

230117 File Upload & Login

BOOTSTRAP

- 사이트 전체 구성을 쉽고 빠르게 템플릿으로 이용하는 방법

link : <https://startbootstrap.com/>

적용예시 : <https://startbootstrap.com/theme/sb-admin-2>

File Upload 구현

사진 전송을 위해서는 multi/part 옵션을 설정해야 하는데, 문제는 기존 방식의 'key:value' 형태로 데이터를 넘기는 것이 아니라, 전체 데이터를 하나의 덩어리로 넘기게 된다.

따라서 게시물의 제목, 내용, 파일이름, 파일내용(사진은 두 개로 분해된다.)으로 각각 추출해서 DB에 넣어주어야 한다. (사진 자체는 DB에 들어가지 않고, DB에는 이름만 들어가고 사진 자체는 서버의 특정 위치로 전송된다.)

(Summernote의 경우 사진이 code로 DB에 들어감)

▼ 파일 입력 form 예시

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css" integrity="sha384-x0olHFLHc0" />
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<div class="container">
    <form action="fboardInsert.jsp" method="POST" enctype="multipart/form-data">
        <div class="form-group">
            <label for="exampleFormControlInput1">Title</label>
            <input name="fbtitle" type="text" class="form-control" id="exampleFormControlInput1">
        </div>
        <div class="form-group">
            <label for="exampleFormControlTextarea1">Content</label>
            <textarea name="fbcontent" class="form-control" id="exampleFormControlTextarea1" rows="3"></textarea>
        </div>
        <div class="form-group">
            <label for="inputfile">파일</label>
            <input name="fbfile" type="file" class="form-control" id="inputfile" multiple>
        </div>
        <div class="text-right">
            <button type="submit" class="btn btn-primary">글 작성</button>
        </div>
    </form>
</div>
</body>
</html>
```

파일 전송 시 enctype="multipart/form-data" 로 데이터를 전송해야 하는데, 해당 파일 데이터를 request 객체를 통해 접근하게 되면 데이터를 추출할 수 없다.

null
null
null

-----7e731e2330a06Content-Disposition: form-data; name="fbtitle"teafdsa-----
7e731e2330a06Content-Disposition: form-data; name="fbcontent" rteafdsagfds-----
7e731e2330a06Content-Disposition: form-data; name="fbfile";
filename="C:\Users\user\Pictures\AWSClass\1203.png"Content-Type: image/pngPNGIHDRU⁸
[sRGB⁸gAMA⁸a⁸
pHYs⁸d⁸o⁸d⁸IDATx⁸r⁸s⁸#⁸=3w⁸tW⁸L&5@h⁸Zk⁸ZkJM⁸p⁸0⁸Z⁸?R⁸H⁸"E⁸)R⁸H⁸"E⁸)R⁸H⁸@⁸M⁸)R⁸H⁸"E⁸)R⁸H⁸"⁸w⁸R⁸&E⁸)R⁸H⁸"E⁸)R⁸H⁸,⁸Z⁸q⁸"E⁸)R⁸H⁸"E⁸)R⁸H⁸0w⁸Z⁸9⁸?⁸=⁸z⁸tv⁸o⁸?⁸-⁸r⁸o⁸7q⁸]]
S⁸d⁸9x⁸H⁸^⁸r⁸3⁸t⁸o⁸鬱⁸g/^?}
pt⁸g[G⁸k⁸+⁸ik⁸+k⁸+S⁸3s⁸c⁸C⁸=⁸]
ÿ⁸ń⁸H8⁸^⁸q⁸\N⁸9⁸f⁸d⁸V⁸d⁸X4f⁸1TL⁸Q⁸2⁸Fj⁸H⁸j⁸~⁸Щ⁸Q⁸P⁸Q⁸F⁸%⁸,⁸P⁸J1⁸["⁸\DEQ))⁸O⁸p2^K2Rn⁸S⁸Y⁸i⁸D⁸F⁸
(b⁸f⁸π⁸^⁸PåA8⁸j6⁸E⁸d⁸*⁸&⁸A⁸h⁸RjK1⁸o⁸h⁸E⁸R]
QS⁸YW⁸P⁸P⁸J⁸T⁸((⁸F⁸O⁸*⁸@⁸))
Zhu⁸+⁸a⁸5mc%45⁸vm⁸w⁸{⁸y⁸'⁸w⁸/P⁸O⁸矛⁸
C⁸L⁸d⁸|⁸r'j⁸/{⁸cWWA⁸c⁸G>⁸z⁸1⁸g⁸Wg⁸OD⁸
G⁸{⁸[⁸3#3SC⁸c⁸}#⁸]}}=⁸p[[⁸]
w⁸?⁸from(h)æxcwtkb⁸ha7 O/⁸O{⁸r7⁸k7⁸ZP⁸"/⁸]

// 넘어오는 데이터 뒹어리를 시각적으로 확인할 수 있음
<%
String str = "";
String line= "";

InputStreamReader isr = new InputStreamReader(request.getInputStream(), "UTF-8");
BufferedReader br = new BufferedReader(isr);
while((line = br.readLine()) != null){
str += line;
}

out.print(str);

>

- multipart 데이터로 전송한 내역을 확인하면 위와 같이 덩어리로 전달되는 것을 확인할 수 있다.

따라서, 전체 데이터 덩어리에서 추출해야 하는데 추출을 도와주는 라이브러리를 설치 후 조금 쉽게 추출할 수 있다.

- 2개의 라이브러리 설치 (추출)

아래 zip 파일 설치

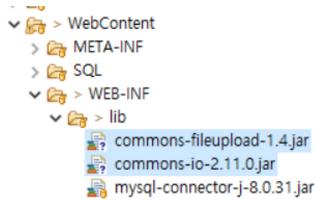
- commons-fileupload-1.4-bin.zip
 - commons-io-2.11.0-bin.zip

Apache Commons - Apache Commons

Apache Commons is an Apache project focused on all aspects of reusable Java components. The Apache Commons project is composed of three parts: You may also read our charter, which spells out the goals of the project in greater detail. The Apache Commons source code repositories are writable for all ASF committers.

 <https://commons.apache.org/>

WebContent > WEB-INF > lib 폴더 안에 jar파일 복사



사용할 JSP 파일에서 import

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

- 추출에 필요한 기본 메서드 정리

isFormField() : 요소가 key-value 쌍으로 구성된 필드를 찾아낸다.

getFieldName() : key값 속성을 추출

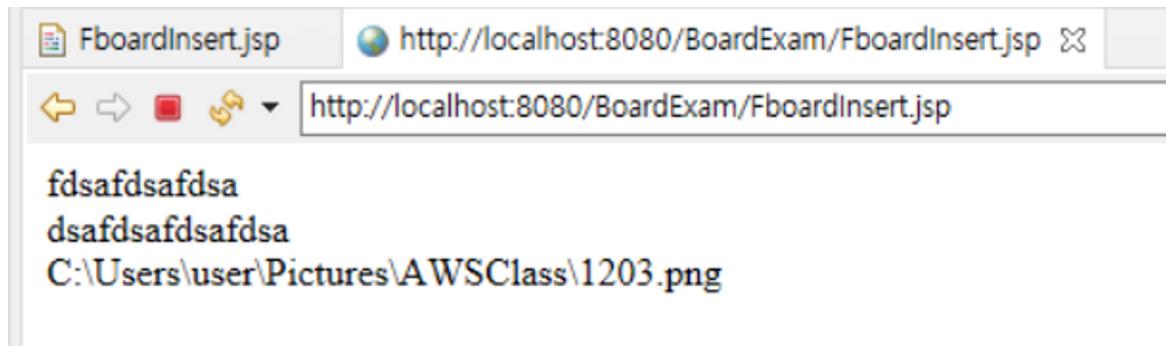
getString() : value값을 추출

getName() : 사진(파일) 이름 추출

get() : 사진 파일 자체를 추출

▼ 추출 결과 및 코드

추출 내용 (title, content, filename 표시)



코드

```
<%@ 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.*" %>

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

String fbttitle = null; //게시물 제목
String fbcontent = null; //게시물 내용
String ffbfile = null; //사진 이름
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 벨류값 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();
            }
        }
    }

%>

<%=fbtitle %><br>
<%=fbcontent %><br>
<%=fbfile %><br>

```

- 파일 서버에 저장하기 위한 방법

- 데이터베이스 내부에 파일을 직접 넣는 방법(summernote가 사용하고 있는 방법)
- 서버의 특정 위치에 파일을 그대로 저장하고, DB에는 그 이름과 경로만 저장하는 방법(중요)

일반적으로 두 방법 중 후자를 많이 사용하게 된다. 그 이유는 DB 내부에 파일을 저장하게 되면, 이미지의 크기에 비례해서 테이블의 크기 자체가 기하급수적으로 커지게 되어 그에 따라 관리 비용이 크게 증가하게 된다. 따라서 후자가 실무에서는 많이 사용됨

▼ 사진 파일을 서버의 특정 위치에 적용하는 방법

- fboard 테이블 구조

Table: fBoard	
Columns:	
fbno	int AI PK
fbtitle	varchar(50)
fbwriter	varchar(50)
fbcontent	longtext
fbdate	timestamp
fbfile	longtext

- util > FileUtils.java : 파일 전송 용 클래스 생성 (서버의 특정 위치에 파일 업로드 처리)

```

package util;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;

public class FileUtils {
    public static void saveImage(String root, String fname, byte[] data) throws IOException {
        root += "/images";
        File f = new File(root);
        if(!f.exists()) f.mkdir(); // images 폴더가 없으면 새로 만들

        f = new File(root + "/" + fname);
        FileOutputStream out = new FileOutputStream(f); // filestream 객체 사용
        out.write(data); // 사진 저장
        out.close();
    }
}

```

- fBoardDTO 생성

```

package jdbc;

public class FboardDTO {
    // 입력 받는 데이터는 모두 문자로 처리되기 때문에 DTO는 String 으로 처리한다.
    private String fbno;
    private String fbttitle;
    private String fbwriter;
    private String fbcontent;
    private String fbfile;
    private String fbdate;

    public FboardDTO(String fbno, String fbttitle, String fbwriter, String fbcontent, String fbfile, String fbdate) {
        super();
        this.fbno = fbno;
        this.fbttitle = fbttitle;
        this.fbwriter = fbwriter;
        this.fbcontent = fbcontent;
        this.fbfile = fbfile;
        this.fbdate = fbdate;
    }
    public String getFbno() {
        return fbno;
    }
    public String getFbttitle() {
        return fbttitle;
    }
    public String getFBwriter() {
        return fbwriter;
    }
    public String getFBcontent() {
        return fbcontent;
    }
    public String getFBfile() {
        return fbfile;
    }
    public String getFBdate() {
        return fbdate;
    }
}

```

- fBoardDAO 생성(insert, select)

```

package jdbc;

import java.sql.*;
import java.util.ArrayList;

import javax.naming.NamingException;

import util.ConnectionPool;

public class FBoardDAO {
    public static int insert(String title, String writer, String content, String file) throws NamingException, SQLException {
        String sql = "INSERT INTO fboard(fbttitle, fbwriter, fbcontent, fbfile) VALUES(?, ?, ?, ?)";

        Connection conn = ConnectionPool.get();

        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1, title);
        pstmt.setString(2, writer);
        pstmt.setString(3, content);
        pstmt.setString(4, file);

        int result = pstmt.executeUpdate();
        return result;
    }

    public static ArrayList<FboardDTO> getList() throws SQLException, NamingException{
        String sql = "select fbno, fbttitle, fbwriter, fbcontent, fbfile, DATE_FORMAT(fbdate, '%Y-%m-%d %H:%i:%s') as fbdate from fboar

```

```

        rs.getString(3),
        rs.getString(4),
        rs.getString(5),
        rs.getString(6)));
    }

    return boards;
}

```

- 파일 insert를 처리하는 JSP파일

위의 추출 메서드를 사용하여 파일과 key-value 데이터를 구분하여 처리한다.

```

<%@ 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 fbttitle = null; //게시물 제목
String fbcontent = null; //게시물 내용
String fbfile = null; //사진 이름
String fbwriter = "someone"; //작성자는 임시로 설정, 실제로는 세션에서 읽어와야 함
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()) { // key-value값으로 구분되는 title, content 처리
        String value = item.getString("utf-8");
        if (name.equals("fbtitle")) fbttitle = value;
        else if (name.equals("fbcontent")) fbcontent = value;
    } else {
        if (name.equals("fbfile")) {
            fbfile = item.getName(); // 사진 파일 이름 추출
            fbu = item.get(); // 사진 파일 추출
            String root = application.getRealPath(java.io.File.separator); // 해당 서버의 실제 경로를 가져옴
            FileUtil.saveImage(root, fbfile, fbu);
        }
    }
}

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

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

%>

```

- 이미지 파일을 표시하는 리스트 JSP 작성

```

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

<!DOCTYPE html>
<html>
<head>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq7"%
<meta charset="UTF-8">
<title>Insert title here</title>

```

```

</head>
<body>
<% ArrayList<FboardDTO> boardList = FBoardDAO.getList(); %>
<div class="container">
    <table class="table table-hover">
        <thead>
            <tr>
                <th scope="col">글번호</th>
                <th scope="col">제목</th>
                <th scope="col">작성자</th>
                <th scope="col">사진</th>
                <th scope="col">작성시간</th>
            </tr>
        <tbody>
<% for(FboardDTO board : boardList){>
            String img = board.getFbfile(); //파일 이름 변수로 저장
            String imgstr = "";
            if( img != null ){ //사진이 있으면
                imgstr=<img src='images/' + img + "' width=200>";
            }
<%>
            <tr>
                <th scope="row"><%=board.getFbno() %></th>
                <td><%=board.getFbtitle() %></td>
                <td><%=board.getFbwriter() %></td>
                <td><%=imgstr %></td>
                <td><%=board.getFbdate() %></td>
            </tr>
<% } %>
        </tbody>
    </table>
</div>
</body>
</html>

```

LOGIN 처리

기본적으로는 index.html이 호출되지만, html에서는 jsp문법을 사용할 수가 없다. 따라서 index.html이 호출되면 자동으로 main.jsp를 호출하여 사용하면 편리하다

▼ index 페이지 접근 시 main.jsp 호출

- index.html

```

<script>
window.location.href="main.jsp"; // index.html 접근하였으나 main.jsp로 이동
</script>

```

- main.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
       pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
    main페이지 입니다.
</body>
</html>

```

▼ 회원 가입 처리

- 처리 방식

db - table - (dto)- dao - signupCheck.jsp - signupForm.jsp

- member 테이블 구조

Table: member	
Columns:	
mno	int AI PK
mname	varchar(50)
mid	varchar(50)
mpass	varchar(50)
mgender	varchar(50)
memail	varchar(50)
mdate	timestamp

Query Completed

- memberDTO 생성

```
package jdbc;

public class memberDTO {
    private String mno;
    private String mname;
    private String mid;
    private String mpass;
    private String mgender;
    private String memail;
    private String mdate;
    public memberDTO(String 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;
    }
    public String 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;
    }
}
```

- MemberDAO 생성(insert 처리)

```
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 NamingException, SQLException {
        String sql = "INSERT INTO member(mname, mid, mpass, mgender, memail) VALUES(?,?,?,?,?)";
        Connection conn = ConnectionPool.get();
        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();
        return result;
    }
}

```

- 회원가입 signupForm.jsp (부트스트랩 사용)

```

<%@ 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">
    <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="exampleFirstName"
                                           placeholder="이름">
                                </div>
                                <div class="col-sm-6">
                                    <input type="text" name="mid" class="form-control form-control-user" id="exampleLastName"
                                           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="exampleInputEmail"
                                           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>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </div>
</form>
</div>

```

- 회원가입을 처리하는 signupCheck.jsp

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

<%
String mname = request.getParameter("mname");
String mid = request.getParameter("mid");
String mpass = request.getParameter("mpass");
String mgender = request.getParameter("mgender");
String memail = request.getParameter("memail");

int result = memberDAO.insert(mname, mid, mpass, mgender, memail);

if (result == 1) {
    out.print("등록 성공");
} else {
    out.print("등록 실패");
}
%>
```

▼ 로그인 처리

- 처리 방식

db - table - dao (login Method) - loginCheck.jsp - loginForm.jsp 회원 로그인

(반환되는 결과의 경우의 수가 3개이다.)

- 입력한 아이디와 비번이 디비의 정보와 모두 일치하는 경우 → 메인화면
- 입력한 아이디는 있지만 비번이 다른 경우 → 다시 로그인 창
- 입력한 아이디와 비번 모두 없는 경우 → 회원 가입

- MemberDTO 생성 (회원가입과 동일)

- MemberDAO login 메서드 추가

```
public static int login(String mid, String mpass) throws NamingException, SQLException {  
    String sql = "SELECT mid, mpass FROM MEMBER WHERE mid=?";  
    Connection conn = ConnectionPool.get();  
    PreparedStatement pstmt = conn.prepareStatement(sql);  
    pstmt.setString(1, mid);  
    ResultSet rs = pstmt.executeQuery();  
  
    if(!rs.next()) return 1; // DB에 회원 정보가 없는 경우  
    if(!mpass.equals(rs.getString(2))) return 2; // 비밀번호만 틀린 경우  
    return 0; // 아이디와 비밀번호 모두 일치하는 경우  
}
```

- 로그인 login.jsp 생성 (부트스트랩 사용)

```
<%@ 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">  
  
        <!-- Outer Row -->  
        <div class="row justify-content-center">  
  
            <div class="col-xl-10 col-lg-12 col-md-9">  
  
                <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-6 d-none d-lg-block bg-login-image"></div>  
                            <div class="col-lg-6">  
                                <div class="p-5">
```

```

        <div class="text-center">
            <h1 class="h4 text-gray-900 mb-4">Welcome Back!</h1>
        </div>
        <form class="user" action="loginCheck.jsp">
            <div class="form-group">
                <input name="mid" type="text" class="form-control"
                    id="exampleInputEmail" aria-describedby="emailHelp"
                    placeholder="Enter ID...">
            </div>
            <div class="form-group">
                <input name="mpass" type="password" class="form-control"
                    id="exampleInputPassword" placeholder="Password">
            </div>
            <div class="form-group">
                <div class="custom-control custom-checkbox small">
                    <input type="checkbox" class="custom-control-input" id="customCheck">
                    <label class="custom-control-label" for="customCheck">Remember
                        Me</label>
                </div>
            </div>
            <div class="form-group">
                <button type="submit" class="form-control btn btn-primary">로그인</button>
            </div>
            <hr>
            <a href="index.html" class="btn btn-google btn-user btn-block">
                <i class="fab fa-google fa-fw"></i> Login with Google
            </a>
            <a href="index.html" class="btn btn-facebook btn-user btn-block">
                <i class="fab fa-facebook-f fa-fw"></i> Login with Facebook
            </a>
        </form>
        <hr>
        <div class="text-center">
            <a class="small" href="forgot-password.html">Forgot Password?</a>
        </div>
        <div class="text-center">
            <a class="small" href="register.html">Create an Account!</a>
        </div>
        </div>
    </div>

```

<!-- Bootstrap core JavaScript-->

```

<script src="vendor/jquery/jquery.min.js"></script>
<script src="vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- Core plugin JavaScript-->
<script src="vendor/jquery-easing/jquery.easing.min.js"></script>

<!-- Custom scripts for all pages-->
<script src="js/sb-admin-2.min.js"></script>

```

```

</body>
</html>

```

- 로그인을 처리하는 loginCheck.jsp 생성

```

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

```

```

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

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

    if(result==0){ // 로그인 성공 케이스, 세션 생성 후 메인으로 이동
        session.setAttribute("mid", mid); // 세션 생성
        response.sendRedirect("main.jsp");
    } else if (result==1){ // DB에 정보가 없어 회원 가입 페이지로 이동
        response.sendRedirect("signupForm.jsp");
    } else { // 비밀번호가 틀린 경우 로그인 페이지로 이동
        response.sendRedirect("login.jsp");
    }
}

```

%>