 사진 이름
byte[] fbdata = null;          // 사진

ServletFileUpload sfu = new ServletFileUpload(new DiskFileItemFactory());

List items = sfu.parseRequest(request);

Iterator iter = items.iterator();

while(iter.hasNext()){
    FileItem item = (FileItem) iter.next();
    String name = item.getFieldName();    // key 값 속성 추출
    if(item.isFormField()) {    // key 값, value 값 형태의 data 추출
        String value = item.getString("UTF-8");
        if(name.equals("fbtitle")){
            fbtitle = value;
        } else if (name.equals("fbcontent")) {
            fbcontent = value;
        }
    } else {
        if(name.equals("fbfile")){
            fbfile = item.getName();
            fbdata = item.get();
            String root = application.getRealPath(java.io.File.separator);
            FileUtil.saveImage(root, fbfile, fbdata);
        }
    }
}

int result = FboardDAO.insert(fbtitle, fbwriter, fbcontent, fbfile);

if(result == 1){
    out.print("success");
} else {
    out.print("fail");
}

%>

```

#### DiskFileItemFactory 클래스

- 업로드 파일의 크기가 지정한 크기를 넘기 전까지는 업로드 한 파일 데이터를 메모리에 저장하고 지정한 크기를 넘길 경우 임시 디렉터리에 파일로 저장

### 파일을 서버에 저장하기 위한 방법

- DB내부에 파일을 직접 넣는 방법 (Summernote 사용)
- 서버의 특정 위치에 파일을 그대로 저장하고 DB에는 이름과 경로만 저장하는 방법 (주로사용)
  - 전자의 경우는 파일의 크기에 비례해 테이블의 크기가 기하급수적으로 커져 유지비용 증가

### 파일을 서버의 특정 위치에 전송하는 방법

util → FileUtil.java

```

package util;

import java.io.*;

public class FileUtil {
    public static void saveImage(String root, String fbname, byte[] data) throws IOException {
        root += "/images";
        File f = new File(root);
        if(!f.exists()) {
            f.mkdirs();    // images 폴더가 없으면 생성
        }

        f = new File(root + "/" + fbname);
        FileOutputStream out = new FileOutputStream(f);
        out.write(data);
        out.close();
    }
}

```

```
}  
}
```

## 파일 업로드

### ▼ FboardDAO.java

```
package jdbc;  
  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.sql.SQLException;  
  
import javax.naming.NamingException;  
  
import util.ConnectionPool;  
  
public class FboardDAO {  
  
    public static int insert(String btitle, String bwriter, String bcontent, String bfile) throws NamingException, SQLException {  
  
        String sql = "INSERT INTO fboard (btitle, bwriter, bcontent, bfile) VALUES(?,?,?,?)";  
  
        Connection conn = ConnectionPool.get();  
  
        PreparedStatement pstmt = conn.prepareStatement(sql);  
        pstmt.setString(1, btitle);  
        pstmt.setString(2, bwriter);  
        pstmt.setString(3, bcontent);  
        pstmt.setString(4, bfile);  
  
        int result = pstmt.executeUpdate();  
  
        return result;  
    }  
  
}
```

### ▼ FboardInsert.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"  
    pageEncoding="UTF-8"%>  
  
<%@ page import="java.util.*"%>  
<%@ page import="java.io.*"%>  
<%@ page import="org.apache.commons.fileupload.*"%>  
<%@ page import="org.apache.commons.fileupload.disk.*"%>  
<%@ page import="org.apache.commons.fileupload.servlet.*"%>  
<%@ page import="jdbc.*"%>  
<%@ page import="util.FileUtil"%>  
  
<%  
    request.setCharacterEncoding("utf-8"); //한글 처리  
  
    String fbtitle = null; //게시물 제목  
    String fbcontent = null; //게시물 내용  
    String fbfile = null; //사진 이름  
    String fbwriter = "someone"; //작성자는 임시로 설정, 실제로는 세션에서 읽어오자  
    byte[] fbu = null; //사진  
  
    ServletFileUpload sfu = new ServletFileUpload(new DiskFileItemFactory());  
  
    List items = sfu.parseRequest(request);  
  
    Iterator iter = items.iterator();  
  
    while(iter.hasNext()) {  
        FileItem item = (FileItem) iter.next();  
        String name = item.getFieldName(); //키값 속성 추출  
        if(item.isFormField()) { // 키값 밸류값 형태의 데이터 추출  
            String value = item.getString("utf-8");  
            if (name.equals("fbtitle")) fbtitle = value;  
            else if (name.equals("fbcontent")) fbcontent = value;  
        } else {  
            if (name.equals("fbfile")) {  
                fbfile = item.getName();  
                fbu = item.get();  
                String root = application.getRealPath(java.io.File.separator);  
            }  
        }  
    }  
}
```

```

        FileUtil.saveImage(root, fbfile, fbv);
    }
}

int result = FboardDAO.insert(fbtitle, fbwriter, fbcontent, fbfile);

if (result == 1) {
    out.print("success");
} else {
    out.print("fail");
}
}

%>

```

#### ▼ FboardDTO.java

```

package jdbc;

public class FboardDTO {

    private String bno;
    private String btitle;
    private String bwriter;
    private String bcontent;
    private String bfile;
    private String bdate;
    public String getBno() {
        return bno;
    }
    public void setBno(String bno) {
        this.bno = bno;
    }
    public String getBtitle() {
        return btitle;
    }
    public void setBtitle(String btitle) {
        this.btitle = btitle;
    }
    public String getBwriter() {
        return bwriter;
    }
    public void setBwriter(String bwriter) {
        this.bwriter = bwriter;
    }
    public String getBcontent() {
        return bcontent;
    }
    public void setBcontent(String bcontent) {
        this.bcontent = bcontent;
    }
    public String getBfile() {
        return bfile;
    }
    public void setBfile(String bfile) {
        this.bfile = bfile;
    }
    public String getBdate() {
        return bdate;
    }
    public void setBdate(String bdate) {
        this.bdate = bdate;
    }

    public FboardDTO(String bno, String btitle, String bwriter, String bcontent, String bfile, String bdate) {
        super();
        this.bno = bno;
        this.btitle = btitle;
        this.bwriter = bwriter;
        this.bcontent = bcontent;
        this.bfile = bfile;
        this.bdate = bdate;
    }
}

```

#### ▼ getList 메소드

```

public static ArrayList<FboardDTO> getList()
    throws NamingException, SQLException {

```

```

String sql = "SELECT * FROM fboard";

Connection conn = ConnectionPool.get();

PreparedStatement pstmt = conn.prepareStatement(sql);

ResultSet rs = pstmt.executeQuery();

ArrayList<FboardDTO> boards = new ArrayList<FboardDTO>();

while(rs.next()) {
    boards.add(new FboardDTO(rs.getString(1),
        rs.getString(2),
        rs.getString(3),
        rs.getString(4),
        rs.getString(5),
        rs.getString(6)));
}

return boards;
}

```

## ■ Hour 4

### DB에 올라간 사진 표시

#### ▼ FboardList.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ page import="jdbc.*, java.util.*" %>

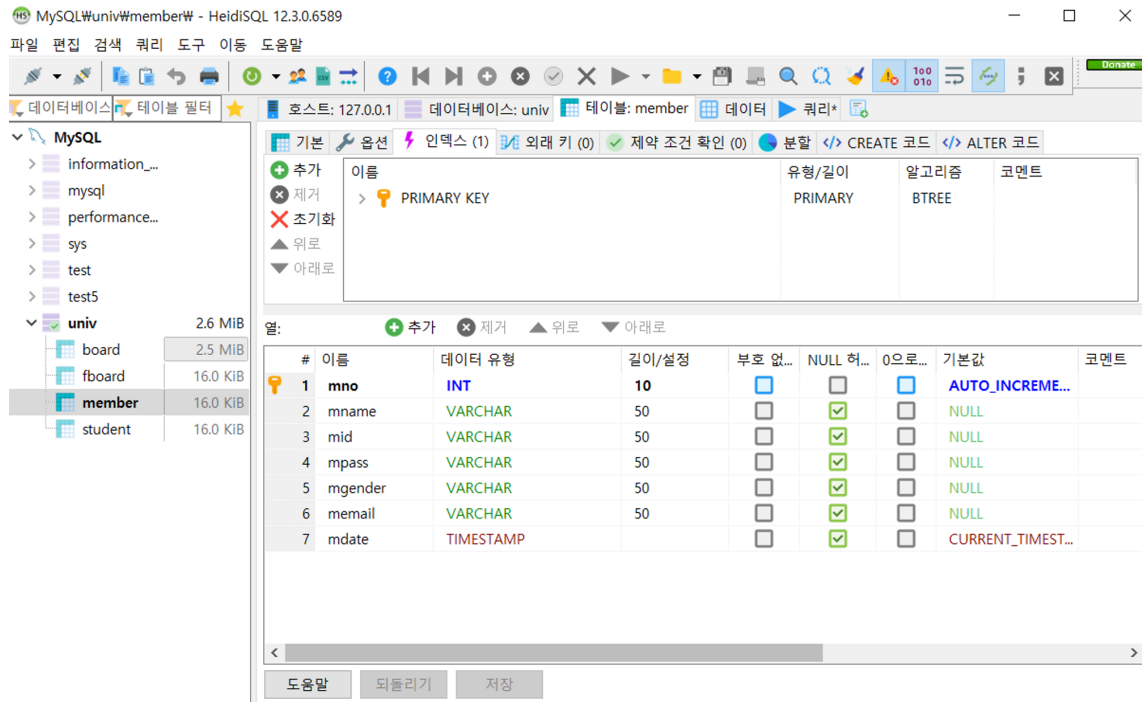
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>글 목록</title>
</head>
<body>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLT
<table class="table table-hover">
    <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">title</th>
            <th scope="col">writer</th>
            <th scope="col">content</th>
            <th scope="col">file</th>
            <th scope="col">date</th>
        </tr>
    </thead>
    <tbody>
<%
    ArrayList<FboardDTO> boards = FboardDAO.getList();
    for(FboardDTO fboard : boards){

        String img = fboard.getBfile();
        String imgstr = "";
        if(img != null) {
            imgstr = "<img src='images/" + img + "' width=200>";
        }
%>
        <tr>
            <td><%=fboard.getBno() %></td>
            <td><%=fboard.getBtitle() %></td>
            <td><%=fboard.getBwriter() %></td>
            <td><%=fboard.getBcontent() %></td>
            <td><%=imgstr %></td>
            <td><%=fboard.getBdate() %></td>
        </tr>
    <% }%>
    </tbody>
</table>

</body>
</html>

```





#### ▼ MemberDTO

```
package jdbc;

public class MemberDTO {
    private int mno;
    private String mname;
    private String mid;
    private String mpass;
    private String mgender;
    private String memail;
    private String mdate;

    public int getMno() {
        return mno;
    }
    public String getMname() {
        return mname;
    }
    public String getMid() {
        return mid;
    }
    public String getMpass() {
        return mpass;
    }
    public String getMgender() {
        return mgender;
    }
    public String getMemail() {
        return memail;
    }
    public String getMdate() {
        return mdate;
    }

    public MemberDTO(int mno, String mname, String mid, String mpass, String mgender, String memail, String mdate) {
        super();
        this.mno = mno;
        this.mname = mname;
        this.mid = mid;
        this.mpass = mpass;
        this.mgender = mgender;
        this.memail = memail;
        this.mdate = mdate;
    }
}
```

#### ▼ MemberDAO



```

package jdbc;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.naming.NamingException;

import util.ConnectionPool;

public class MemberDAO {
    public static int insert(String mname, String mid, String mpass, String mgender, String memail ) throws SQLException, NamingException {
        // CRUD
        String sql = "INSERT INTO member (mname, mid, mpass, mgender, memail) VALUES(?,?,?,?,?)";

        Connection conn = ConnectionPool.get(); // 커넥션 사용

        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, mname);
        pstmt.setString(2, mid);
        pstmt.setString(3, mpass);
        pstmt.setString(4, mgender);
        pstmt.setString(5, memail);

        int result = pstmt.executeUpdate();
        // sql 구문 실행 성공 여부가 1과 0으로 돌아온다
        return result;
    }

    public static int login(String mid, String mpass) throws SQLException, NamingException {
        String sql = "select mid, mpass from member where mid=?";
        Connection conn = ConnectionPool.get(); // 커넥션 사용

        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, mid);
        ResultSet rs = pstmt.executeQuery();
        if(!rs.next()) {
            return 1; // DB에 회원정보가 없는 경우
        } else if (!mpass.equals(rs.getString(2))) {
            return 2; // 비밀번호만 틀린 경우
        } else {
            return 0; // id와 비밀번호 모두 일치
        }
    }
}

```

#### ▼ signupCheck.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ page import="jdbc.*" %>
<%
String mid = request.getParameter("mid");
String mname = request.getParameter("mname");
String mpass = request.getParameter("mpass");
String mgender = request.getParameter("mgender");
String memail = request.getParameter("memail");

MemberDAO.insert(mname, mid, mpass, mgender, memail);

if(MemberDAO.insert(mname, mid, mpass, mgender, memail)==1){
    response.sendRedirect("login.jsp");
} else {
    out.print("fail");
}
%>

```

#### ▼ signupForm.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">

```

```

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">

<title>SB Admin 2 - Register</title>

<!-- Custom fonts for this template-->
<link href="vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">
<link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">

<!-- Custom styles for this template-->
<link href="css/sb-admin-2.min.css" rel="stylesheet">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384"

</head>

<body class="bg-gradient-primary">

    <div class="container">
        <form action="signupCheck.jsp" method="post">
            <div class="card o-hidden border-0 shadow-lg my-5">
                <div class="card-body p-0">
                    <!-- Nested Row within Card Body -->
                    <div class="row">
                        <div class="col-lg-5 d-none d-lg-block bg-register-image"></div>
                        <div class="col-lg-7">
                            <div class="p-5">
                                <div class="text-center">
                                    <h1 class="h4 text-gray-900 mb-4">회원 가입</h1>
                                </div>
                                <form class="user">
                                    <div class="form-group row">
                                        <div class="col-sm-6 mb-3 mb-sm-0">
                                            <input type="text" name="mname" class="form-control form-control-user" id="exampleFirs"
                                                placeholder="이름">
                                        </div>
                                        <div class="col-sm-6">
                                            <input type="text" name="mid" class="form-control form-control-user" id="exampleLastNa"
                                                placeholder="아이디">
                                        </div>
                                    </div>
                                    <div class="form-group row">
                                        <div class="col-sm-6 mb-3 mb-sm-0">
                                            <input type="email" name="memail" class="form-control form-control-user" id="exampleInputE"
                                                placeholder="이메일">
                                        </div>
                                        <div class="col-sm-6">
                                            <select class="form-select" name="mgender" aria-label="Default select example">
                                                <option selected>성별 선택</option>
                                                <option value="male">남성</option>
                                                <option value="female">여성</option>
                                            </select>
                                        </div>
                                    </div>
                                    <div class="form-group row">
                                        <div class="col-sm-6 mb-3 mb-sm-0">
                                            <input type="password" name="mpass" class="form-control form-control-user"
                                                id="exampleInputPassword" placeholder="암호">
                                        </div>
                                        <div class="col-sm-6">
                                            <input type="password" class="form-control form-control-user"
                                                id="exampleRepeatPassword" placeholder="암호 확인">
                                        </div>
                                    </div>
                                    <div class="d-flex justify-content-center">
                                        <button type="submit" class="btn btn-primary">등록</button>
                                    </div>
                                </form>
                                <hr>
                                <a href="index.html" class="btn btn-google btn-user btn-block">
                                    <i class="fab fa-google fa-fw"></i> Register with Google
                                </a>
                                <a href="index.html" class="btn btn-facebook btn-user btn-block">
                                    <i class="fab fa-facebook-f fa-fw"></i> Register with Facebook
                                </a>
                                </form>
                                <hr>
                                <div class="text-center">
                                    <a class="small" href="forgot-password.html">Forgot Password?</a>
                                </div>
                                <div class="text-center">

```



```

border: solid rgba(0, 0, 0, .15);
border-width: 1px 0;
box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em rgba(0, 0, 0, .15);
}

.b-example-vr {
  flex-shrink: 0;
  width: 1.5rem;
  height: 100vh;
}

.bi {
  vertical-align: -.125em;
  fill: currentColor;
}

.nav-scroller {
  position: relative;
  z-index: 2;
  height: 2.75rem;
  overflow-y: hidden;
}

.nav-scroller .nav {
  display: flex;
  flex-wrap: nowrap;
  padding-bottom: 1rem;
  margin-top: -1px;
  overflow-x: auto;
  text-align: center;
  white-space: nowrap;
  -webkit-overflow-scrolling: touch;
}
}
</style>

<!-- Custom styles for this template -->
<link href="sign-in.css" rel="stylesheet">
</head>
<body class="text-center">

<div class="container col-md-4">
  <form action="loginCheck.jsp" method="post">
    <h1 class="h3 mb-3 fw-normal">Please sign in</h1>

    <div class="form-floating">
      <input type="text" class="form-control" id="floatingInput" placeholder="ID" name="mid">
      <label for="floatingInput">ID</label>
    </div>
    <div class="form-floating">
      <input type="password" class="form-control" id="floatingPassword" placeholder="Password" name="mpass">
      <label for="floatingPassword">Password</label>
    </div>

    <div class="checkbox mb-3">
      <label>
        <input type="checkbox" value="remember-me"> Remember me
      </label>
    </div>
    <button class="w-100 btn btn-lg btn-primary" type="submit">Sign in</button>
    <p class="mt-5 mb-3 text-muted">&copy; 2017-2022</p>
  </form>
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPFDkM"
  </body>
</html>

```

#### ▼ loginCheck.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ page import="jdbc.*"%>
<%

String mid = request.getParameter("mid");
String mpass = request.getParameter("mpass");

int result = MemberDAO.login(mid, mpass);

switch (result) {
case 1:
// DB에 정보가 없는 경우 : 회원가입 페이지로 이동

```

```

response.sendRedirect("signupForm2.jsp");

    return;
case 2:
// 비번이 틀린 경우 : 다시 로그인 페이지로 이동
response.sendRedirect("login.jsp");

    return;
case 0:
// 로그인 성공 케이스 : 세션 생성 후 메인으로 이동
session.setAttribute("mid", mid);
response.sendRedirect("main.jsp");
    return;
}
%>

```

## ▼ MemberDAO

```

package jdbc;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.naming.NamingException;

import util.ConnectionPool;

public class MemberDAO {
    public static int login(String mid, String mpass) throws SQLException, NamingException {
        String sql = "select mid, mpass from member where mid=?";
        Connection conn = ConnectionPool.get(); // 커넥션 사용

        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, mid);
        ResultSet rs = pstmt.executeQuery();
        if(!rs.next()) {
            return 1; // DB에 회원정보가 없는 경우
        } else if (!mpass.equals(rs.getString(2))) {
            return 2; // 비밀번호만 틀린 경우
        } else {
            return 0; // id와 비밀번호 모두 일치
        }
    }
}

```

## ▼ MemberDTO

```

package jdbc;

public class MemberDTO {
    private int mno;
    private String mname;
    private String mid;
    private String mpass;
    private String mgender;
    private String memail;
    private String mdate;

    public int getMno() {
        return mno;
    }
    public String getMname() {
        return mname;
    }
    public String getMid() {
        return mid;
    }
    public String getMpass() {
        return mpass;
    }
    public String getMgender() {
        return mgender;
    }
    public String getMemail() {
        return memail;
    }
    public String getMdate() {
        return mdate;
    }
}

```

```

public MemberDTO(int mno, String mname, String mid, String mpass, String mgender, String memail, String mdate) {
    super();
    this.mno = mno;
    this.mname = mname;
    this.mid = mid;
    this.mpass = mpass;
    this.mgender = mgender;
    this.memail = memail;
    this.mdate = mdate;
}
}

```

## ▼ DB

MySQL#univ#member# - HeidiSQL 12.3.0.6589

파일 편집 검색 쿼리 도구 이동 도움말

호스트: 127.0.0.1 데이터베이스: univ 테이블: member 데이터 쿼리\*

MySQL

- information\_...
- mysql
- performance...
- sys
- test
- test5
- univ
  - board 2.6 MIB
  - fboard 2.5 MIB
  - member 16.0 KIB
  - student 16.0 KIB

기본 옵션 인덱스 (1) 외래 키 (0) 제약 조건 확인 (0) 분할 </> CREATE 코드 </> ALTER 코드

추가 제거 초기화 위로 아래로

이름 유형/길이 알고리즘 코멘트

> PRIMARY KEY PRIMARY BTREE

열: + 추가 ✕ 제거 ▲ 위로 ▼ 아래로

#	이름	데이터 유형	길이/설정	부호 없...	NULL 허...	0으로...	기본값	코멘트
1	mno	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...	
2	mname	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
3	mid	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
4	mpass	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
5	mgender	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
6	memail	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	
7	mdate	TIMESTAMP		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	CURRENT_TIMEST...	

< >

도움말 되돌리기 저장