

# 아파트통합관리시스템

## AnyApart

### 주제선정배경

우리 생활과 밀접한 프로그램을 만들고자 하던 중, 우리나라 인구의 절반이 아파트에 거주하고 있다는 기사를 접하게 되었습니다. 기존 아파트 관리시스템을 이용하기 위해서는 별도의 프로그램을 설치해야 했기 때문에 사용자로서는 다소 번거로운 부분이 있었습니다.

이러한 개선점을 보완하기 위하여 우리 프로젝트에서는 SaaS(Software-as-a-Service)를 지향, 별도의 설치나 관리의 부담 없이 클라우드를 통해 이용할 수 있는 아파트 관리시스템을 만들고자 하였습니다.



### 사이트



# 로그인 - 로그인, 비밀번호 변경

## Log In

아이디 ⓘ

Password

☐ 아이디 기억하기 ☐ Remember me

SIGN IN

[아이디 찾기](#) | [비밀번호 찾기](#)

비밀번호 변경

## 계정설정

현재 비밀번호

\*\*비밀번호 변경시 영문자, 숫자, 특수문자 포함해 8~12 까지 가능합니다.

새 비밀번호

비밀번호 확인

변경

```
<beans:bean id="authenticationProvider"
  class="org.springframework.security.authentication.dao.DaoAuthenticationProvider"
  p:userDetailsService-ref="customUserService"
  p:passwordEncoder-ref="passwordEncoder"
  p:hideUserNotFoundExceptions="false" />
```

```
<form-login login-page="/login"
  username-parameter="memId" password-parameter="memPass"
  login-processing-url="/login/loginProcess.do"
  authentication-success-handler-ref="customAuthenticationSuccessHandler"
  authentication-failure-url="/login" />
```

회원 권한 메인 페이지 이동

## Login-success-handler

```
public class CustomAuthenticationSuccessHandler extends SavedRequestAwareAuthenticationSuccessHandler{

    @Inject
    private IAuthDAO dao;

    @Override
    protected void handle(HttpServletRequest request, HttpServletResponse response, Authentication authentication)
        throws IOException, ServletException {
        String mem_id = authentication.getName();

        MemberVO memberVO = dao.authMember(mem_id);

        WebUtils.setSessionAttribute(request, "memId", memberVO.getMemId());
        String tmp = (String) WebUtils.getSessionAttribute(request, "memId");

        if(authentication.getAuthorities().contains(new SimpleGrantedAuthority("ROLE_RES"))){
            setDefaultTargetUrl("/resident");
        }else if(authentication.getAuthorities().contains(new SimpleGrantedAuthority("ROLE_ADMIN"))){
            setDefaultTargetUrl("/vendor");
        }else {
            setDefaultTargetUrl("/office");
        }

        super.handle(request, response, authentication);
    }
}
```

# 로그인 – Remember me

로그인 정보 기억하기

## Log In

아이디 ⓘ

Password

☐ 아이디 기억하기 ☐ Remember me

SIGN IN

[아이디 찾기](#) | [비밀번호 찾기](#)

```
<remember-me remember-me-parameter="rememberMe"
remember-me-cookie="rememberMeCookie" data-source-ref="dataSource"
user-service-ref="customUserService" />

<intercept-url pattern="/login/**" access="permitAll" />

<intercept-url pattern="/office/setting/**" access="hasRole('ROLE_HEAD')"/>
<intercept-url pattern="/vendor/**" access="hasRole('ROLE_ADMIN')"/>
<intercept-url pattern="/resident/**" access="hasRole('ROLE_RES')"/>
<intercept-url pattern="/**" access="permitAll" />
<access-denied-handler error-page="/AccessDenied.do" />
```

DB 모델

TEAM2_202007F.PERSISTENT_LOGINS	
	USERNAME VARCHAR2 (64 BYTE)
P	SERIES VARCHAR2 (64 BYTE)
	TOKEN VARCHAR2 (64 BYTE)
	LAST_USED TIMESTAMP
🔑	PERSISTENT_LOGINS_PK (SERIES)
🔑	PERSISTENT_LOGINS_PK (SERIES)

PERSISTAENT\_LOGINS 테이블의 정보를 가지고 인증 토큰을 생성해줍니다.

## 로그인 - ID, PW 찾기



ANYAPART

함께 사는 즐거움, 애니아파트

아이디 찾기

\*\* 본인확인 이메일과 입력한 이메일과 일치하면 인증번호를 받을 수 있습니다.

이름

이메일

인증번호 받기

애니 아파트



ANYAPART

함께 사는 즐거움, 애니아파트

비밀번호 찾기

\*\* ID 및 본인확인 이메일과 일치하면 인증번호를 받을 수 있습니다.

아이디

이메일

인증번호 받기

☆ AnyApart ID/PW 찾기 인증번호

보낸사람 VIP <hkjs9620@gmail.com>

받는사람 <tlsgks678@naver.com>

인증번호 : 301823

## 메일 API와 구글 SMTP 서버이용

```
public String mailSend(MemberVO member) {
    ServerSpec mailServer = ServerKind.GMAIL.getSmtp();
    String username = "hkjs9620";
    String password = "gqhqpumkxhgfdke";
    String to = member.getMemEmail();
    String from = "AnyApart";

    // 1) MailSender 기본 설정
    JavaMailSenderImpl mailSender = new JavaMailSenderImpl();
    mailSender.setHost(mailServer.getHost());
    mailSender.setPort(mailServer.getPort());
    mailSender.setUsername(username);
    mailSender.setPassword(password);
    Properties props = new Properties();
    props.put("mail.smtp.starttls.enable", true);
    mailSender.setJavaMailProperties(props);

    String certNum = RandomStringUtils.randomNumeric(6);
    // 2) SimpleMailMessage를 이용한 메시지 설정
    SimpleMailMessage mail = new SimpleMailMessage();
    mail.setTo(to);
    mail.setFrom(from);
    mail.setSubject("AnyApart ID/PW 찾기 인증번호");
    mail.setText("인증번호 : " + certNum);

    // 3) 전송
    mailSender.send(mail);
    System.out.println("전송 완료");
    return certNum;
}
```

# 민감정보 암호화 – Jasypt 이용

## Pom.xml dependency

```
<!-- dbInfoProperties 암호화하기 위함 -->
<!-- https://mvnrepository.com/artifact/org.jasypt/jasypt-spring31 -->
<dependency>
    <groupId>org.jasypt</groupId>
    <artifactId>jasypt-spring31</artifactId>
    <version>1.9.2</version>
    <scope>compile</scope>
</dependency>
```

## Bean 설정

```
<bean id="encryptorConfig" class="org.jasypt.encryption.pbe.config.EnvironmentPBEConfig">
    <property name="algorithm" value="PBEWithMD5AndDES" />
    <property name="password" value="java" />
</bean>

<bean id="encryptor" class="org.jasypt.encryption.pbe.StandardPBEStringEncryptor">
    <property name="config" ref="encryptorConfig" />
</bean>

<bean class="org.jasypt.spring31.properties.EncryptablePropertyPlaceholderConfigurer">
    <constructor-arg ref="encryptor" />
    <property name="locations" value="classpath:kr/or/anyapart/db/dbInfo.properties">
    </property>
</bean>

<bean id="dataSource" class="org.apache.commons.dbcp2.BasicDataSource"
    p:driverClassName="${maindb.driverClassName}"
    p:url="${maindb.url}"
    p:username="${maindb.user}"
    p:password="${maindb.password}"
    p:initialSize="${maindb.initialSize}"
    p:maxTotal="${maindb.maxActive}"
    p:maxWaitMillis="${maindb.maxWait}"
/>
```

## dbInfo.properties

```
maindb.url=ENC(VXjGUQ7OB+pWA7AMmLIAXggp/nJDp9CfEAJMqe9V6RyaoQxwSpJeT63tvcIUpr18fU8zKLBsFkyNByTj6qtdKQ==)
maindb.user=ENC(o432phE0oMiZkAIitzKuK0lJm3nb4cex2)
maindb.password=ENC(8kY+ajL9VAI8rqIutdXeIw==)
```

# 물품 관리 - 물품 조회

물품관리

Q

검색 조건

분류선택

전체

검색

초기화

\* 클릭시 상세 조회가 가능합니다.

10 개 씩 보기

등록

No.	물품등록번호	물품분류	물품명	제조사	가격	수량 (개)
14	A0024P014	사무용품	검정유성매직	모나미	7,500 원	12
13	A0024P013	사무용품	커터칼	피스코리아	350 원	10
12	A0024P012	사무용품	검정네임펜1다스	모나미	8,700 원	10
11	A0024P011	청소용품	청소기	다이슨	489,000 원	2
10	A0024P010	사무용품	파란볼펜	모나미	300 원	50
9	A0024P009	사무용품	빨간볼펜	모나미	300 원	50
8	A0024P008	사무용품	빨간연필	동아문구	500 원	50
7	A0024P007	사무용품	검정볼펜	모나미	300 원	50
6	A0024P006	사무용품	A4 500매	더블에이	6,000 원	98

클릭

물품 상세 정보

* 물품코드	A0001P004	* 물품분류	모니터
* 물품명	LG모니터 27인치	* 제조사/구매처	LG전자
* 가격(원)	248,000원	* 수량(개)	6

구매/사용 내역

No.	일자	수량(개)	구분
12	2021/01/02	6	구매

수리 이력

No.	일자	수리내용	수리비
1	2021/01/04	화소불량 제품교환	6,000 원
19	2021/02/19	모니터 화소 불량 수리	10,000 원
18	2021/02/19	모니터 화소 불량 수리	10,000 원
7	2021/01/04	화소불량 제품교환	6,000 원
6	2021/01/04	화소불량 제품교환	5,000 원
5	2021/01/04	화소불량 제품교환	5,000 원

닫기

클릭 시 해당 물품 정보의 상세보기를 모달 창으로 보여줍니다.



# 물품 관리 - 물품 등록

여러 개 입력

물품등록

시연용 추가하기

추가하기

물품분류코드	물품명	가격	제조사	수량 (개)	삭제
사무용품 ▾	검정네임펜1다:	8700	모나미	10	
사무용품 ▾	커터칼	350	피스코리아	10	-
사무용품 ▾	검정유성매직	7500	모나미	12	-

저장

닫기

추가/제거 하기

```
$("#plusBtn").on("click", function(){
// let clickTr = $('#insertArea').find('tr').eq(0);
let clickTr = $('#insertArea').find('tr').last();
let newTr = clickTr.clone();
let childInput = newTr.find(":input");
let name;
$(childInput).each(function(idx,element){
// 각 input 값을 찾아서 name을 변경
name = parseInt(element.name.substring(9,10));
name = String(name + 1);
str = element.name.replaceAt(9, name);
element.setAttribute("name",str);
});
let minus = newTr.find(".minusBtn");
$(minus).show();
let inputTag = newTr.find(":input");
inputTag.val("");
clickTr.after(newTr);

newTr.find("input[type=hidden]").val("${pagingVO.searchVO.searchAptCode }");
});

$("#insertArea").on("click", ".minusBtn", function(){
let deleteTr = $(this).parent().parent();
deleteTr.prev().find(".plusBtn").show();
deleteTr.detach();
});
});
```

```
<tbody id="insertArea" class="tbody-light tbodyScroll">
  <tr>
    <td>
      <input type="hidden" name="prodList[0].aptCode" value="${pagingVO.searchVO.searchAptCode }"/>
      <select name="prodList[0].prodCode" class="custom-select md-6 prodCode">
        <option value>전체</option>
      </select>
    </td>
    <td><input name="prodList[0].prodName" size="12" type="text"></td>
    <td><input name="prodList[0].prodPrice" size="12" type="number" onkeypress="return onlyNumber(event)" min=
    <td><input name="prodList[0].prodCompany" size="12" type="text"></td>
    <td><input name="prodList[0].prodQty" size="12" type="number" onkeypress="return onlyNumber(event)" min=
    <td class="last now"><span class='btn btn-danger minusBtn'>-</span></td>
  </tr>
</tbody>
```

# 물품 관리 – 수리 이력, 구매/사용내역

클릭 시 변경가능 상태

수리 이력

🔍 검색 조건

분류선택 

전체

물품선택 

전체

연도-월-일

연도-월-일

검색

초기화

10

개 씩 보기

등록

No.

수리일자

물품명

수리내용

수리비 (원)

삭제

10

2021-03-09

LG 30인치 모니터

모니터 다리 교체함

10,000 원

삭제

9

2021-03-09

HP 오피스젯 7110

배터리 무상 교체

10,000 원

삭제

8

2021-03-09

청소기

배터리 교체

30,000 원

삭제

7

2021-03-09

LG 30인치 모니터

화면 연결구 손상

10,000 원

삭제

6

2021-03-09

HP 오피스젯 7110

뒷개 부서짐

0 원

삭제

5

2021-03-09

사무용데스크탑

CPU 접촉 불량

10,000 원

삭제

사용 · 구매 이력

🔍 검색조건

분류선택 

전체

구매/사용구분 

구분

연도-월-일

연도-월-일

검색

초기화

10

개 씩 보기

등록

No.

일자

물품등록번호

물품분류

물품명

구분

수량 (개)

삭제

18

2021-03-12

A0024P014

사무용품

검정유성매직

구매

12

삭제

17

2021-03-12

A0024P013

사무용품

커터칼

구매

10

삭제

16

2021-03-12

A0024P012

사무용품

검정네임펜1다스

구매

10

삭제

구매/사용 내역 등록

추가하기

물품선택

사용/구매일자

수량

내역구분

추가

선택하세요

연도-월-일

전체

저장

닫기

수리이력등록

\* 물품분류

분류 목록

\* 물품명

물품 목록

\* 수리일자

연도-월-일

\* 수리비

원

\* 수리내용

저장

초기화

닫기



Q&A 게시판

벤더사이트 Q&A

19	[기능]결재 알림창에 추가 기능넣어주세요	대덕아트힐	2021-03-10	33	<div>답글</div> <div>삭제</div>
18	[기능]급여 항목은 어떻게 보나요? <div>답변</div>	대덕아트힐	2021-03-10	29	<div>삭제</div>
17	↳ re: [기능]급여 항목	애니아파트	2021-03-10	28	<div>수정</div> <div>삭제</div>
16	[기능]단지설정메뉴는 무엇인가요? <div>답변</div>	대덕아트힐	2021-03-10	31	<div>삭제</div>
15	↳ re: [기능]단지 설정 메뉴..	애니아파트	2021-03-10	33	<div>수정</div> <div>삭제</div>

관리사무소 사이트 Q&A

No.	제목	작성자	작성일	조회수
12	[일반]관리소장 퇴사시 권한 양도는 어떻게 하나요?	김동수	2021-03-23	1
11	[기능]결재 알림창에 추가 기능넣어주세요	이경륜	2021-03-10	33
10	[기능]급여 항목은 어떻게 보나요? <div>답변</div>	이경륜	2021-03-10	29
9	↳ re: [기능]급여 항목	애니아파트	2021-03-10	28
8	[일반]검색은 어떻게 이용하나요?	이경륜	2021-03-10	40

계층형 게시글 구조

```
<select id="selectOfficeQnalist" parameterType="kr.or.anyapart.vo.PagingVO" resultType="kr.or.anyapart.vo.QnaList">
    <include refid="apartBoard"/>
    SELECT B.*
    FROM (
        SELECT A.*, ROWNUM RNUM
        FROM (
            SELECT
                C.BO_NO, SUBSTR(CODE_NAME,5) AS BO_TYPE, C.BO_TITLE,
                C.MEM_NICK AS BO_WRITER, C.BO_CONTENT, TO_CHAR(C.BO_DATE, 'YYYY-MM-DD')BO_DATE,
                C.BO_HIT, C.BO_PARENT, C.BO_DEPTH
                , E.BO_NO AS ANSWER_FLAG , C.APT_NAME
            FROM APARTBOARD C JOIN COMMONCODE D ON ( C.BO_TYPE = D.CODE_ID)
                LEFT OUTER JOIN BOARD E ON (C.BO_NO = E.BO_PARENT)
            <include refid="searchFrag"/> <!-- 이거 동적으로 변경되어야하는 부분 -->
            <include refid="unansweredFrag"/>
            START WITH C.BO_PARENT IS NULL
            CONNECT BY PRIOR C.BO_NO = C.BO_PARENT <!-- 여기는 필수 -->
            ORDER SIBLINGS BY C.BO_NO DESC
        ) A
    ) B
    <![CDATA[
        WHERE RNUM >= #{startRow} AND RNUM <= #{endRow}
    ]]>
</select>
```

관리사무소 사이트 게시글 상세 조회

Q&A

글번호	341	작성자	김동수
조회수	2	작성일	2021-03-23
제목	관리소장 퇴사시 권한 양도는 어떻게 하나요?		
내용	관리 소장 퇴사시 권한 양도를 어떻게 하는지 알고 싶습니다.		
첨부파일			

목록

벤더 사이트 게시글 상세 조회

단지설정메뉴는 무엇인가요?

대덕아트힐

2021-03-10

조회수:31

단지 설정 메뉴는 무엇인가요?

삭제

목록

## 시설 - 시설 예약

### 입주민사이트 시설 예약

\* 시설 선택시 예약된 일정들이 조회됩니다. 시설명 푸항항 줌바댄스 시설 정보 조회하기

« < > » TODAY 2021년 3월 11일 DAY

	목요일
오전 9시	오전 9:00 - 오전 9:30 - 푸항항 줌바댄스 : 5명
	오전 9:30 - 오전 10:00 - 푸항항 줌바댄스 : 5명
오전 10시	오전 10:00 - 오전 10:30 - 푸항항 줌바댄스 : 5명
	오전 10:30 - 오전 11:00 - 푸항항 줌바댄스 : 5명
오전 11시	
오후 12시	
오후 1시	

원하는 시간대 드래그

### 시설 예약 하기

시설예약  
\* 예약은 하루전까지 가능하며, 수용 인원이 초과시 예약이 실패할 수 있습니다.

시설명	푸항항 줌바댄스		
예약자명	박정민	예약일	2021-04-01
핸드폰 번호	01024516235	예약시간	09:00 - 12:00
예약 인원	5		

예약 하기 닫기

원하는 시간대를 드래그하면 모달 창으로 예약 정보가 넘어갑니다.

# 시설 - 시설 예약 프로시저

```
create or replace procedure p_reservation(
    p_resv_date IN varchar2, p_resv_start IN reservation.resv_start%type,    p_resv_end IN reservation.resv_end%type,
    p_resv_cnt IN reservation.resv_cnt%type,    p_cmnt_no IN COMMUNITY.CMNT_NO%type,    p_mem_id IN member.mem_id%type
    , p_resv_no out number    -- 성공하면 예약 번호를 내보냄,
)
is
    v_resv_no reservation.resv_no%type := 0;    -- 예약 상세 변수
    v_apt_code apart.apt_code%type := NULL; -- 매개변수로 받은 멤버 값에서 아파트 코드를 추출해 넣어준다.
    val number := 0;
    capa number := 0;
    V_STARTFLAG NUMBER := 0;    -- 30분에서 시작하는 지 정각에서 시작하는지 확인하는 플래그 변수
    V_ENDFLAG NUMBER := 0;    -- 30분으로 끝나는지 정각에서 끝나는지 확인하는 플래그 변수
    V_TIME NUMBER := 0;    -- 시각의 차이 * 2 를 담고 있을 변수
    V_CNT NUMBER := 0;    -- 반복하는 숫자를 담고 있을 변수
    V_CHANGE NUMBER := 0;    -- 계산식을 변경할 녀석
    V_START_TIME VARCHAR2(4) := '00';    -- 시작하는 시간을 담고 있는다.
    V_CHECK number := -1;    -- 예약이 가능한지 불가능 한지 체크하는 녀석
begin
    /*
    (끝 시각 - 시작 시각) * 2 -1(30분에 시작시) +1(30분에 끝날시)
    */
    SELECT CMNT_CAPA INTO capa FROM COMMUNITY WHERE CMNT_NO = P_cmnt_no; -- 최대 수용인원을 가져온다.

    -- 조회할 시작 시간과, 끝나는 시간을 확인해줘야한다. 변수를 만들어서 담아준다.
    -- 반복될 횟수도 계산해서 가지고 있어야한다.

    IF SUBSTR(p_resv_start,3,4) = '30' THEN
        V_STARTFLAG := -1;
        -- 시작시간 30분이면 -1
        V_CHANGE := 1;
    ELSIF SUBSTR(p_resv_end,3,4) = '30' THEN
        V_ENDFLAG := -1;
        -- 끝나는 시간 30분이면 -1
    END IF;

    V_TIME := ( TO_NUMBER(SUBSTR(p_resv_end,1,2)) - TO_NUMBER(SUBSTR(p_resv_start,1,2)) ) * 2 ; -- 시각 차이 * 2 를 담음.

    -- 반복을 몇번이나 할 것인지 정함?
    IF V_STARTFLAG < 0 THEN
        V_TIME := V_TIME - 1;
    ELSIF V_ENDFLAG < 0 THEN
        V_TIME := V_TIME + 1;
    END IF;

    V_START_TIME := substr(p_resv_start,1,2);    -- 시작 시간을 담아준다.

    V_APT_CODE := substr(p_mem_id,1,5);    -- 반복문 시작전에 아파트 코드 넣어주기

    loop
        EXIT WHEN V_TIME = V_CNT; -- 루프 다 돌았으면 나가기

        -- 30분 시작인지 정각 시작인지 정한다.
        IF V_CHANGE = 1 THEN
            SELECT NVL(MAX(A.RESV_CNT),0) AS RESV_CNT
            into val
            FROM( SELECT
                    B.CMNT_NO
                    , C.CMNT_NAME
                    , DETAIL_START AS RESV_START
                    , DETAIL_END AS RESV_END
                    , SUM(RESV_CNT) AS RESV_CNT
                FROM
                    RESVDETAIL A LEFT OUTER JOIN RESERVATION B ON (A.RESV_NO = B.RESV_NO)
                    LEFT OUTER JOIN COMMUNITY C  ON (B.CMNT_NO = C.CMNT_NO)
                WHERE C.APT_CODE = v_apt_code
                    and c.cmnt_no = p_cmnt_no
                GROUP BY B.CMNT_NO, C.CMNT_NAME, DETAIL_START, DETAIL_END
            ) A LEFT OUTER JOIN COMMUNITY B  ON (A.CMNT_NO = B.CMNT_NO)

            WHERE A.RESV_CNT = B.CMNT_CAPA
            and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'00'
            and RESV_END = p_resv_date||'T'||V_START_TIME||':'||'30';
            -- 30분 간격으로 인원을 조회한다.

            V_CHECK := -1;
            EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
            V_CHECK := 1;

            V_CHANGE := 1;

        END IF;
        V_CNT := V_CNT+1;
    end loop;

    p_resv_no := -1;
    -- 실패시 들어갈 값
    IF V_CHECK > 0 THEN
        SELECT MAX(RESV_NO)+1 AS RESV_NO INTO v_resv_no FROM RESERVATION;

        INSERT INTO reservation (
            resv_no,
            resv_date,
            resv_start,
            resv_end,
            resv_cnt,
            cmnt_no,
            mem_id
        ) VALUES (
            v_resv_no,
            TO_DATE(P_RESV_DATE,'YYYY/MM/DD'),
            P_RESV_START,
            P_RESV_END,
            P_RESV_CNT,
            P_CMNT_NO,
            P_MEM_ID
        );
        p_resv_no := V_RESV_NO;
        -- 인서트 성공시 해당 예약번호 반환
    END IF;
end;
```

```
WHERE A.RESV_CNT = B.CMNT_CAPA
    and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'30'
    and RESV_END = p_resv_date||'T'||LPAD(TO_NUMBER(V_START_TIME)+1,2,'0')||':'||'00';
    -- 30분 간격으로 인원을 조회한다.

    V_CHECK := -1;
    EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
    V_CHECK := 1;

ELSE
    SELECT NVL(MAX(A.RESV_CNT),0) AS RESV_CNT
    into val
    FROM( SELECT
            B.CMNT_NO
            , C.CMNT_NAME
            , DETAIL_START AS RESV_START
            , DETAIL_END AS RESV_END
            , SUM(RESV_CNT) RESV_CNT
        FROM
            RESVDETAIL A LEFT OUTER JOIN RESERVATION B ON (A.RESV_NO = B.RESV_NO)
            LEFT OUTER JOIN COMMUNITY C  ON (B.CMNT_NO = C.CMNT_NO)
        WHERE C.APT_CODE = v_apt_code
            and c.cmnt_no = p_cmnt_no
        GROUP BY B.CMNT_NO, C.CMNT_NAME, DETAIL_START, DETAIL_END
    ) A LEFT OUTER JOIN COMMUNITY B  ON (A.CMNT_NO = B.CMNT_NO)
    WHERE RESV_CNT = B.CMNT_CAPA
    and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'00'
    and RESV_END = p_resv_date||'T'||V_START_TIME||':'||'30';
    -- 30분 간격으로 인원을 조회한다.

    V_CHECK := -1;
    EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
    V_CHECK := 1;

    V_CHANGE := 1;

END IF;
V_CNT := V_CNT+1;
end loop;

p_resv_no := -1;
-- 실패시 들어갈 값
IF V_CHECK > 0 THEN
    SELECT MAX(RESV_NO)+1 AS RESV_NO INTO v_resv_no FROM RESERVATION;

    INSERT INTO reservation (
        resv_no,
        resv_date,
        resv_start,
        resv_end,
        resv_cnt,
        cmnt_no,
        mem_id
    ) VALUES (
        v_resv_no,
        TO_DATE(P_RESV_DATE,'YYYY/MM/DD'),
        P_RESV_START,
        P_RESV_END,
        P_RESV_CNT,
        P_CMNT_NO,
        P_MEM_ID
    );
    p_resv_no := V_RESV_NO;
    -- 인서트 성공시 해당 예약번호 반환
END IF;
end;
```

# 시설 - 시설 예약 세부내역 프로시저

```
create or replace procedure p_reservation_detail(
  p_resv_date in varchar2, -- 예약 날짜
  p_resv_start in varchar2, -- 예약 시작 시간
  p_resv_end in varchar2, -- 예약 끝나는 시간
  P_cmnt_no in number, -- 시설 번호
  p_resv_cnt in number, -- 예약인원
  p_resv_no in number, -- 예약 번호
  P_APT_CODE IN VARCHAR2 -- 아파트 코드
,p_result out number -- 성공했는지 못했는지 내보낼 결과값 양수면 예약이 성공한 것
)
is
  val number :=0 ;
  capa number := 0;
  detailNo number := 0;
  V_STARTFLAG NUMBER := 0;   -- 30분에서 시작하는 지 정각에서 시작하는지 확인하는 플래그 변수
  V_ENDFLAG NUMBER := 0;     -- 30분으로 끝나는지 정각에서 끝나는지 확인하는 플래그 변수
  V_TIME NUMBER := 0;        -- 시각의 차이 * 2 를 담고 있을 변수
  V_CNT NUMBER := 0;         -- 반복하는 숫자를 담고 있을 변수
  V_CHANGE NUMBER := 0;      -- 계산식을 변경할 녀석
  V_START_TIME VARCHAR2(4) := '00'; -- 시작하는 시간을 담고 있다.
begin
  /*
  (끝 시각 - 시작 시각) * 2 -1(30분에 시작시) +1(30분에 끝날시)
  */
  SELECT CMNT_CAPA INTO capa FROM COMMUNITY WHERE CMNT_NO = P_cmnt_no; -- 최대 수용인원을 가져온다.

  -- 조회할 시작 시간과, 끝나는 시간을 확인해줘야한다. 변수를 만들어서 담아준다.
  -- 반복될 횟수도 계산해서 가지고 있어야한다.

  IF SUBSTR(p_resv_start,3,4) = '30' THEN
    V_STARTFLAG := -1;      -- 30분이면 -1
    V_CHANGE := 1;
  ELSIF SUBSTR(p_resv_end,3,4) = '30' THEN
    V_ENDFLAG := -1;        -- 30분이면 -1
  END IF;

  V_TIME := ( TO_NUMBER(SUBSTR(p_resv_end,1,2)) - TO_NUMBER(SUBSTR(p_resv_START,1,2)) ) * 2 ; -- 시각 차이 * 2 를 담음.

  -- 반복을 몇번이나 할 것인지 정함?
  IF V_STARTFLAG < 0 THEN
    V_TIME := V_TIME - 1;
  ELSIF V_ENDFLAG < 0 THEN
    V_TIME := V_TIME + 1;
  END IF;

  V_START_TIME := substr(p_resv_start,1,2); -- 시작 시간을 담아준다.

  loop
    EXIT WHEN V_TIME = V_CNT; -- 루프 다 돌았으면 나가기

    -- 30분 시작인지 정각 시작인지 정한다.
    IF V_CHANGE = 1 THEN
      SELECT NVL(MAX(A.RESV_CNT),0) AS RESV_CNT
      into val
      FROM( SELECT
        B.CMNT_NO
        , C.CMNT_NAME
        , DETAIL_START AS RESV_START
        , DETAIL_END AS RESV_END
        , SUM(RESV_CNT) AS RESV_CNT
      FROM
        RESVDETAIL A LEFT OUTER JOIN RESERVATION B ON (A.RESV_NO = B.RESV_NO)
        LEFT OUTER JOIN COMMUNITY C ON (B.CMNT_NO = C.CMNT_NO)
      WHERE C.APT_CODE = P_APT_CODE
        and c.cmnt_no = P_cmnt_no
      GROUP BY B.CMNT_NO, C.CMNT_NAME, DETAIL_START, DETAIL_END
      ) A LEFT OUTER JOIN COMMUNITY B ON (A.CMNT_NO = B.CMNT_NO)
    WHERE RESV_CNT = B.CMNT_CAPA
      and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'00'
      and RESV_END = p_resv_date||'T'||V_START_TIME||':'||'30';
    -- 30분 간격으로 인원을 조회한다.

    p_result := val; -- 실패라고 알려준다.
    EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
    p_result := -1; -- 실패가 아니면 다시 참으로 결과 값을 변경해준다.

    -- 반복횟수 만큼 돌린다?
    select max(detail_no)+1 into detailNo from resvdetail;

    INSERT INTO resvdetail (
      detail_no, resv_no, detail_start, detail_end
    ) VALUES (
      detailNo,p_resv_no, p_resv_date||'T'||V_START_TIME||':'||'30', p_resv_date||'T'||LPAD(TO_NUMBER(V_START_TIME)+1,2,'0')||':'||'00'
    );

    V_START_TIME := LPAD(TO_NUMBER(V_START_TIME)+1,2,'0'); -- 시작 시각을 1 증가시킨다.
    V_CHANGE := 0;

  ELSE
    SELECT NVL(MAX(A.RESV_CNT),0) AS RESV_CNT
    into val
    FROM( SELECT
      B.CMNT_NO
      , C.CMNT_NAME
      , DETAIL_START AS RESV_START
      , DETAIL_END AS RESV_END
      , SUM(RESV_CNT) RESV_CNT
    FROM
      RESVDETAIL A LEFT OUTER JOIN RESERVATION B ON (A.RESV_NO = B.RESV_NO)
      LEFT OUTER JOIN COMMUNITY C ON (B.CMNT_NO = C.CMNT_NO)
    WHERE C.APT_CODE = P_APT_CODE
      and c.cmnt_no = P_cmnt_no
    GROUP BY B.CMNT_NO, C.CMNT_NAME, DETAIL_START, DETAIL_END
    ) A LEFT OUTER JOIN COMMUNITY B ON (A.CMNT_NO = B.CMNT_NO)
    WHERE RESV_CNT = B.CMNT_CAPA
      and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'00'
      and RESV_END = p_resv_date||'T'||V_START_TIME||':'||'30';
    -- 30분 간격으로 인원을 조회한다.

    p_result := val; -- 실패라고 알려준다.
    EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
    p_result := -1; -- 실패가 아니면 다시 참으로 결과 값을 변경해준다.

    -- 반복횟수 만큼 돌린다?
    select max(detail_no)+1 into detailNo from resvdetail;

    INSERT INTO resvdetail (
      detail_no, resv_no, detail_start, detail_end
    ) VALUES (
      detailNo, p_resv_no, p_resv_date||'T'||V_START_TIME||':'||'00', p_resv_date||'T'||V_START_TIME||':'||'30'
    );

    V_CHANGE := 1;

  END IF;

  V_CNT := V_CNT+1;
end loop;

end;
```

```
WHERE A.RESV_CNT = B.CMNT_CAPA
      and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'30'
      and RESV_END = p_resv_date||'T'||LPAD(TO_NUMBER(V_START_TIME)+1,2,'0')||':'||'00';
-- 30분 간격으로 인원을 조회한다.

p_result := val; -- 실패라고 알려준다.
EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
p_result := -1; -- 실패가 아니면 다시 참으로 결과 값을 변경해준다.

-- 반복횟수 만큼 돌린다?
select max(detail_no)+1 into detailNo from resvdetail;

INSERT INTO resvdetail (
  detail_no, resv_no, detail_start, detail_end
) VALUES (
  detailNo,p_resv_no, p_resv_date||'T'||V_START_TIME||':'||'30', p_resv_date||'T'||LPAD(TO_NUMBER(V_START_TIME)+1,2,'0')||':'||'00'
);

V_START_TIME := LPAD(TO_NUMBER(V_START_TIME)+1,2,'0'); -- 시작 시각을 1 증가시킨다.
V_CHANGE := 0;

ELSE
  SELECT NVL(MAX(A.RESV_CNT),0) AS RESV_CNT
  into val
  FROM( SELECT
    B.CMNT_NO
    , C.CMNT_NAME
    , DETAIL_START AS RESV_START
    , DETAIL_END AS RESV_END
    , SUM(RESV_CNT) RESV_CNT
  FROM
    RESVDETAIL A LEFT OUTER JOIN RESERVATION B ON (A.RESV_NO = B.RESV_NO)
    LEFT OUTER JOIN COMMUNITY C ON (B.CMNT_NO = C.CMNT_NO)
  WHERE C.APT_CODE = P_APT_CODE
    and c.cmnt_no = P_cmnt_no
  GROUP BY B.CMNT_NO, C.CMNT_NAME, DETAIL_START, DETAIL_END
  ) A LEFT OUTER JOIN COMMUNITY B ON (A.CMNT_NO = B.CMNT_NO)
  WHERE RESV_CNT = B.CMNT_CAPA
    and RESV_START = p_resv_date||'T'||V_START_TIME||':'||'00'
    and RESV_END = p_resv_date||'T'||V_START_TIME||':'||'30';
  -- 30분 간격으로 인원을 조회한다.

  p_result := val; -- 실패라고 알려준다.
  EXIT WHEN val + p_resv_cnt > capa ; --인원이 다 차있으면 예약 등록 취소
  p_result := -1; -- 실패가 아니면 다시 참으로 결과 값을 변경해준다.

  -- 반복횟수 만큼 돌린다?
  select max(detail_no)+1 into detailNo from resvdetail;

  INSERT INTO resvdetail (
    detail_no, resv_no, detail_start, detail_end
  ) VALUES (
    detailNo, p_resv_no, p_resv_date||'T'||V_START_TIME||':'||'00', p_resv_date||'T'||V_START_TIME||':'||'30'
  );

  V_CHANGE := 1;

END IF;

V_CNT := V_CNT+1;
end loop;

end;
```

# 시설 - 나의 예약 조회하기

## 입주민사이트 나의 예약

## 예약 상세 조회

시설명

시설 조회

시설별 예약 조회하기

10

개씩 보기

이전

1

다음

시설 분류	시설명	예약날짜	예약시간	예약취소
사우나	힘찬 사우나	2021-03-13	09:00 ~ 11:00	
기타	푸향향 줌바댄스	2021-04-04	12:00 ~ 14:00	취소하기
기타	테니스장	2021-03-31	09:00 ~ 11:00	취소하기
기타	갯김치 필라테스	2021-03-11	09:00 ~ 11:00	
기타	푸향향 줌바댄스	2021-03-11	09:00 ~ 11:00	

예약정보

\* 예약 취소는 하루전까지 가능합니다.

시설명	푸향향 줌바댄스
예약자명	박정민
예약일	2021-04-04
핸드폰 번호	010-2451-6235
예약시간	12:00 ~ 14:00
예약 인원	5

닫기

예약 정보 클릭 시 모달 창으로 예약 정보 조회 결과를 보여줍니다.

# 아파트 관리 - 목록, 등록

## 아파트 목록

아파트 등록

아파트코드	아파트명	시작일	만료일	활성화여부
A0026	오투그란데3차유성숲	2021-03-12	2026-03-11	활성
A0025	가람아파트	2021-03-10	2022-03-10	활성
A0024	대덕아트힐	2019-01-01	2025-12-31	활성
A0020	한마루아파트	2021-03-08	2022-03-31	활성
A0016	대청아파트	2021-03-05	2022-03-05	활성

전체

검색

이전

1

2

3

다음

## 우편번호 찾기



## 아파트 등록

아파트 명	오투그란데3차유성숲	
우편번호	34159	우편번호 찾기
기본주소	대전 유성구 학하중앙로 60	
상세주소	상세주소	
아파트이미지	파일 선택    선택된 파일 없음	
난방정책	지역난방	
관리사무소 전화번호	010-5572-7833	
관리소장명	이미정	
계약시작일	2021-03-12	
계약만료일	2026-03-11	

아파트 등록

시연용 입력

등록된 아파트 단지 리스트를 활성화 여부로 정렬하여 보여줍니다.

Daum 우편번호 API를 이용했습니다.



# 아파트 관리 – 상세보기, 수정

## 벤더 사이트 상세보기

아파트 코드	A0026	아파트 명	오투그란데3차유성숲
우편번호	34159	소장ID	A0026E001
기본주소	대전 유성구 학하중앙로 60	상세주소	
세대수	15	총 주거면적	1470㎡
활성화여부	활성	난방정책	지역난방
관리사무소 전화번호	010-5572-7833	관리소장명	이미정
계약시작일	2021-03-12	계약만료일	2026-03-11
아파트이미지	이미지 없음		

단지설정

수정

목록

## 관리사무소 사이트 상세보기

아파트정보			
아파트코드	A0024	아파트명	대덕아트힐
난방정책	지역난방	우편번호	34940
기본주소	대전 중구 중앙로 76	상세주소	4층 관리사무소
세대수	180	총 주거면적	15900㎡
관리사무소 전화번호	042-222-8202	관리소장명	이경륜
계약시작일	2019-01-01	계약만료일	2025-12-31
아파트이미지			

홈으로 가기

아파트 단지 정보 수정하기

## 벤더 사이트 정보수정

아파트이미지	<div>파일 선택</div> 선택된 파일 없음
난방정책	<div>난방 선택</div>
관리사무소 전화번호	<div>01055727833</div>
관리소장명	<div>이미정</div>
계약시작일	<div>2021-03-12</div> <div></div>
계약만료일	<div>2026-03-11</div> <div></div>

아파트 정보 변경

# 아파트 관리 – 아파트 단지 생성

## 벤더 사이트 아파트 단지 생성

아파트

호

층

면적

단지생성

돌아가기

아파트	호	면적	삭제
			삭제
1201	0301	98㎡	삭제
1201	0401	98㎡	삭제
1201	0501	98㎡	삭제
1201	0601	98㎡	삭제
1201	0701	98㎡	삭제

## 단지 생성 프로시저

```
create or replace PROCEDURE p_house(
  p_aprcode in apart.aprt_code%type
  , p_dong in house.dong%type
  , p_floor in house.ho%type
  , p_ho in house.ho%type
  , p_area in HOUSE.HOUSE_AREA%type
  , p_result out number
)
is
  val number;
  v_cnt number := 1;
  out_cnt number :=0;
begin
  loop
    select count(*)
      into val
    from house
    where instr(house_code ,p_aprcode||'D'||LPAD(p_dong,4,'0'))||'H'||to_char(v_cnt,'FM00')||'0'||p_ho) > 0;
    if val = 0 then
      insert into house(
        house_code,
        dong,
        ho,
        move_yn,
        apt_code,
        house_area
      )
      values (
        p_aprcode||'D'||LPAD(p_dong,4,'0')||'H'||to_char(v_cnt,'FM00')||'0'||p_ho
        , LPAD(p_dong,4,'0')
        , LPAD(to_char(v_cnt)||'0'||p_ho,4,'0')
        , 'N'
        , P_APTCODE
        , p_area
      );
      out_cnt := out_cnt+1;
    end if;
    exit when v_cnt = TO_NUMBER(p_floor);
    v_cnt := v_cnt+1;
  end loop;
  p_result := out_cnt;
end;
```