

Market Barly_(마켓 빨리)

음식 주문 사이트

안은종

프로젝트 기획의도 & 목차

- 기획 의도

코로나로 인한 온라인 쇼핑 증가와 함께 오프라인으로 구매가 꺼려지는 생활 필수 품목인 음식을 가지고 온라인 비 대면으로 편하고 안전한 음식 쇼핑 구성은 어떨까 하는 취지에서 프로젝트 기획.

- 목차

시스템 구성도

보유 및 사용 기술

개발 과정

순서도 (기능 분담)

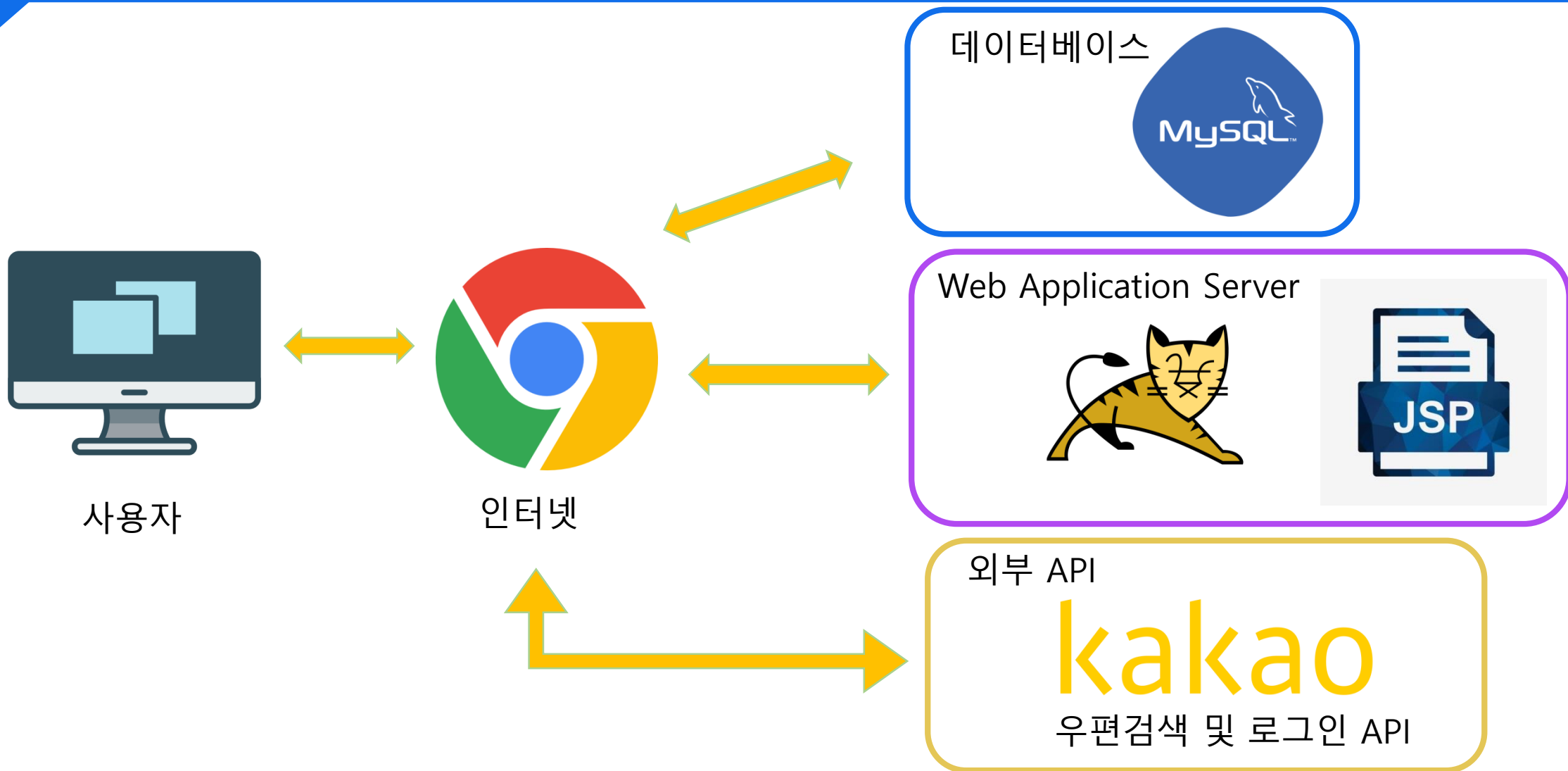
프로젝트 트리

UML / 시퀀스 다이어그램 / 메뉴 설명

기능별 사용 기술 설명

프로젝트 소감

프로젝트 시스템 구성도





Eclipse IDE for Enterprise Java Developers
(includes Incubating components)
Version: 2020-09 (4.17.0)



mysql-connector-java-5.1.46



kakao developers

Develop with
API Products



Product Version: 5.5.62



Kakao 우편번호

Kakao 로그인



Community Edition 7.1.5



Tomcat Version 8.5



개발과정

전체 스케줄 회의, 기능 분담, 시작 프로젝트 공유.



DB,DAO,DTO(vo) 만들기 + 각자의 페이지 구성.



2차 회의 및 진행 현황 수정 및 공지사항 상세 제작



전체 진행 상황 체크(진행 상황에 맞게 기능 회의)



프로젝트 전체 1차 합본

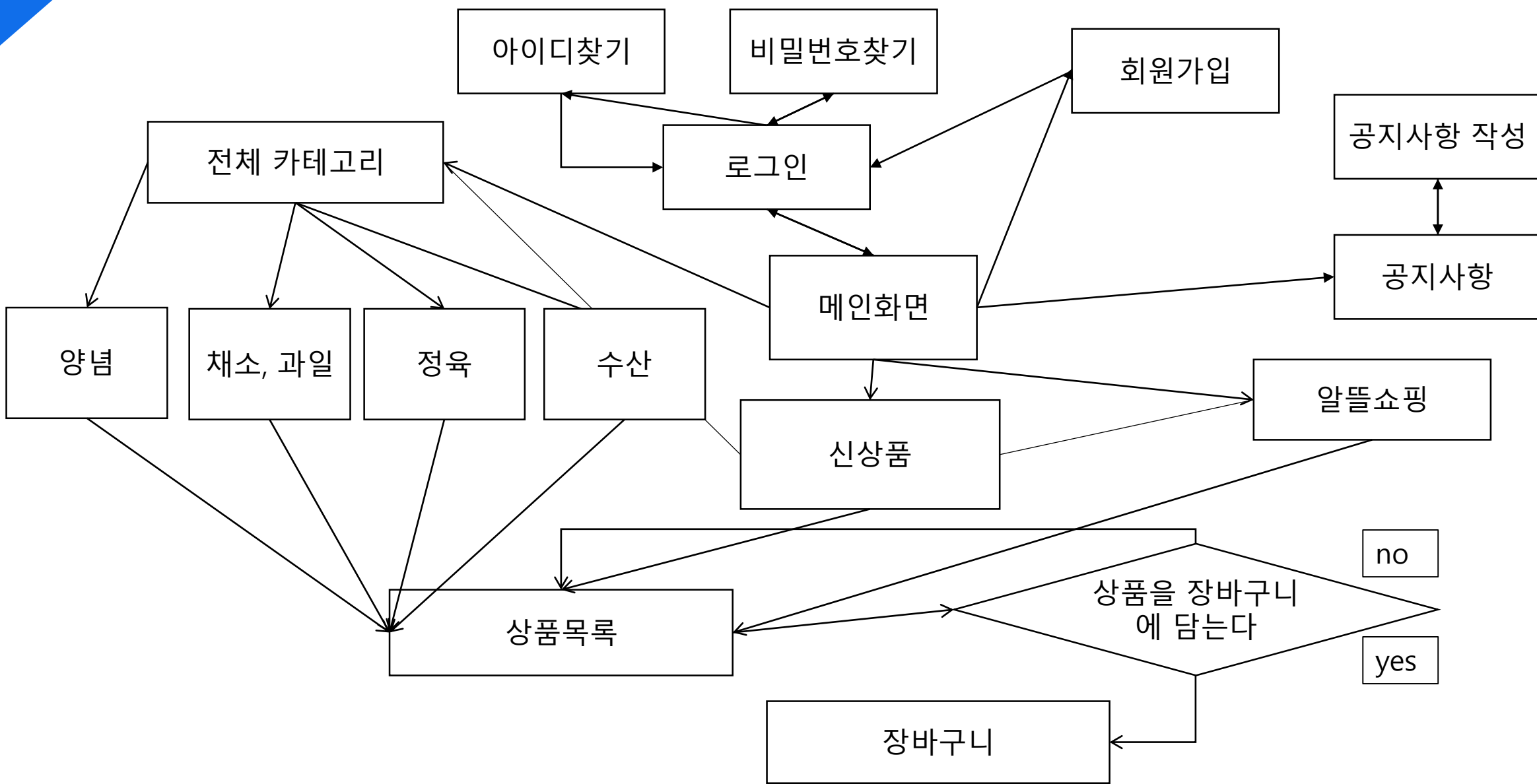


프로젝트 마무리(프로젝트 최종 합본, 영상 촬영, ppt)



프로젝트 발표!

Market Barly 메뉴 순서도



회원

```
user_barly 테이블 (  
  `id_user` varchar(30) NOT NULL,  
  `pw_user` varchar(30) NOT NULL,  
  `name_user` varchar(30) NOT NULL,  
  `email_user` varchar(50) NOT NULL,  
  `tel_user` varchar(11) NOT NULL,  
  `user_postcode` varchar(30) NOT NULL,  
  `user_address` varchar(100) NOT NULL,  
  `user_detailAddress` varchar(50) NOT NULL,  
  `user_extraAddress` varchar(50) DEFAULT NULL,  
  `birth_year` varchar(4) NOT NULL,  
  `birth_month` varchar(2) NOT NULL,  
  `birth_day` varchar(2) NOT NULL,  
  PRIMARY KEY (`id_user`)  
)
```

관리자

```
bbs 테이블 (  
  `bbsID` int(11) NOT NULL,  
  `bbsTitle` varchar(100) NOT NULL,  
  `bbsDate` datetime NOT NULL,  
  `bbsContent` varchar(2048) NOT NULL,  
  `bbsAvailable` int(11) NOT NULL  
)
```


장바구니

Category 테이블 (

```
`productid` varchar(1024) DEFAULT NULL,  
`title` varchar(1024) DEFAULT NULL,  
`price` varchar(1024) DEFAULT NULL,  
`content` varchar(1024) DEFAULT NULL,  
`img` varchar(1024) DEFAULT NULL,  
`cat` varchar(1024) DEFAULT NULL
```

)

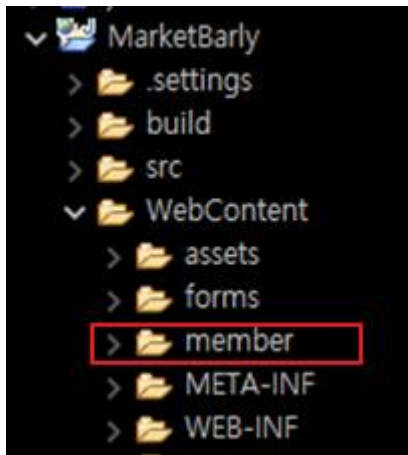
Basket 테이블 (

```
`productid` varchar(1024) DEFAULT NULL,  
`userid` varchar(1024) DEFAULT NULL,  
`img` varchar(1024) DEFAULT NULL,  
`title` varchar(1024) DEFAULT NULL,  
`price` varchar(1024) DEFAULT NULL,  
`content` varchar(1024) DEFAULT NULL
```

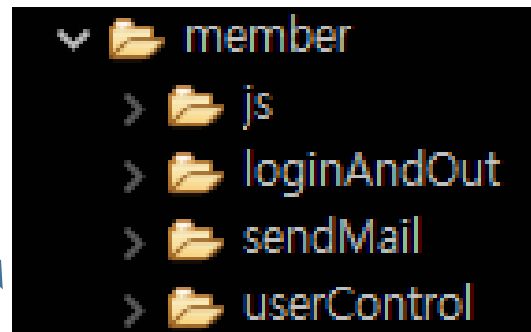
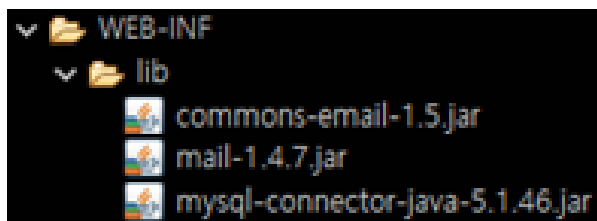
)

프로젝트 트리 - 회원 기능 페이지

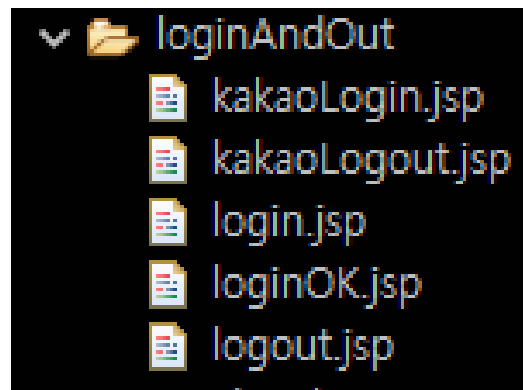
회원 폴더



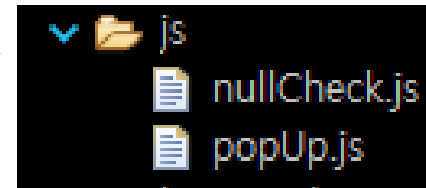
사용 jar 파일



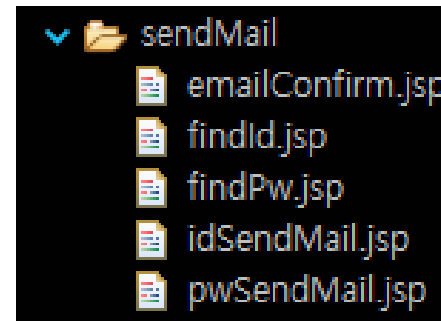
로그인 & 로그아웃 폴더



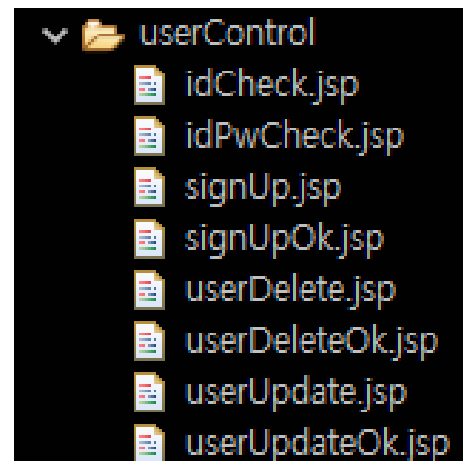
js 파일



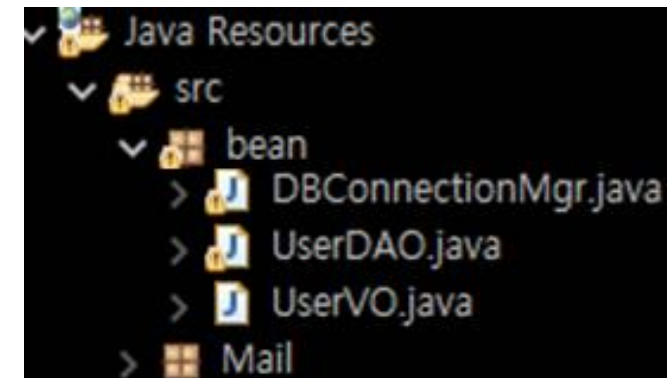
ID/PW 메일 발송 폴더



회원 수정, 삭제 폴더



사용 DB 파일

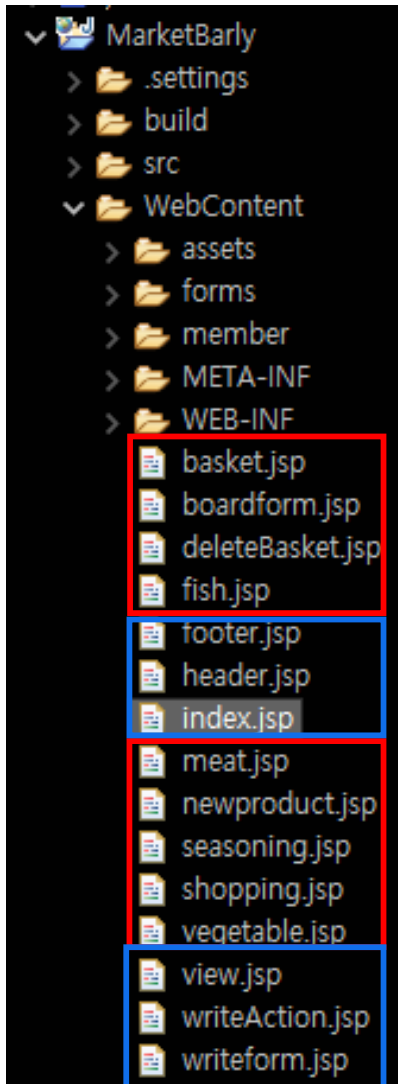


메일 클래스

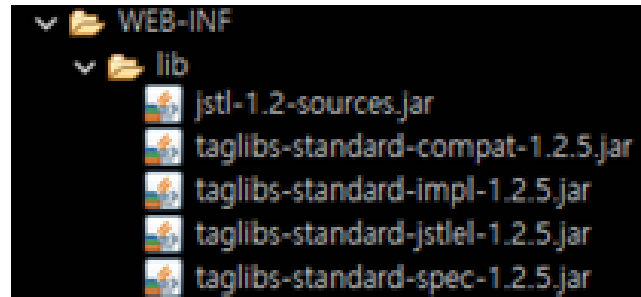


프로젝트 트리 - 관리자 & 장바구니 페이지

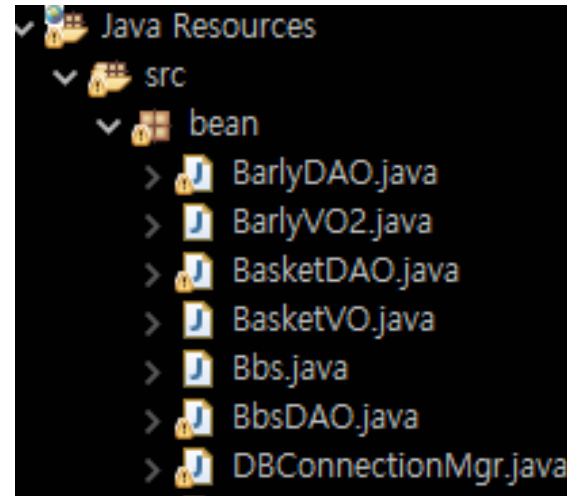
전체 페이지



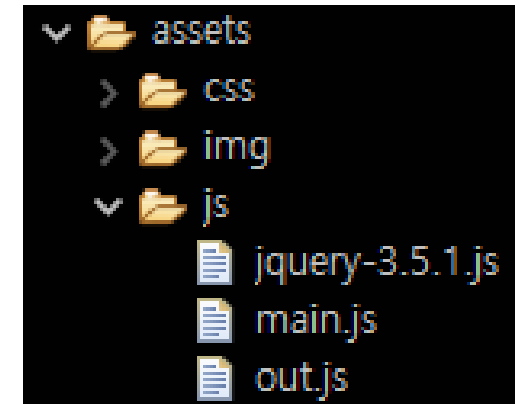
사용 jar 파일



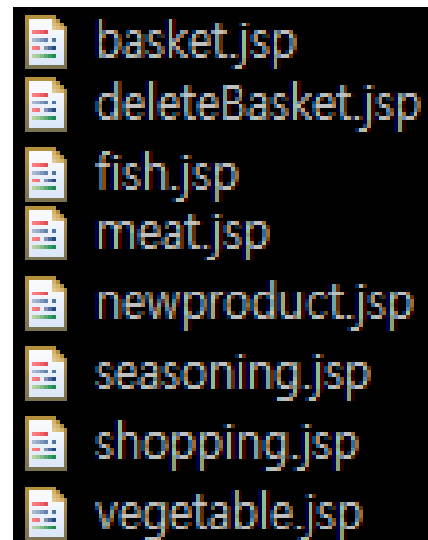
사용 DB 파일



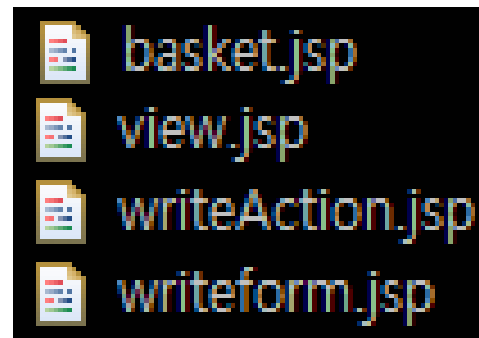
js 파일



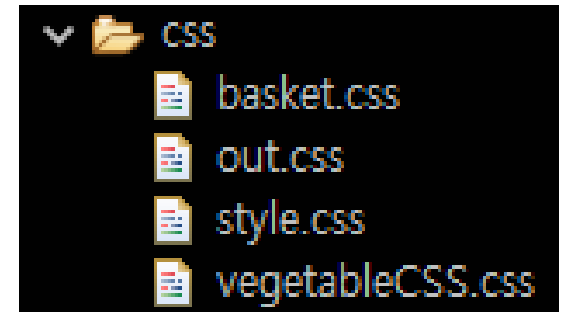
장바구니 페이지



관리자 페이지

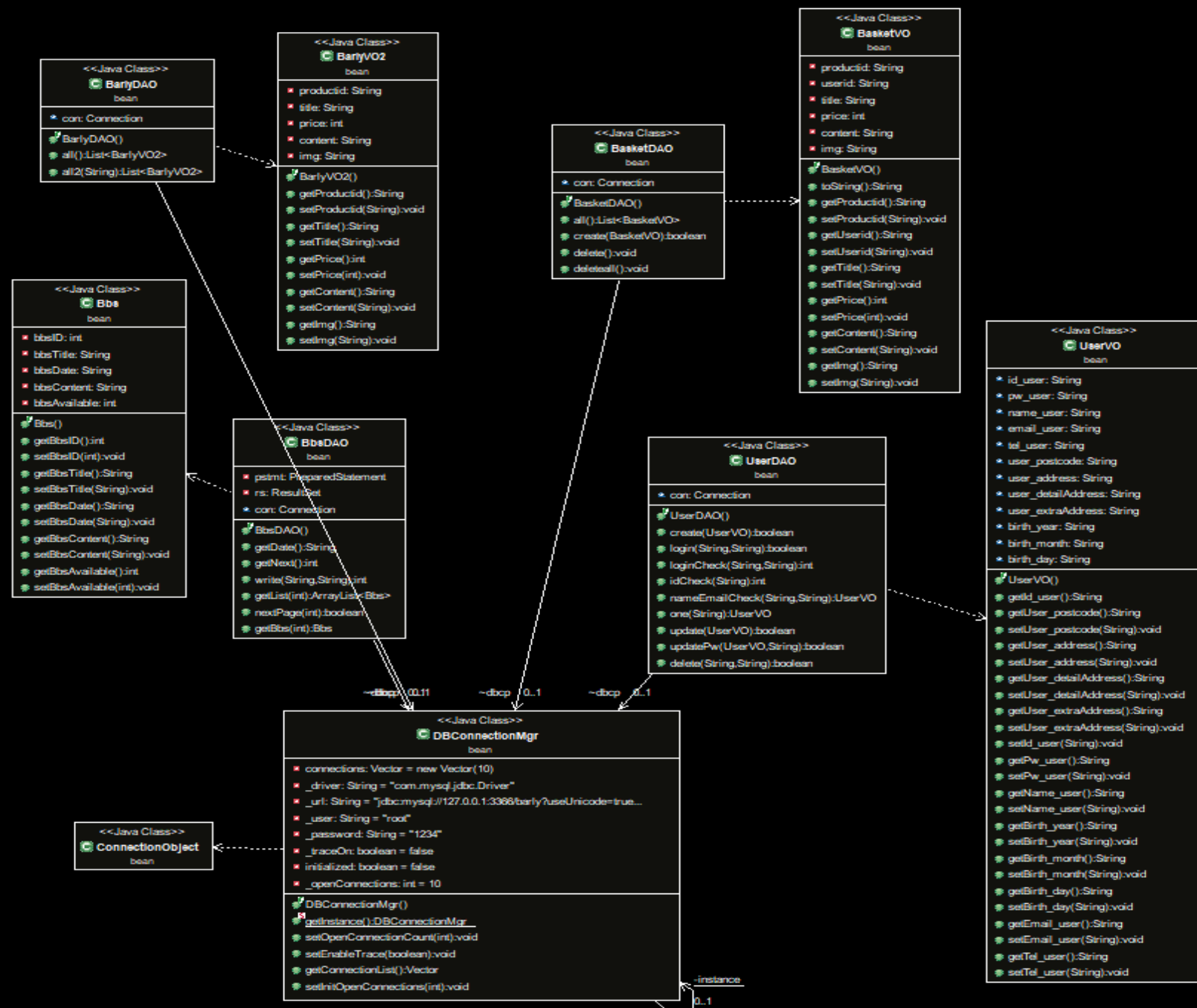


css 파일

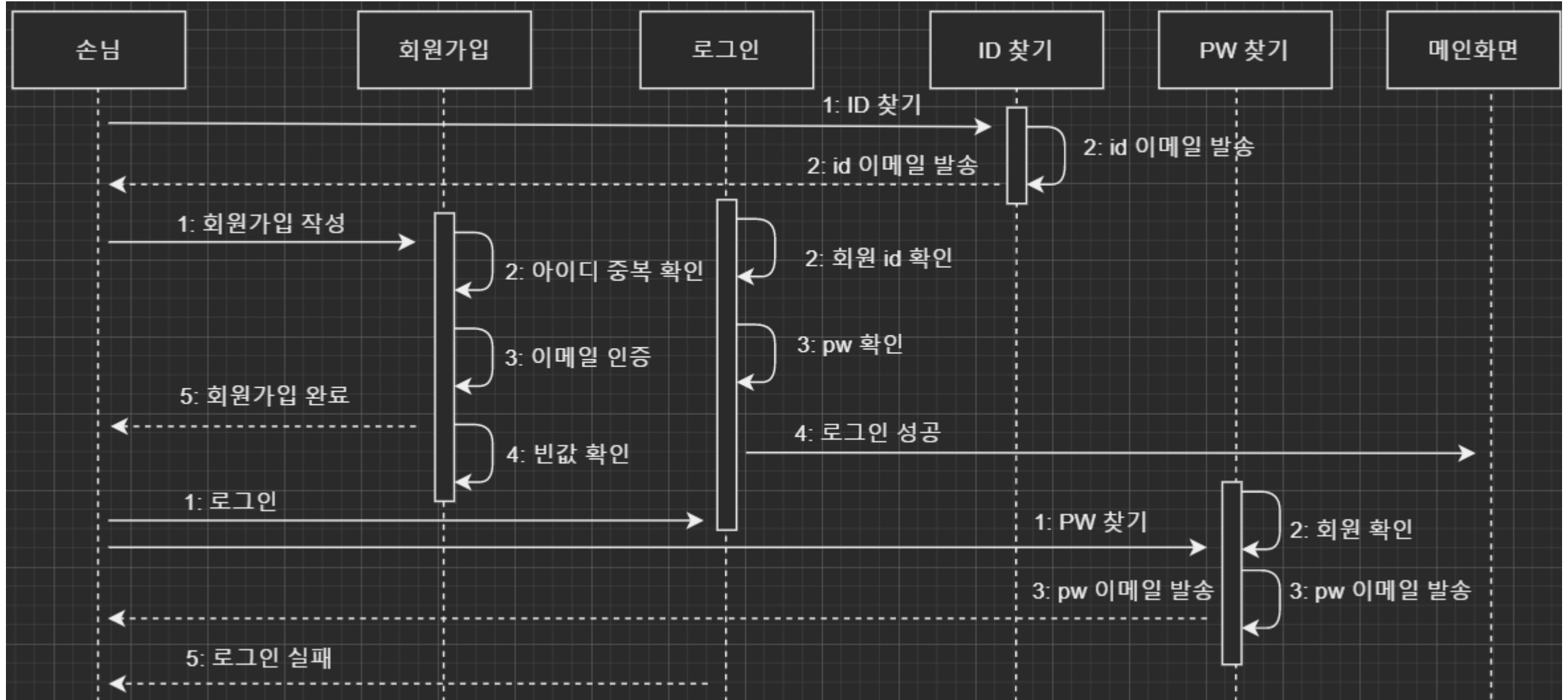


DB UML

MarketBarlyDBUML.ucl



시퀀스 다이어그램 - 회원 기능



메뉴 설명 – Market Barly 상단 버튼

[회원가입](#) [로그인](#) [공지사항](#)

Market Barly.

[전체 카테고리](#) ▼

[Home](#)

[신상품](#)

[알뜰쇼핑](#)

[장바구니](#)

상품명을 입력해주세요.

검색

채소 과일(견과류,쌀)

수산물

정육

양념

아이디

아이디 길이가 짧습니다.(최소 4자이상)

아이디

아이디 입력이 잘못 되었습니다.

아이디

사용가능한 아이디 입니다.

```
<!-- 아이디 -->
<div class="col form-group">
  <span class="badge badge-primary">아이디</span>
  <input type="text" name="id_user" id="id_user"
    class="form-control id" value="" maxlength="16"
    placeholder="6자 이상의 영문 혹은 영문과 숫자를 조합" />
  <span id="idNok">아이디는 영어와 숫자만 가능합니다.</span>
</div>
```

```
//아이디 유효검사
$('#id').keyup(function() {
  var id = $('#id_user').val();
  var idExp = /^[a-z0-9_]{6,20}$/; //아이디 길이 6~20사이 정규식
  var hangulcheck = /[ㄱ-ㅎ|ㅏ-ㅣ|가-힣]/; //한글 정규식
  var special = /[~`!>@?/<#"'\$%;:\%\.\\^,&\[*( )_+ \- =|\\{\}]/ig; // 특수문자
  //아이디 길이 확인
  if(id.length < 6){
    $('#idNok').html('아이디 길이가 짧습니다.(최소 4자이상)');
    $('#idNok').css("color","orange");
    if (hangulcheck.test(id)) {
      $('#idNok').html('한글을 입력하셨습니다. ');
      $('#idNok').css("color","red");
    }else if(special.test(id)){
      $('#idNok').html('특수 문자를 입력하셨습니다. ');
      $('#idNok').css("color","red");
    }
  }
  //아이디 공백 확인
  if(id == ""){
    $('#idNok').html('사용할 아이디를 입력해주세요. ');
    $('#idNok').css("color","orange");
  }
  //아이디 유효성 검사
  }else if (!idExp.test(id)) {
    $('#idNok').html('아이디 입력이 잘못 되었습니다. ');
    $('#idNok').css("color","red");
  }
}
```

Js에서 Class가 'id'인 태그가 keyup하면 '아이디 유효검사' 함수 동작 특수문자, 길이, 공백을 검사해서 span에 text 출력

```
String idCheck = request.getParameter("id_user");
UserDAO dao = new UserDAO();
int idSameCheck = dao.idCheck(idCheck);
%><%= idSameCheck %>
```

아이디

123

아이디 길이가 짧습니다.(최소 4자이상)

아이디

test@\$

아이디 입력이 잘못 되었습니다.

아이디

test123

사용가능한 아이디 입니다.

아이디

admin1

이미 사용중 혹은 탈퇴한 아이디 입니다.

Id 중복 시

	id_user
1	admin1

```
//아이디검사, 중복된 아이디면 '주소버튼' 비활성화
$('#id_user').focusout(function() {
$.ajax({
url: "../userControl/idCheck.jsp",
data: {
id_user: $('#id_user').val()
},
success: function(idSameCheck) {
if(idSameCheck == 0){
$('#idNok').html('이미 사용중 혹은 탈퇴한 아이디 입니다.')
$('#idNok').css("color","red");
$('#duplicateCheck').prop("disabled",true); //버튼 비활성화
}else if(idSameCheck == 1){
$('#idNok').html('사용가능한 아이디 입니다.')
$('#idNok').css("color","green");
$('#duplicateCheck').prop("disabled",false); //버튼 활성화
}
} //success
}) //ajax
}) //id_user에서 focusout될때
```

Js에서 id가 'id_user'인 태그가 focusout 하면 '아이디 검사' 함수 동작 ajax로 화면 전환 없이 DB에 아이디 등록 여부를 확인 후 결과 값을 반환하여 span에 text 출력, 만약 중복된 아이디가 확인되면 '가입하기 버튼' 비활성화로 중복 아이디 가입 방지.

이메일

aej9300@gmail.com

이메일 인증

MarketBarly 신규 Market Barly에서 이메일을 발송하였습니다.

```
public class MailUtil {
    public static void sendMail(String email, String subject, String content)
        // Mail Server 설정
        String charSet = "utf-8"; //메일 한글 설정
        String hostSMTP = "smtp.gmail.com"; // SMTP 서버명
        String hostSMTPid = "marketbarly"; // 아이디
        String hostSMTPpwd = "barly2020!"; // 비밀번호
        // 보내는 사람
        String fromEmail = "marketbarly@gmail.com"; //보내는사람@메일.com
        String fromName = "MarketBarly"; //보내는사람 이름
        // email 전송
        try {
            HtmlEmail mail = new HtmlEmail();
            mail.setDebug(true);
            mail.setCharset(charSet);
            mail.setSSLonConnect(true); // SSL 사용 (TLS가 없는 경우 SSL 사용)
            mail.setHostName(hostSMTP);
            mail.setSmtPort(465); // SMTP 포트 번호

            mail.setAuthentication(hostSMTPid, hostSMTPpwd);
            mail.setStartTLSenabled(true); // TLS 사용
            mail.addTo(email);
            mail.setFrom(fromEmail, fromName, charSet);
            mail.setSubject(subject);
            mail.setHtmlMsg(content);
            mail.send();
        } catch (Exception e) {
```

```
<%@page import="Mail.MailUtil"%>
<%@page import="bean.UserVO"%>
<%@page import="bean.UserDAO"%>
<%@page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    String name_user = request.getParameter("name_user");
    String email_user = request.getParameter("email_user");

    UserDAO dao = new UserDAO();
    UserVO vo = dao.nameEmailCheck(name_user,email_user );
    String user_id = vo.getId_user();

    if(user_id == null){
        user_id = "가입하신 아이디가 없습니다.";
    }

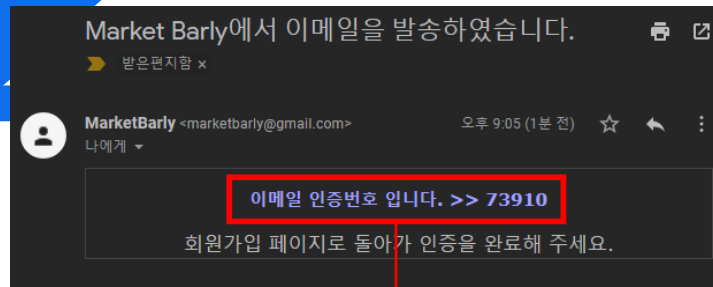
    String title = "Market Barly에서 이메일을 발송하였습니다.";

    String content = "";
    content += "<div align='center' style='border:1px solid black; font-size: 1.2em; color: blue; text-align: center; padding: 10px;'>";
    content += "<h3 style='color: blue;'>고객님의 아이디는 </h3>";
    content += user_id;
    content += "<div style='font-size: 1.2em;'>";
    content += "<a href='http://localhost:8888/MarketBarly/member/login'>로그인</a>";
    content += "</div>";

    MailUtil.sendMail(email_user, title, content);

    System.out.println(name_user+"님에게 아이디 발송 메일이 성공하였습니다.");
%>
```

'이메일 인증' 버튼 클릭 시 ajax가 동작하며 메일을 보내기 위한 jsp 화면에서 ajax로 넘긴 값을 확인 후 Java src 폴더에서 MailUtil 클래스에 있는 sendMail 함수를 가져와 이름과 이메일이 DB에 등록된 것과 같으면 id를 가져와서 sendMail함수에 넣어서 사용자가 등록한 이메일로 메일 발송.



이메일
aej9300@gmail.com

이메일 인증

인증코드 73910

인증코드 739101

인증코드를 다시 확인해 주세요

인증코드를 다시 확인해 주세요.

인증코드 73910

인증이 성공하였습니다.

```
//이메일 인증 메일 발송
$('#emailConfirm').click(function() {
    $.ajax({
        url: "../sendMail/emailConfirm.jsp",
        data: {
            email_user: $('#email_user').val(),
        },
        success: function(tempW) {
            tempW2 = tempW.trim("//엔터를 지우는 코드
            console.log("-----메일 인증 코드: " + tempW2)
            //인증 div 보이기
            $('#userEmailCodeDiv').css('display', 'inline-block');
            //인증코드 확인 안하고 가입하기 클릭시 alert
            $('#userEmailOK').mouseover(function() {
                alert("인증코드를 다시 확인해 주세요.");
                $('#userEmailCode').focus();
            })//userEmailOK mouseover 시
            //인증코드 비교 확인
            $('#userEmailCodeButton').click(function() {
                userEmailCode = $('#userEmailCode').val();
                console.log("-----회원가입 input 값: " + userEmailCode)
                if(tempW2 == userEmailCode){
                    $('#emailCodeCheck').html('인증이 성공하였습니다.');

```

Ajax로 넘겨 받아 emailConfirm을 거쳐 만들어진 랜덤 값을 jsp화면으로 넘겨서 success에서 사용자가 입력한 값과 만들어진 인증코드 값을 확인 후 비교 상태를 #emailCodeCheck 가 부여된 태그에 text를 출력해주며 인증 '확인' 버튼 클릭 시에만 회원가입이 완료되도록 설정.

우편번호를 입력해주세요

확인

이메일

aej9300@gmail.com

이메일 인증

인증코드

73910

확인

인증이 성공하였습니다.

휴대폰

01044861158

주소 찾기

우편번호

주소

상세주소

참고항목

생년월일

1999 12 31

이용약관

☒ 동의하지 않으면 회원가입이 불가능 합니다.
*배송을 위한 정보 확인외에는 사용되지 않습니다.

가입하기

```
<form action="signUpOk.jsp" name="signUp" onsubmit="return formCheck()">
```

```
//tel null값 체크
if(!document.signUp.tel_user.value){
    alert("휴대폰을 입력해주세요");
    document.signUp.tel_user.focus();
    return false;
}
//주소1 null값 체크
if(!document.signUp.user_postcode.value){
    alert("우편번호를 입력해주세요");
    document.signUp.user_postcode.focus();
    return false;
}
//주소2 null값 체크
if(!document.signUp.user_address.value){
    alert("주소를 입력해주세요");
    document.signUp.user_address.focus();
    return false;
}
//주소3 null값 체크
if(!document.signUp.user_detailAddress.value){
    alert("상세주소를 입력해주세요");
    document.signUp.user_detailAddress.focus();
    return false;
}
//주소4 null값 체크
if(!document.signUp.user_extraAddress.value){
    alert("참고항목을 입력해주세요");
    document.signUp.user_extraAddress.focus();
    return false;
}
```

사용자가 입력한 값이 없이 '가입 버튼'을 클릭 하여도 form에서 return formCheck() 함수를 확인 하여 빈 태그가 있으면, alert로 사용자에게 알려주며, focus() 내장 함수로 해당 위치에 커서를 옮겨 준다.

생년월일 1234 MM DD 년도가 잘못 되었습니다.

생년월일 1999 13 DD 1월~12월 중 입력해주세요.

생년월일 1999 12 32 날짜를 확인해주세요.

```
//생년월일 유효검사
$('.birth').keyup(function() {
    var birth_year = $("#birth_year").val();
    var birth_month = $("#birth_month").val();
    var birth_day = $("#birth_day").val();
    if (birth_year < 1920 || birth_year > 2006){
        $('#birthNok').html('년도가 잘못 되었습니다.');
```

이용약관

☐ 동의하지 않으면 회원가입이 불가능 합니다.
*배송을 위한 정보 확인외에는 사용되지 않습니다.

가입하기

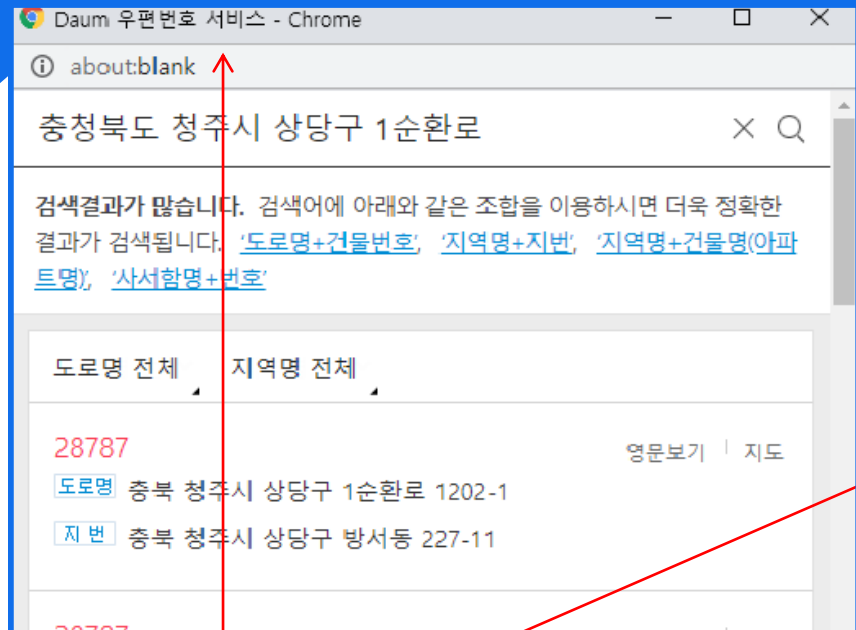
```
//이용약관 체크박스 체크 후 가입하기 버튼 활성화
$(".sumbit button").prop("disabled", true);
$("#customCheck").on('click',function(){
    var chk = $('input:checkbox[id="customCheck"]').is(":checked");
    if(chk==true){
        $(".sumbit button").removeProp('disabled');
```

이용약관

☒ 동의하지 않으면 회원가입이 불가능 합니다.
*배송을 위한 정보 확인외에는 사용되지 않습니다.

가입하기

Js로 클래스가 'birth'인 태그가 keyup되면 생년월일 input 태그에서 사용자가 입력한 값으로 유효검사 후, 잘못 입력했을 경우 text로 표시 이후, 사용자가 '이용약관'을 체크하면 가입하기 버튼이 활성화 된다.



주소 찾기

28787

충북 청주시 상당구 1순환로 1202-1

123|

(방서동)

```
<!-- 주소 -->
<div class="col form-group">
  <input type="button" onclick="searchAddress()" value="주소 찾기"
    class="btn btn-primary btn-lg" id="duplicateCheck"><br> <input
```

```
//외부 api로 주소 받아오는 함수
function searchAddress(){
  new daum.Postcode({
    oncomplete: function(data) {
      // 팝업에서 검색결과 항목을 클릭했을때 실행할 코드를 작성하는 부분.

      // 각 주소의 노출 규칙에 따라 주소를 조합한다.
      // 내려오는 변수가 값이 없는 경우엔 공백('')값을 가지므로, 이를 참고하여 분기 한다.
      var addr = ''; // 주소 변수
      var extraAddr = ''; // 참고항목 변수

      //사용자가 선택한 주소 타입에 따라 해당 주소 값을 가져온다.
      if (data.userSelectedType === 'R') { // 사용자가 도로명 주소를 선택했을 경우
        addr = data.roadAddress;
      } else { // 사용자가 지번 주소를 선택했을 경우(J)
        addr = data.jibunAddress;
      }
    }
  });
}
```

'주소 찾기'버튼 클릭시 연결해 놓은 kakao 우편 검색 api 가 동작 하여 input 태그에 설정한 'id'에 맞추어서 검색한 값이 들어간다.

아이디를 입력해주세요

비밀번호를 입력해주세요

아이디 찾기 비밀번호 찾기

로그인 회원가입

*게스트로 이용시만 카카오 로그인을 추천합니다

카카오계정으로 로그인

kakao

aej9300@gmail.com

.....

☒ 로그인 상태 유지

로그인

또는

QR코드 로그인

회원가입 카카오계정 비밀번호 찾기

클릭 시

로그인 성공시

```
<!-- 카카오톡 아이디 연동해서 로그인 -->
<script src = "//developers.kakao.com/sdk/js/kakao.min.js"></script>
<a id="kakao-login-btn"></a>
<a href="http://developers.kakao.com/logout"></a>
```

```
Kakao.Auth.createLoginButton({
  container: '#kakao-login-btn',
  success: function(authObj) {
    Kakao.API.request({
      url: '/v2/user/me',
      success: function(res) {
```

'카카오계정로그인'버튼 클릭 시 연결해 놓은 kakao 로그인 api 가 동작 하여 로그인이 성공하면 카카오에서 받은 '토큰' 을 가지고 세션을 등록하여 로그인 여부를 확인.

```
//카카오 토큰저장
String kakao_token = request.getParameter("kakao_token");
session.setAttribute("kakao_sessionToken", kakao_token);
response.sendRedirect("../../index.jsp");
```

카카오로 로그인하여 사용하시는 이용자는 '회원가입'을 해야합니다.

공지사항 로그아웃

아이디 찾기

test

aej9300@gmail.com

확인

사용자의 이름과 이메일을
확인하여 DB에서 맞는 값
이 있으면 ID를 이메일로
전달해 준다.

Market Barly에서 이메일을 발송하였습니다.

받은편지함 x



MarketBarly <marketbarly@gmail.com>

나에게 ▾

오후 9:12 (0분 전)



고객님의 아이디는

test123

[로그인 페이지로 돌아가 로그인해주세요.](#)

<%@page import="Mail.MailUtil"%>

```
String name_user = request.getParameter("name_user");
String email_user = request.getParameter("email_user");

UserDAO dao = new UserDAO();
UserVO vo = dao.nameEmailCheck(name_user, email_user);
String user_id = vo.getId_user();

if(user_id == null){
    user_id = "가입하신 아이디가 없습니다.";
}

String title = "Market Barly에서 이메일을 발송하였습니다.";

String content = "";
content += "<div align='center' style='border:1px solid";
content += "<h3 style='color: blue;'>고객님의 아이디는 </h3>";
content += user_id;
content += "<div style='font-size: 130%'>";
content += "<a href='http://localhost:8888/MarketBarly/";

MailUtil.sendMail(email user, title, content);
```


비밀번호 찾기

test

123

회원이 아닙니다.

가입시 등록하신 이메일을 입력해주세요

확인

test

test123

회원이십니다.

aej9300@gmail.com

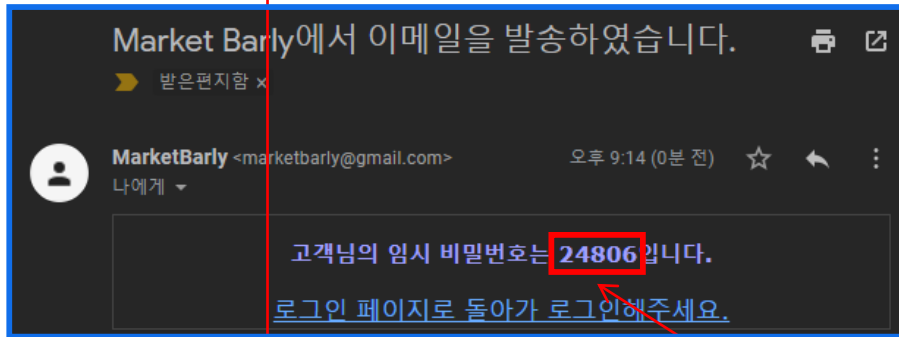
확인

```
String idCheck = request.getParameter("id_user");
UserDAO dao = new UserDAO();
int idSameCheck = dao.idCheck(idCheck);
%><%= idSameCheck %>
```

사용자의 id를 체크 후, 맞는 id가 있다면 회원인지 아닌지 확인 후, text로 출력하여 보여준다.

```
//이메일 비밀번호 찾기시 아이디(회원) 확인
$('#id_user').focusout(function() {
    $.ajax({
        url: "../userControl/idCheck.jsp",
        data: {
            id_user: $('#id_user').val()
        },
        success: function(idSameCheck) {
            if(idSameCheck == 0){
                $('#pwNok').html('회원이십니다.')
                $('#pwNok').css("color","green");
            }else if(idSameCheck == 1){
                $('#pwNok').html('회원이 아닙니다.')
                $('#pwNok').css("color","red");
            }
        }
    })//success
})//ajax
})//id_user에서 focusout될때
```


	id_user	pw_user	name_user	email_user
1	admin1	123	admin_test	admin@gmail.com
2	test123	123	test	aej9300@gmail.com



2	test123	24806	test	aej9300@gmail.com
---	---------	-------	------	-------------------

```
UserDAO dao = new UserDAO();
boolean result = dao.updatePw(vo, temPw);
MailUtil.sendMail(email_user, title, content);
```

```
<jsp:useBean id="vo" class="bean.UserVO"></jsp:useBean>
<jsp:setProperty property="*" name="vo"/>
<%! public int getRandom(){
    int random = 0;
    random = (int)Math.floor((Math.random()*(99999-10000+1))+10000);
    return random;
} %>
<%
    int temporaryPw = getRandom();
    String temPw = Integer.toString(temporaryPw);

    String name_user = request.getParameter("name_user");
    String email_user = request.getParameter("email_user");
```

고객님의 임시 비밀번호는 "+temPw+"입니다.<

사용자의 이름과 이메일을 받아서 사용자가 잊어버린 암호를 임의 값으로 수정하여 이메일로 보냄과 동시에 DB에 해당 id와 이메일이 같으면 임의 값을 수정해준다.