



## 정상우 포트폴리오

안녕하세요. 웹 개발자를 준비 중인 정상우입니다. 새로운 기술을 적용하고 공부하면서 흥미를 느끼며 완성된 결과물을 보면 성장해나가는 것에 보람을 느끼는 개발자로 성장하고 싶습니다.

프로젝트 경험을 통해 개발자로서 갖추어야 하는 필요한 역량을 배우면서 회사에 도움이 되는 개발자로 한걸음 나아가겠습니다.



## 정기배송형 도시락 판매 사이트 '둔든'

### 개발기간

- 개발인원 : 7인
- 2023.06.21.~ 07.12

### 개발환경

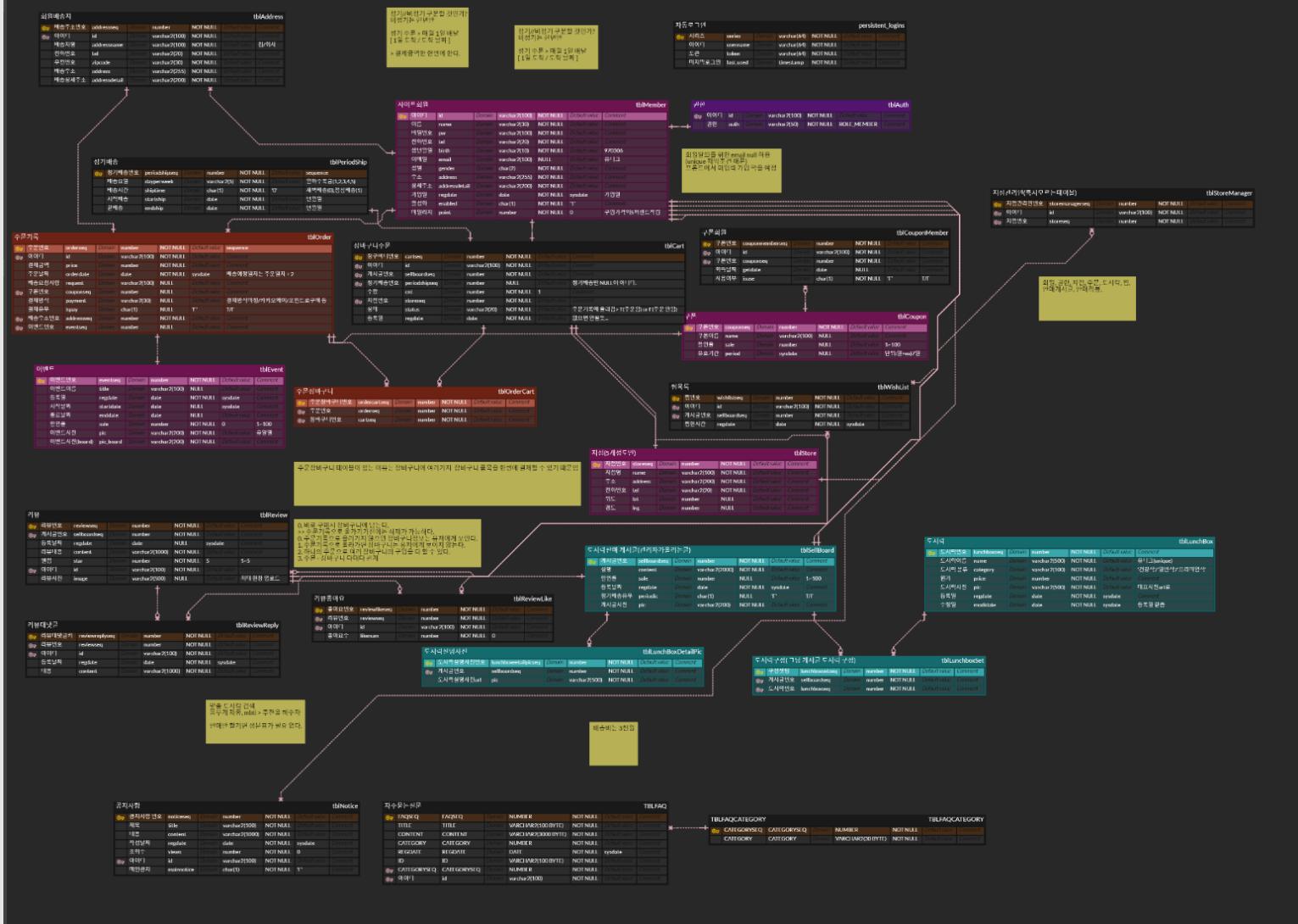
- Eclipse(Spring Tool Suite 3)
- SourceTree
- ErdCloud
- Github
- Sql Developer
- Figma
- Apache Tomcat 8.5

### 사용기술

- Java 11
- Spring MVC
- Spring Security
- MyBatis, Oracle 11g(AWS에 설치하여 공용으로 사용)
- HTML, CSS, JavaScript, Jquery, JSP, Bootstrap
- Kakao login api, Kakao chat api, javax.mail api, daum 주소 api, limport api, sweatalert2, kakao map api, summernote, Full calendar api 등 ~

# 든든 프로젝트

## 데이터 베이스 설계



데이터 베이스 설계

# 든든 프로젝트

## 구현 기능

The screenshot shows a web-based administration interface for a food delivery service. On the left, a sidebar menu includes '관리자 Admin', '이용자 관리 User Management' (selected), '가맹점 관리 Franchise Management', '상품 관리 Product Management', '주문 관리 Order Management', '이벤트 관리 Event Management', '쿠폰 관리 Coupon Management', and '통계 Statistics'. The main content area displays a user profile for '홍길동' (Hong Gil Dong) with ID 'Test'. The profile includes fields for Name, ID, Contact, Gender, Birthdate, Email, Address, Zipcode, Grade, and Registration Date. Below this is a section titled '주문 리스트' (Order List) with a table showing 10 entries per page. The table columns are Order Number, Product Name, Price, and Order Date. Each row lists a different food item from the '로제 쉬림프 치킨 도시락' series.

주문번호	상품명	결제금액	주문날짜
11	로제 쉬림프 치킨 도시락	5657	2023-07-06 06:00:59
11	로제 쉬림프 치킨+두루치기 도시락	5657	2023-07-06 06:00:59
11	뜨끈뜨끈 소불고기 도시락	5657	2023-07-06 06:00:59
11	뜨끈뜨끈 두루치기 도시락	5657	2023-07-06 06:00:59
11	VIP 도시락	5657	2023-07-06 06:00:59
11	명품 연잎밥 도시락	5657	2023-07-06 06:00:59
11	명품 연잎밥 도시락	5657	2023-07-06 06:00:59
11	파스타 샐러드	5657	2023-07-06 06:00:59
11	파스타 샐러드	5657	2023-07-06 06:00:59

## 회원 관리 기능

Rest 기반을 통한 회원 CRUD기능  
검색한 회원의 상품구매 내역 조회 기능  
을 구현하였습니다.

# 든든 프로젝트

## 구현 기능

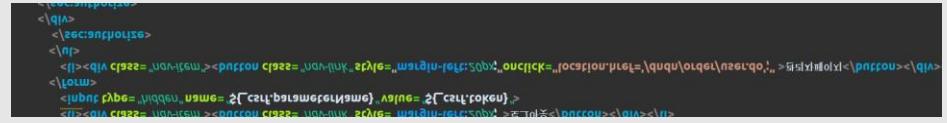
The screenshot shows a web-based event management system. At the top, there's a header with a logo, user information (Admin), and navigation tabs like '관리자', '이용자 관리', '가맹점 관리', '상품 관리', '주문 관리', '이벤트 관리', '쿠폰 관리', and '통계'. Below the header, there's a section for '이벤트 썸네일' and '이벤트 콘텐츠' with preview images. A main table lists events with columns: '이벤트 이름', '시작날짜', '종료날짜', and '할인율'. Below this is a file upload section for event photos. The bottom part shows a table titled '이벤트 리스트' with columns: '이벤트 번호', '이벤트 이름', '등록일', '시작날짜', '종료날짜', '할인율', and '이벤트 사진'. It lists six events with their respective details and file names.

이벤트 번호	이벤트 이름	등록일	시작날짜	종료날짜	할인율	이벤트 사진
5	신규회원이라면 누구나!	2023-06-30	2023-06-30	2024-06-30	20 %	event-content_1.jpg
25	5월이라 준비했5	2023-07-10	2023-05-05	2024-05-05	55 %	01.jpg
26	4고 10% 할인받고 또 4고	2023-07-10	2023-04-14	2024-04-16	15 %	0428_intro.jpg
4	도시락 먹고 여름 JUNE비 하자	2023-06-03	2023-06-11	2023-07-11	15 %	EC9DB8ED8A8E8EBA19C.jpg
2	홍보리더를 위한 조류가 할인!	2023-04-28	2023-05-02	2023-05-31	12 %	0428_intro.jpg
1	도시락 먹고 여름 JUNE비 하자	2023-03-27	2023-04-03	2023-03-27	27 %	EC9DB82CEA83B5ECA780.jpg

## 이벤트 관리 기능

파일업로드를 통하여 이벤트 썸네일과 이벤트 내용을 담은 이미지를 업로드하여 이벤트 페이지에서 이벤트를 제공하는 기능을 구현하였습니다.

이벤트 리스트를 조회하여 상단의 이벤트 상세 페이지를 통하여 이벤트를 관리할 수 있도록 구현하였습니다.



## 관리자권한 확인

Csrf 토큰을 이용하여 db에 저장되어있는 관리자 권한을 가진 회원 만 관리자 페이지에 접속할 수 있도록 적용시켰습니다.

# 든든 프로젝트

## 구현 기능

The screenshot shows a web-based event management system. At the top, there's a navigation bar with links for '메뉴소개', '매장찾기', '이벤트', '고객지원', '회원가입', and '로그인'. Below the navigation is a logo for '오늘은 든든' (Today is Dunn's). The main content area is titled 'EVENT | 이벤트' and features three event cards:

- 신규회원 월컵쿠폰 20,000원**: A green card for new members with a 20,000 won coupon.
- 5월이라 준비했5**: An orange card for May with a family illustration.
- 4고 10% 할인받고 또 4고**: An orange card for April with a man pointing upwards.

Each card includes a status indicator ('진행중') and a date range ('2023-06-30 ~ 2024-06-30' for the first, '2023-05-05 ~ 2024-05-05' for the second, and '2023-04-14 ~ 2024-04-16' for the third).

## 이벤트 관리 기능

업로드를 통해 데이터베이스에 저장된 이벤트 데이터를 통하여 이벤트페이지 구현하였습니다.

The screenshot shows a detailed view of an event page. At the top right is a logo for '오늘은 든든'. The main title is 'EVENT | 이벤트 상세'. The central part of the page features a large banner for the '5월이라 준비했5' event, dated from May 5 to May 5, 2024. The banner includes text about preparing for May and offers for 6-piece and 8-piece sets. Below the banner are two images of food items: '연생도시락 1팩 증정' and '연생도시락 2팩 증정'. At the bottom, there's a section titled '▶ 유의사항' (Notes) with several bullet points in Korean.



## 정다목적 커뮤니티 사이트

### 개발기간

- 개발인원 : 7인
- 2023.06.21.(수) ~ 07.12(수)

### 개발환경

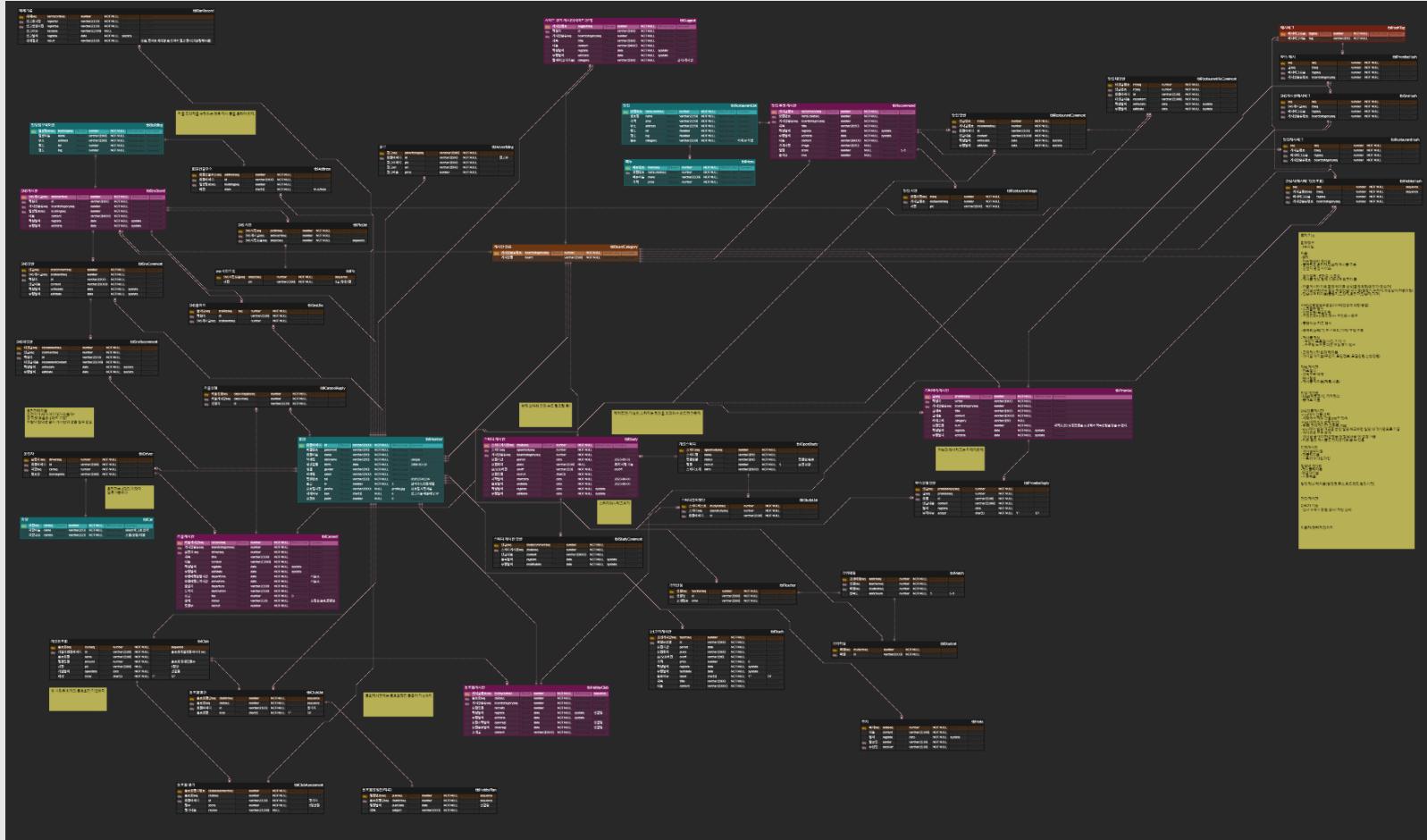
- Eclipse
- SourceTree
- ErdCloud
- Github
- Sql Developer
- Figma
- Apache Tomcat 8.5

### 사용기술

- Java 11
- Servlet jsp
- MyBatis, Oracle 11g(AWS에 설치하여 공용으로 사용)
- HTML, CSS, JavaScript, Jquery, JSP, Bootstrap

# 빌딩숲 프로젝트

## 데이터 베이스 설계



데이터베이스 설계

## 구현 기능

The left screenshot shows the 'Study Group Creation' page ('스터디 개설') with fields for 'Study Name' (스터디명), 'Meeting Day' (모집일), 'Meeting Frequency' (모집 주기), 'Location' (주소), 'Start Date' (시작 날짜), 'End Date' (종료 날짜), and 'Description' (소개 내용). The right screenshot shows the 'Study Group List' page ('스터디 게시판') displaying a grid of study groups, each represented by a thumbnail image of a book and a pen, with the text '진행 중' (In Progress) above it.

## 모집글 개설

회원이 스터디모집글을 개설하여 을 등록하여 데이터베이스에 추가하여 회원을 모으는 기능을 구현하였습니다.  
데이터베이스의 유저정보를 통하여 회원만이 모집글을 작성할 수 있도록 구현하였습니다.

# 빌딩숲 프로젝트

## 구현 기능

The screenshot shows a study detail page titled "자바 공부 스터디". It displays basic study information like name, location, and schedule. Below this is a table of posts from study members, each with a timestamp and a "like" button. At the bottom, there's a search bar and a post creation form.

번호	제목	작성자	작성일	조회
1	안녕하세요	이알알	2023-06-11	2
2	test	위축되는돼지338	2023-06-11	0
3	test	위축되는돼지338	2023-06-11	0
4	test	위축되는돼지338	2023-06-11	0
5	test	위축되는돼지338	2023-06-11	0

The screenshot shows a post creation and editing interface. It includes fields for title, content, and attachments. A preview of the post is shown at the top, and a list of existing posts is visible below it.

## 게시글 작성 기능

Rest 기반을 통하여 스터디별로 가입한 회원에 한하여 게시판기능을 구현하였습니다.

해당스터디에 가입한 회원만이 게시글을 작성, 조회, 수정, 삭제 가능하도록 유효성검사를 구현하였습니다.



## 오라클 프로젝트 교육센터 운영 데이터베이스

- 개발인원 : 5인
- 2023.04.03 ~ 04.14

### 사용기술

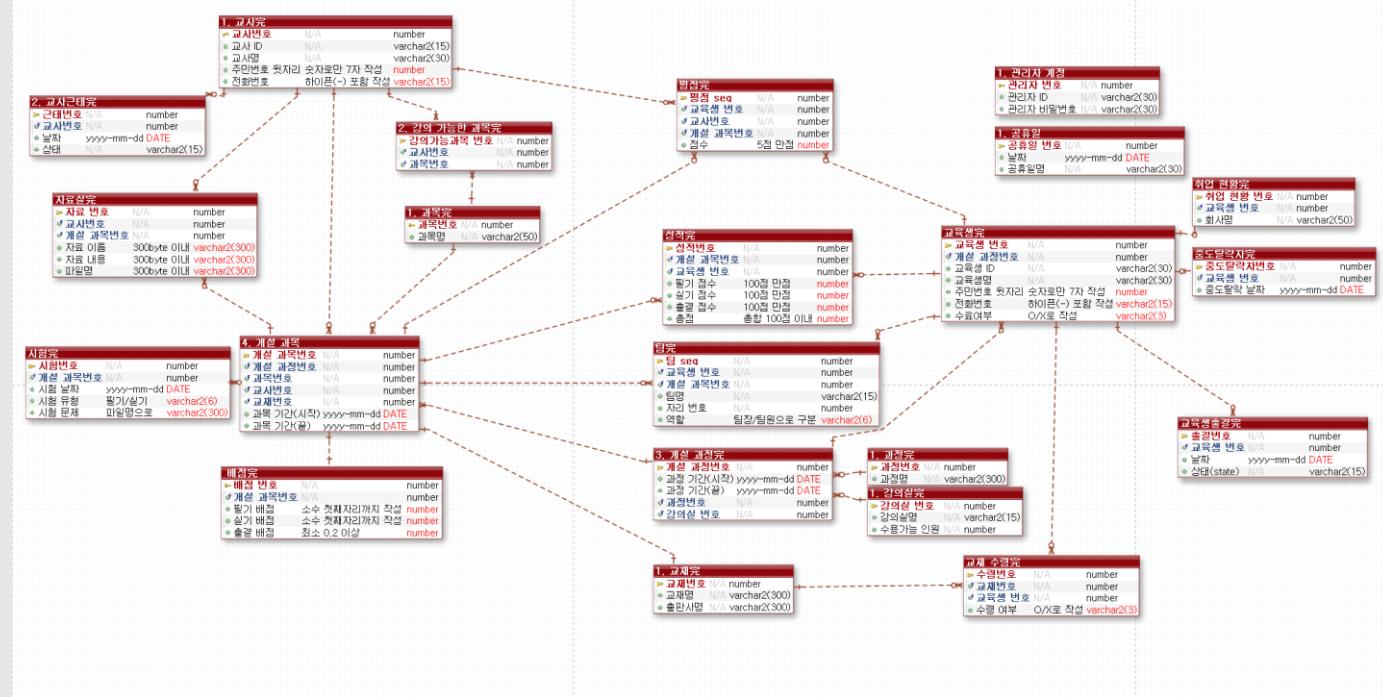
- Oracle 11g

### 개발환경

- Exerd
- Sql Developer

# 교육센터 운영 프로젝트

## 데이터베이스 설계



## 데이터베이스 설계

교사, 교육생 릴레이션을 통하여 해당 목표의 DBMS 에 맞게 물리적인 구조를 구현하였습니다.

# 교육센터 운영 프로젝트

## 구현 쿼리문

```
-- 과정 목록 출력 (과정 선택 받기)
select * from vwOpenClass;

-- 선택한 과정의 과목 목록 출력 (과목 선택 받기)
select
    과목번호,
    과목명,
    과목기간,
    교재명,
    교사명
from vwOpenSubject
    where 과정번호 = 14 -- 선택한 과정 입력
        order by 과목번호;

-- 선택한 과목의 팀 정보 출력
select
    t.name as 팀명,
    t.role as 역할,
    s.s_seq as 교육생번호,
    s.name as 이름,
    t.seatnum as 자리
from tblTeam t
    inner join tblStudent s
        on s.s_seq = t.s_seq
            inner join tblOpenClass oc
                on oc.oc_seq = s.oc_seq
                    where t.os_seq = 77 -- 선택한 과목 번호
                        order by t.name, role desc;
```

## 해당과목의 팀정보 출력

해당과목을 수강하는 학생들의 정보를 출력하는 쿼리문입니다.

과정번호	과정명	과정기간	강의설명	팀명	역할	교육생번호	이름	자리
1	1 JAVA 기반 백엔드, 클라우드 활용 프로젝트	22/01/03 ~ 22/06/17	1강의실	1 1팀	팀장	96	강아민	11
2	2 전자정부표준 프레임워크 기반 풀-스택 개발자 양성과정	22/02/07 ~ 22/08/05	2강의실	2 1팀	팀원	93	조솔오	22
3	3 빅데이터를 활용한 인공지능(AI) 기술 구현 엔지니어 양성 과정	22/02/23 ~ 22/09/21	3강의실	3 1팀	팀원	106	조아민	17
4	4 AWS 클라우드 활용 자바(Java) Full-Stack 개발자 양성 과정	22/04/04 ~ 22/09/30	4강의실	4 1팀	팀원	104	조예잔	25
5	5 AWS 클라우드와 Elasticsearch를 활용한 Java(자바) Full-Stack 개발자 양성과정	22/05/09 ~ 22/10/21	5강의실	5 1팀	팀원	115	윤민현	8
6	6 빅데이터분석(with 파이썬)과 엘라스틱서치를 활용한 자바(Java) 웹 개발자 양성과정	22/06/07 ~ 22/12/02	6강의실	6 2팀	팀장	109	정영우	26
7	7 SPA 프론트엔드 금융 서비스 개발 실무	22/06/27 ~ 23/01/18	1강의실	7 2팀	팀원	107	윤진찬	20
8	8 플러터(Flutter)로 시작하는 앱개발 종합반	22/08/11 ~ 23/02/09	2강의실	8 2팀	팀원	100	정은민	21
9	9 자바(JAVA), 안드로이드 개발자 양성과정	22/09/26 ~ 23/03/09	3강의실	9 2팀	팀원	94	정예지	5
10	10 빅데이터분석, AWS클라우드 기반 데이터 운영관리 개발자	22/10/06 ~ 23/04/04	4강의실	10 2팀	팀원	110	이아연	13
11	11 JAVA 기반 백엔드, 클라우드 활용 프로젝트	22/11/03 ~ 23/04/14	5강의실	11 3팀	팀장	112	조영선	14
12	12 전자정부표준 프레임워크 기반 풀-스택 개발자 양성과정	22/12/15 ~ 23/06/13	6강의실	12 3팀	팀원	91	박지아	24
13	13 빅데이터를 활용한 인공지능(AI) 기술 구현 엔지니어 양성 과정	23/01/30 ~ 23/07/07	1강의실	13 3팀	팀원	99	정하유	15
14	14 AWS 클라우드 활용 자바(Java) Full-Stack 개발자 양성 과정	23/02/16 ~ 23/07/27	2강의실	14 3팀	팀원	113	김정아	10
15	15 AWS 클라우드와 Elasticsearch를 활용한 Java(자바) Full-Stack 개발자 양성과정	23/03/14 ~ 23/09/08	3강의실	15 3팀	팀원	98	최아연	6
16	16 빅데이터분석(with 파이썬)과 엘라스틱서치를 활용한 자바(Java) 웹 개발자 양성과정	23/04/10 ~ 23/09/19	4강의실	16 4팀	팀장	116	최찬민	1
17	17 자바(JAVA), 안드로이드 개발자 양성과정	23/04/26 ~ 23/11/21	5강의실	17 4팀	팀원	105	박은우	23
18	18 플러터(Flutter)로 시작하는 앱개발 종합반	23/06/19 ~ 23/12/15	6강의실	18 4팀	팀원	101	임선호	4

과목번호	과목명	과목기간	교재명	교사명
1	76 Java	23/02/16 ~ 23/03/17	Java의 정석	윤인성
2	77 Oracle	23/03/20 ~ 23/04/18	오라클로 배우는 데이터베이스 개론과 실습	이현룡
3	78 데이터베이스	23/04/19 ~ 23/05/23	Do it! 오라클로 배우는 데이터베이스 입문	김서윤
4	79 HTML	23/05/24 ~ 23/06/23	한 권으로 끝내는 HTML	윤인성
5	80 AWS 클라우드	23/06/26 ~ 23/07/27	그림으로 이해하는 AWS 구조와 기술	이현룡

# 교육센터 운영 프로젝트

## 구현 쿼리문

```
create or replace procedure procstudentscore (
    pos_SEQ number
)
is
cursor vcursor is
select
st.name as 이름,st.tel as 전화번호,st.complete as 수료여부,
case
    when q.quitdate is null then ' x '
    else '중도탈락 '
end as 중도탈락여부
,sc.attendscore as 출결점수,sc.pilgiscore as 필기점수,sc.silgiscore as 실기점수
from tblStudent st
inner join tblOpenSubject os
on st.oc_seq = os.oc_seq
left outer join tblScore sc
on os.os_Seq = sc.os_Seq and sc.s_Seq = st.s_Seq
left outer join tblQuit q
on st.s_seq = q.s_seq
where os.os_seq = pos_SEQ
order by st.s_seq;

vosname varchar2(300);

begin
    select name into vosname from tblBasicSubject bs
    inner join tblOpenSubject os
    on os.bs_seq = bs.bs_seq
    where os.os_seq = pos_seq;

    dbms_output.put_line('< || vosname || \' 과목 수강 교육생 성적\' || >');
for vrow in vcursor loop
    dbms_output.put_line('||vrow.이름|| \'전화번호: ' ||vrow.전화번호|| ' ||수료여부: '
    ||vrow.수료여부|| ' ||중도탈락여부: ' ||vrow.중도탈락여부|| ' ||출결점수: ' ||vrow.출결점수|| ' ||필기'
    ||vrow.필기점수|| ' ||실기점수: ' ||vrow.실기점수|| ' ||');
end loop;

```

## 수강생 성적 조회

해당과목의 시험을 치룬 학생들의  
성적정보를 출력하는 쿼리문입니다.

< CSS 과목 수강 교육생 성적 >							
이름:	김하우	전화번호:	010-4135-9308	수료여부:	o	중도탈락여부:	x
이름:	김영호	전화번호:	010-2921-9415	수료여부:	o	중도탈락여부:	x
이름:	조민은	전화번호:	010-9945-8126	수료여부:	x	중도탈락여부:	중도탈락
이름:	정아선	전화번호:	010-0548-9417	수료여부:	o	중도탈락여부:	x
이름:	강지솔	전화번호:	010-1359-4420	수료여부:	o	중도탈락여부:	x
이름:	이민아	전화번호:	010-0211-3237	수료여부:	o	중도탈락여부:	x
이름:	임예지	전화번호:	010-9708-5311	수료여부:	o	중도탈락여부:	x
이름:	윤선호	전화번호:	010-3237-7596	수료여부:	o	중도탈락여부:	x
이름:	이선연	전화번호:	010-6556-3036	수료여부:	o	중도탈락여부:	x
이름:	이아솔	전화번호:	010-9303-1230	수료여부:	o	중도탈락여부:	x
이름:	정찬유	전화번호:	010-6586-8483	수료여부:	o	중도탈락여부:	x
이름:	정무선	전화번호:	010-4048-5971	수료여부:	o	중도탈락여부:	x
이름:	최아지	전화번호:	010-6849-4893	수료여부:	o	중도탈락여부:	x
이름:	장선영	전화번호:	010-6567-4505	수료여부:	o	중도탈락여부:	x
이름:	조선민	전화번호:	010-8336-0864	수료여부:	o	중도탈락여부:	x
이름:	조술연	전화번호:	010-4532-2774	수료여부:	o	중도탈락여부:	x
이름:	정지호	전화번호:	010-0774-1051	수료여부:	o	중도탈락여부:	x
이름:	장솔민	전화번호:	010-9047-6001	수료여부:	o	중도탈락여부:	x
이름:	김현오	전화번호:	010-6071-5889	수료여부:	o	중도탈락여부:	x
이름:	정현연	전화번호:	010-9549-7264	수료여부:	o	중도탈락여부:	x
이름:	임하유	전화번호:	010-4016-5449	수료여부:	o	중도탈락여부:	x
이름:	강선선	전화번호:	010-1740-7999	수료여부:	o	중도탈락여부:	x
이름:	장선현	전화번호:	010-0693-3219	수료여부:	o	중도탈락여부:	x
이름:	정선술	전화번호:	010-8165-8813	수료여부:	o	중도탈락여부:	x
이름:	박오하	전화번호:	010-2662-7732	수료여부:	o	중도탈락여부:	x
이름:	박하예	전화번호:	010-4900-5645	수료여부:	o	중도탈락여부:	x
이름:	이예연	전화번호:	010-5545-4476	수료여부:	o	중도탈락여부:	x
이름:	정현영	전화번호:	010-3047-6225	수료여부:	o	중도탈락여부:	x

# 교육센터 운영 프로젝트

## 구현 쿼리문

```
select
    st.name as 이름, st.tel as 전화번호, st.complete as 수료여부,
    case
        when q.quitdate is null then 'x'
        else '중도탈락'
    end as 중도탈락여부
    ,sc.attendscore as 출결점수, sc.pilgiscore as 필기점수, sc.silgiscore as 실기점수
from tblStudent st
    inner join tblOpenSubject os
        on st.oc_seq = os.oc_seq
            left outer join tblScore sc
                on os.os_seq = sc.os_seq and sc.s_seq = st.s_seq
                    left outer join tblQuit q
                        on st.s_seq = q.s_seq
                            where os.os_seq = 10
                                order by st.s_seq;
```

## 수강생 조회

수강 기간이 끝난 학생중 수료하였거나  
중도탈락한 학생의 데이터를 조회하는 쿼  
리문입니다.

#	이름	전화번호	수료여부	중도탈락여부	출결점수	필기점수	실기점수
1	김하우	010-4135-9308	O	X	98	73	89
2	김영호	010-2921-9415	O	X	86	78	82
3	조민은	010-9945-8126	X	중도탈락	(null)	(null)	(null)
4	정아선	010-0548-9417	O	X	96	86	96
5	강지솔	010-1359-4420	O	X	91	70	50
6	이민아	010-0211-3237	O	X	96	72	84
7	임예지	010-9708-5311	O	X	97	95	72
8	윤선호	010-3237-7596	O	X	95	76	86
9	이선연	010-6556-3036	O	X	87	82	75
10	이아솔	010-9303-1230	O	X	98	96	52
11	정찬유	010-6586-8483	O	X	93	91	75
12	정우선	010-4048-5971	O	X	94	97	61
13	최아지	010-6849-4893	O	X	81	86	68
14	장선영	010-6567-4505	O	X	93	88	94
15	조선민	010-8336-0864	O	X	98	80	75
16	조솔연	010-4532-2774	O	X	93	95	63
17	정지호	010-0774-1051	O	X	92	93	72
18	장솔민	010-9047-6001	O	X	85	87	51
19	김현오	010-6071-5889	O	X	84	90	80
20	정현연	010-9549-7264	O	X	81	79	75
21	임하유	010-4016-5449	O	X	82	87	61
22	강선선	010-1740-7999	O	X	92	96	89
23	장선현	010-0693-3219	O	X	93	90	99
24	정선솔	010-8165-8813	O	X	84	97	97
25	박오하	010-2662-7732	O	X	90	99	96
26	박하예	010-4900-5645	O	X	92	89	57
27	이예연	010-5545-4476	O	X	87	90	96
28	정현영	010-3047-6225	O	X	90	72	79
29	김현호	010-5154-2915	O	X	80	85	64
30	조유현	010-1131-0640	O	X	98	99	83



## Java 프로젝트 배달관리 프로그램

- 개발인원 : 4인
- 2023.03.01 ~ 03.12

### 사용기술

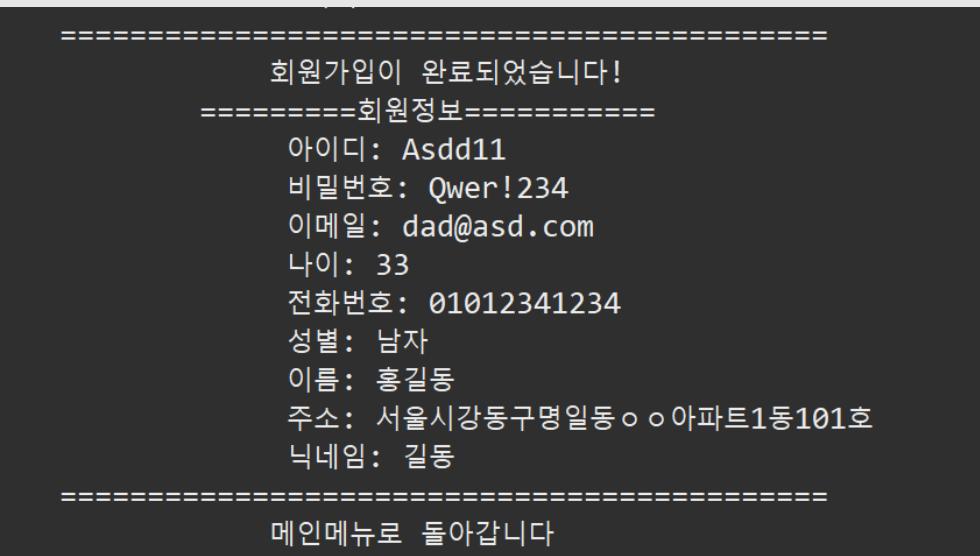
- Java11

### 개발환경

- Eclipse

# 배달 앱 구현 프로젝트

## 구현 기능



```
public class storekeeperJoin {
    final static String TXT = "C:\\class\\code\\java\\DeliveryService\\DeliveryService\\dat\\점주정보.txt";

    /**
     * 점주회원가입 화면출력,회원가입메소드 호출 메소드
     * 점주회원가입에 필요한 매소드들을 호출하여 return받은 값으로 점주데이터파일에 데이터를 추가하는 메소드
     *
     * @see java.io.BufferedReader;
     * @see java.io.FileWriter;
     * @see java.util.Scanner;
     * @see com.test.project.Main;
     * @see com.test.validation.validation;
     * @see validation.IdCheckStorekeeper()
     * @see validation.PasswordCheck()
     * @see validation.AgeCheck()
     * @see validation.TelCheck()
     * @see validation.GenderCheck()
     * @see validation.NameCheck()
     * @see validation.BusinessNumberCheck()
     * @see Main.main()
     * @see
     *
     */
    public static void storeMemRegJoin(){
        Scanner sc = new Scanner (System.in);
        System.out.println("\t\t\t\t\t[점주회원가입]");
        System.out.println("\t\t\t\t\t=====");
        String userId = validation.IdcheckStorekeeper();
        System.out.println("\t\t\t\t\t=====");
        String userPw = validation.PasswordCheck();
        System.out.println("\t\t\t\t\t=====");
        String userAge = validation.AgeCheck();
        System.out.println("\t\t\t\t\t=====");
        String tel = validation.TelCheck();
        System.out.println("\t\t\t\t\t=====");
        String gender = validation.GenderCheck();
        System.out.println("\t\t\t\t\t=====");
        String name = validation.NameCheck();
        System.out.println("\t\t\t\t\t=====");
        String businessNumber = validation.BusinessNumberCheck();
        System.out.println("\t\t\t\t\t=====");
        try {

            FileWriter fw = new FileWriter(TXT, true);
            BufferedWriter writer = new BufferedWriter(fw);

            writer.write(String.format("%s,%s,%s,%s,%s,%s\n",userId,userPw,userAge,tel,gender,name,businessNumber ));
            writer.flush();
            System.out.println();
        }
    }
}
```

## 회원가입 기능

일반회원,점주 회원가입시 필요한 데이터를 입력받고 유효성검사를 통하여 회원정보.txt 파일에 회원정보를 저장하는 기능

# 배달 앱 구현 프로젝트

## 구현 기능

```
/*
 * 매개변수 값이 String형 아이디,비밀번호를 인자값으로 전달받아
 *
 * @param inputId
 * @param inputPw
 * @return
 */
public static String searchAdmin(String inputId, String inputPw) {
    try {
        BufferedReader reader = new BufferedReader(new FileReader(ADMIN));
        String line = null;
        String s = "성공";
        while((line = reader.readLine()) != null) {
            String[] temp = line.split("=");
            if(temp[1].equals(inputId) & temp[2].equals(inputPw)) {
                return s;
            }
        }
        reader.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return null;
}

public static void loginAdminFail() {
    System.out.println();
    System.out.println("=====로그인 실패=====");
    System.out.println("\t로그인 실패");
    System.out.println();
    System.out.println("\t\t메인페이지");
    System.out.println();
    System.out.println("=====번호 입력=====");
    System.out.println();
    System.out.print("\t\t 번호 입력: ");

    Scanner scan = new Scanner(System.in);
    String input = scan.nextLine();

    if (input.equals("0")) {
        UI.mainUi();
    }
}
```

## 로그인 기능

아이디와 패스워드를 스캐너를 통해 매개 변수로 받아 데이터가 일치시 로그인하고 불일치시 메인페이지로 이동하도록 구현

```
=====
▽관리자 로그인

아이디 : admin001

비밀번호 : 45jY1lsKm

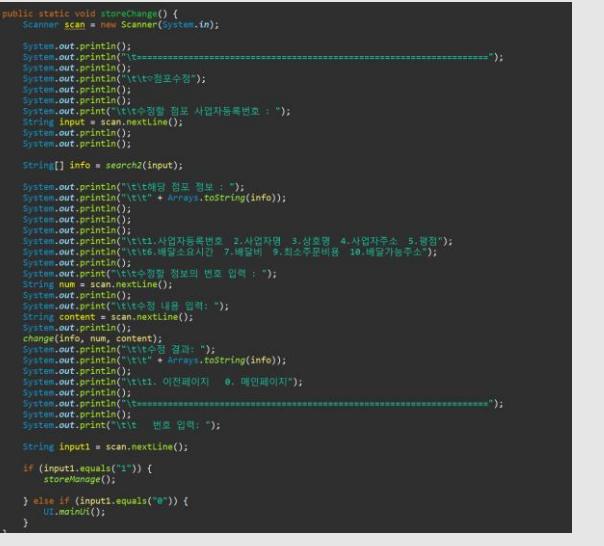
=====
▽관리자 메인메뉴

1.점포관리 2.회원관리 3.이벤트관리 0.메인페이지

=====
번호 입력:
```

# 배달 앱 구현 프로젝트

## 구현 기능



```
public static void storeChange() {
    Scanner scan = new Scanner(System.in);

    System.out.println();
    System.out.println("=====점포수정=====");
    System.out.println();
    System.out.println("1.점포 조회 2.점포수정 3.점포삭제 0.메인페이지");
    System.out.println();
    System.out.println();
    System.out.print("원하시는 기능을 선택하세요 : ");
    String input = scan.nextLine();
    System.out.println();
    System.out.println();
    System.out.println();

    String[] info = search2(input);

    System.out.println("점포 정보 : ");
    System.out.println("titl" + Arrays.toString(info));
    System.out.println();
    System.out.println();
    System.out.println();
    System.out.println();
    System.out.println("titl 사업자등록번호 2.사업자명 3.상호명 4.사업자주소 5.평점");
    System.out.println("titl6.배달소시간 7.배달비 9.최소주문비용 10.배달가능주소");
    System.out.println();
    System.out.println();
    System.out.println("titl 현재 정보의 번호 입력 : ");
    String num = scan.nextLine();
    System.out.println();
    System.out.println();
    System.out.println("수정할 점포 내용 입력 : ");
    String content = scan.nextLine();
    System.out.println();
    change(info, num, content);
    System.out.println("수정 결과 : ");
    System.out.println("titl" + Arrays.toString(info));
    System.out.println();
    System.out.println();
    System.out.println("titl 이전페이지 0.메인페이지");
    System.out.println();
    System.out.println();
    System.out.print("titl 번호 입력 : ");
    String input1 = scan.nextLine();

    if (input1.equals("1")) {
        storeManage();
    } else if (input1.equals("0")) {
        UI.main();
    }
}
```

```
=====점포관리=====

1.점포조회 2.점포수정 3.점포삭제 0.메인페이지
=====

번호 입력: 2

=====

▽점포수정

수정할 점포 사업자등록번호 : 423-60-45421

해당 점포 정보 :
[423-60-45421, 유재희, 김밥천국, 서울시 금천구 신사동 321-311, 페스트푸드, 2.8, 25, 44, 1000, 10000, [강동구]]

1.사업자등록번호 2.사업자명 3.상호명 4.사업자주소 5.평점
6.배달소시간 7.배달비 9.최소주문비용 10.배달가능주소

수정할 정보의 번호 입력 : 1

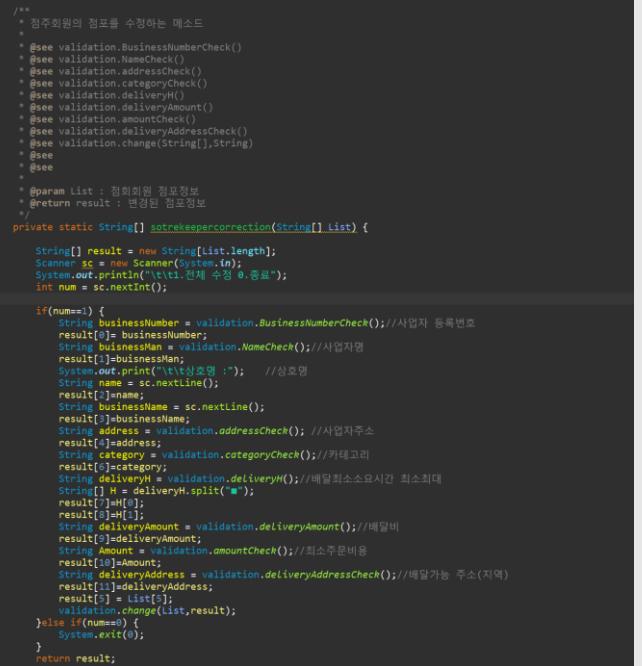
수정 내용 입력: 123-12-12345

수정 결과:
[123-12-12345, 유재희, 김밥천국, 서울시 금천구 신사동 321-311, 페스트푸드, 2.8, 25, 44, 1000, 10000, [강동구]]

1.이전페이지 0.메인페이지

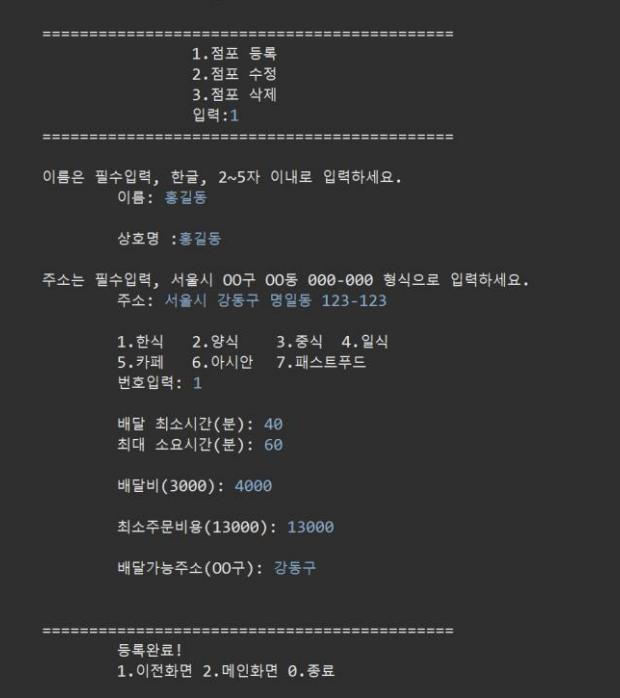
=====

번호 입력:
```



```
/*
 * 점주회원의 정보를 수정하는 메소드
 *
 * @see validation.BusinessNumberCheck()
 * @see validation.NameCheck()
 * @see validation.AddressCheck()
 * @see validation.CategoryCheck()
 * @see validation.Delivery()
 * @see validation.DeliveryAmount()
 * @see validation.AmountCheck()
 * @see validation.Change()
 * @see validation.DeliveryAddressCheck()
 * @see validation.Change(String[],String)
 * @see ...
 */
private static String[] setRekeepercorrection(String[] list) {
    String[] result = new String[list.length];
    Scanner sc = new Scanner(System.in);
    System.out.println("titl.점포 수정 0.종료");
    int num = sc.nextInt();

    if(num==0) {
        String businessNumber = validation.BusinessNumberCheck(); //사업자 등록번호
        result[0]=businessNumber;
        String businessMan = validation.NameCheck(); //사업자명
        result[1]=businessMan;
        System.out.print("titl상호명 :"); //상호명
        String name = sc.nextLine();
        result[2]=name;
        String businessName = sc.nextLine();
        result[3]=businessName;
        String address = validation.AddressCheck(); //사업자주소
        result[4]=address;
        String category = validation.CategoryCheck(); //카테고리
        result[5]=category;
        String deliveryH = validation.DeliveryH(); //배달최소소요시간 최소최대
        String[] H = deliveryH.split("=");
        result[6]=H[0];
        result[7]=H[1];
        String deliveryAmount = validation.DeliveryAmount(); //배달비
        result[8]=deliveryAmount;
        String amount = validation.AmountCheck(); //최소주문비용
        result[9]=amount;
        String deliveryAddress = validation.DeliveryAddressCheck(); //배달가능 주소(지역)
        result[10]=deliveryAddress;
        result[11]=List[5];
        validation.Change(list,result);
    }else if(num==0) {
        System.exit(0);
    }
    return result;
}
```



```
=====
1.점포 등록
2.점포 수정
3.점포 삭제
입력:1

이름은 필수입력, 한글, 2~5자 이내로 입력하세요.
이름: 흥길동

상호명 :흥길동

주소는 필수입력, 서울시 00구 00동 000-000 형식으로 입력하세요.
주소: 서울시 강동구 명일동 123-123

1.한식 2.양식 3.중식 4.일식
5.카페 6.아시안 7.페스트푸드
번호입력: 1

배달 최소시간(분): 40
최대 소요시간(분): 60

배달비(3000): 4000

최소주문비용(13000): 13000

배달가능주소(00구): 강동구

=====

등록완료!
1.이전화면 2.메인화면 0.종료
```

## 점포관리 기능

점주회원에 한하여 점포를 관리하는 기능 구현 점포등록에 필요한 데이터를 입력아 등록하는 기능과 등록된 점포 수정,삭제기능 구현