230118 KAKAO LOGIN

카카오 로그인

로그인 프로세스 전체를 카카오나 구글 등에 완전히 위임하게 되면 (자체 디비에 고객 정보가 전혀 없이) 마케팅 등이 어렵게 된다. 따라서 처음 사이트에 방문시 카카오 로그인을 했을때 그때 한번 회원 가입을 할수 있게 하면 추후 로그인만 카카오를 통해서 허용하고 따라서 고객의 정보도 직접 보관할수 있어 다양한 프로모션 등의 고객 관리도 할수 있다.

- 1. 단지 고객 디비 관리 없이 카카오 로그인 만으로 회원인증
- 2. 로그인은 카카오로 인증을 할수 있지만 최초 접속시 회원 가입을 한번 진행하여 회원 관리/인증

카카오 로그인 → 해당 회원의 정보가 자체 디비에 존재하는지 확인 → 없으면 회원 가입 진행 → 있으면 로그인 성공(세션 만들기)

자체 로그인 페이지(아이디 비번) \rightarrow loginCheck.jsp \rightarrow DAO \rightarrow DB Table(로그인 성공/실패) 자체 로그인 페이지(카카오 호출) \rightarrow 카카오 로그인 페이지(아이디, 비번) \rightarrow 로그인 성공/실패

카카오 사이트 설정

▼ 카카오 개발자로 들어가서 설정

카카오 개발자 등록

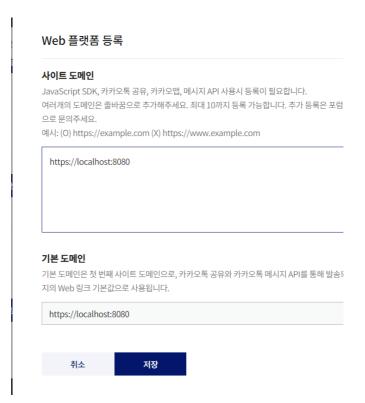
애플리케이션으로 들어 가면 최초 개발자 등록(회원가입)을 진행하게 된다.



JavaScript 7 1d944d4a9cb22483964d7e130a9b4c2a

플랫폼

설정된 플랫폼 정보가 없습니다. 플랫폼 설정하기



활성화 클릭



OpenID Connect 활성화 설정

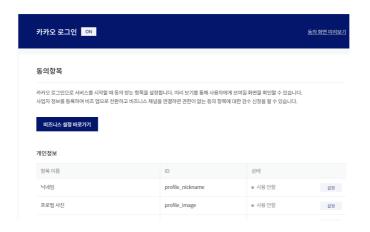


카카오 로그인의 확장 기능인 OpenID Connect를 활성화합니다. 이 설정을 활성화하면 카카오 로그인 시 사용자 인증 정보가 담긴 ID 토큰을 액세스

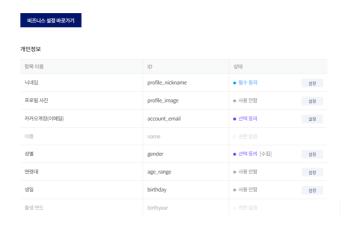
Redirect URI

Redirect URI 카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다. 여러개의 URI를 줄바꿈으로 추가해주세요. (최대 10개) REST API로 개발하는 경우 필수로 설정해야 합니다. 예시: (O) https://example.com/oauth (X) https://www.example.com/oauth http://localhost:8080

카카오 로그인에 성공했을 때 넘겨 받을 데이터 설정



로그인 성공시 닉네임, 이메일, 성별 데이터를 넘겨 받게 된다.



닉네임, 카카오계정(이메일), 성별 [설정]

카카오 사이트 설정 끝.

Kakao Login 해보기~!

자체 로그인 페이지(카카오 호출) → 카카오 로그인 페이지(아이디, 비번) → 로그인 성공/실패

▼ 로그인 페이지(카카오 호출)

loginKakao.jsp 안에 아래 코드 적절하게 배치한다

```
<%@ page language="java" contentType="text/html; charset=UTF-8"</pre>
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<a id="custom-login-btn" href="javascript:kakaoLogin()">
<img src="kakao.png">
</a>
=>요기는 카카오로그인 버튼!
<script src="https://developers.kakao.com/sdk/is/kakao.is"></script>
 .
//발급 받은 키
 Kakao.init("1d944d4a9cb22483964d7e130a9b4c2a");
  function kakaoLogin() {
   window.Kakao.Auth.login({
     // 개발자 사이트에서 체크한 항목들과 반드시 일치해야 한다.
      scope:'profile_nickname, account_email, gender',
     success: function (authObj) {window.Kakao.API.request({url:'/v2/user/me',
       success:res => {const kakaoAccount = res.kakao_account;
          location.href='loginCheckKakao.jsp?mname=' + kakaoAccount.profile.nickname
             + '&memail=' + kakaoAccount.email
+ '&mgender=' + kakaoAccount.gender;}
     });
   });
</script>
</body>
</html>
```

▼ memberDAO(exist 메서드)

```
public static int exist(String memail) throws NamingException, SQLException {

//C R U D

String sql = "SELECT memail FROM member WHERE memail=?";

//커넥션 풀 사용

Connection conn = ConnectionPool.get();

PreparedStatement pstmt = conn.prepareStatement(sql);

pstmt.setString(1, memail);

ResultSet rs = pstmt.executeQuery();

if (!rs.next()) return 1; //디비에 회원 정보가 전혀 없는 경우
```

```
return 0; // 디비에 존재하는 경우
}
```

▼ loginCheckKakao.jsp

우리 DB에 해당 회원이 존재하는지 여부 확인 exist 메서드로 확인 후 존재하면0, 세션처리(로그인상태) 디비에 존재하지 않으면 회원가입 페이지로 이동 가입 유도

```
<<pre><%@ page import="jdbc.*"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"</pre>
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
 request.setCharacterEncoding("utf-8");
 String mname = request.getParameter("mname");
 String memail = request.getParameter("memail");
 String mgender = request.getParameter("mgender");
//우리 DB에 해당 회원이 존재하는지 여부 확인 exist 메서드로 확인 후
//존재하면0, 세션처리(로그인상태)
//디비에 존재하지 않으면 회원가입 페이지로 이동 가입 유도
int result = memberDAO.exist(memail);
if (result == 0) {
 out.print("로그인 성공");
} else {
 response.send Redirect ("signupForm.jsp?mname=" + java.net.URLEncoder.encode(mname) + "\&memail="+memail+"\&mgender="+mgender); \\
</body>
</html>
```

▼ signupForm.jsp

디비에 존재하지 않아서 회원가입 페이지로 넘어오게되면

카카오로 로그인 했을때 가져올수있는

이름, 이메일, 성별 받아서 readonly 처리 해주고 회원가입 할수있게 해줌

```
<%@ page language="java" contentType="text/html; charset=UTF-8"</pre>
   pageEncoding="UTF-8"%>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
 String mname = request.getParameter("mname");
 String memail = request.getParameter("memail");
 String mgender = request.getParameter("mgender");
<div class="jumbotron">
   <h1 class="display-4">회원 가입</h1></div>
</div>
<div class="container">
 <form action="signupCheck.jsp" id="joinFrm" class="form-horizontal" method="get">
   <div class="form-group row">
    <label class="col-sm-2">0|0|E|</label>
    <div class="col-sm-3">
```

```
<input type="text" name="mid" id="mid" class="form-control" readonly required></div>
     <div class="col-sm-6">
       <span id="idChkMsg" class="text-danger"></span></div></div>
   <div class="form-group row">
     <label class="col-sm-2">비밀번호</label>
     <div class="col-sm-3">
       <input type="password" name="mpass" id="mpass" class="form-control" required></div></div>
   <div class="form-group row">
     <label class="col-sm-2">이름</label>
     <div class="col-sm-3">
       <input type="text" name="mname" value=<%=mname %> class="form-control" required></div></div>
    <div class="form-group row">
     <label class="col-sm-2">이메일</label>
     <div class="col-sm-3">
       <input type="text" value=<%=memail %> id="memail" class="form-control" value=memail readonly required></div></div>
   <select class="form-select" name="mgender" aria-label="Default select example">
   if (mgender.equals("male")) {
   <option value="male">남성</option>
   <%}else{%>
   <option value="female">여성</option>
   <%} %>
   </select>
   <div class="row">
     <div class="col-sm-10 text-right">
       <!-- <input type="button" value="이전으로" class="btn btn-secondary backBtn" > -->
       <input type="submit" value="회원가입" class="btn btn-info regBtn">
     </div></div>
 </form>
</div>
</body>
</html>
```