클래스 설계 - DAO 설계

\*플래닛

플래닛 생성 -> 플래닛 멤버도 생성

INSERT INTO planet VALUES( #{플래닛ID}, #{플래닛이름}, ‘Y’ )

플래닛 찾기(id별, 플래닛 이름 별)

1) id별

SELECT planet\_name

FROM p\_mem pm

JOIN planet p ON pm.planet\_id=p.planet\_id

where member\_id=#{id}

2)이름 별

SELECT planet\_name FROM planet WHERE planet\_name=#{name}

플래닛 삭제(상태 status 수정)

UPDATE planet SET use\_status=’N’ WHERE planet\_id=#{id}

플래닛 수정(이름)

UPDATE planet SET planet\_name=#{name} WHERE planet\_id=#{id}

\*멤버

회원가입

INSERT INTO member VALUES( #{id}, #{pw}, #{name}, #{birth}, #{dept}, #{mail}, ‘Y’ )

비밀번호 변경

1SELECT member\_pw FROM member WHERE member\_id=#{id}

UPDATE member SET member\_pw=#{pw} WHERE member\_id=#{id}

계정 삭제(삭제상태status 수정)

1SELECT member\_pw FROM member WHERE member\_id=#{id}

UPDATE member SET use\_status=’N’ WHERE member\_id=#{id}

회원정보 수정

1SELECT member\_pw FROM member WHERE member\_id=#{id}

SELECT \* FROM member WHERE member\_id=#{id}

UPDATE member SET member\_name=#{name}, birthdate=#{birthdate}, dept=#{dept},

email=#{email} WHERE member\_id=#{id}

멤버 찾기(중복 체크)

SELECT \* FROM member WHERE member\_id=#{id}

\*플래닛\_멤버

회원이 플래닛에 가입했을 때 -> 멤버상태N

INSERT INTO p\_mem VALUES( #{member\_id}, #{planet\_id}, #{nickname}, #{reg\_date}, ‘N’, ‘Y’ )

회원이 플래닛을 만들었을 때 -> 멤버상태Y

INSERT INTO p\_mem VALUES( #{member\_id}, #{planet\_id}, #{nickname}, #{reg\_date}, ‘Y’, ‘Y’ )

플래닛 회원정보 수정(닉네임)

UPDATE p\_mem SET nickname=#{nickname} WHERE member\_id=#{id}

플래닛 탈퇴(삭제status 수정)

UPDATE p\_mem SET use\_status=’N’ WHERE member\_id=#{member\_id}

마스터 위임

UPDATE p\_mem SET member\_status=’Y’ WHERE member\_id=#{member\_id1}

UPDATE p\_mem SET member\_status=’N’ WHERE member\_id=#{member\_id2}

멤버 초대

SELECT member\_id FROM member WHERE member\_id=#{id}

INSERT INTO p\_mem VALUES( #{member\_id}, #{planet\_id}, #{nickname}, #{reg\_date}, ‘N’, ‘N’ )

~~멤버 찾기~~

~~SELECT \* FROM p\_mem WHERE member\_id=#{id}~~

멤버 정보 조회

SELECT \*

FROM p\_mem pm

JOIN member m ON pm.member\_id=m.member\_id

WHERE pm.member\_id=#{id}

\*팀

팀을 생성한다 -> 팀멤버(마스터, status=Y)가 생성된다

INSERT INTO team VALUES(#{team\_id},#{team\_name},#{member\_status})

팀 삭제(팀상태 stats 수정)

UPDATE team SET use\_status=#{use\_status} WHERE team\_id=#{id}

팀 이름 수정

UPDATE team SET team\_name=#{team\_name} WHERE team\_id=#{id}

\*팀\_멤버

팀멤버 추가(플래닛 멤버 찾기)

SELECT \* FROM p\_mem WHERE member\_id=#{member\_id} ( 멤버찾기)

INSERT INTO t\_mem VALUES ( #{member\_id}, #{team\_id}, #{planet\_id}, ‘Y’, ‘Y’ ) (마스터일 때)

INSERT INTO t\_mem VALUES ( #{member\_id}, #{team\_id}, #{planet\_id}, ‘N’, ‘Y’ ) (일반 팀원일 때)

\*\*팀멤버 추방(사용자상태 status 수정)

UPDATE t\_mem SET use\_status=’N’ WHERE member\_id=#{id}

팀장 위임(사용자상태 status 수정)

UPDATE t\_mem SET member\_status=’Y’ WHERE member\_id=#{id}

UPDATE t\_mem SET member\_status=’N’ WHERE member\_id=#{id}

팀멤버 목록 보기

SELECT \* FROM t\_mem WHERE team\_id=#{team\_id}

팀 탈퇴(사용자상태 status 수정)

UPDATE t\_mem SET use\_status=’n’ WHERE member\_id=#{id}

\*그룹

그룹 추가(이름, 공개/비공개 설정)

INSERT INTO nation VALUES(#{nation\_id},#{nation\_name}, ’y’)(공개)

INSERT INTO nation VALUES(#{nation\_id},#{nation\_name}, ’n’)(비공개)

그룹 수정(이름, 공개/비공개 설정)

UPDATE nation SET nation\_name=#{nation\_name}, member\_status=#{member\_status}

WHERE nation\_id=#{id}

그룹 찾기(내가 가입하지 않은, 공개 상태의 그룹 select. 아래 그룹 목록 보기의 반대)

SELECT \*

FROM nation

WHERE not nation\_id IN ( SELECT nation\_id FROM n\_mem WHERE member\_id='id1')

AND nation\_status='Y'

그룹 목록 보기(내가 가입 한 그룹. 그룹멤버에서 멤버ID가 나인 그룹멤버를 찾아서 그룹ID를 찾은 후 그룹ID를 갖는 그룹의 이름을 보여줌)

SELECT \*

FROM nation

WHERE nation\_id IN (SELECT nation\_id FROM n\_mem WHERE member\_id='id1')

그룹 삭제(그룹상태 status 수정)

UPDATE nation SET use\_status=’n’ WHERE nation\_id=#{id}

\*그룹\_멤버

그룹멤버 찾기

select \* from nation where nation\_id=#{nation\_id}

그룹멤버 추가 - 초대(플래닛 멤버 찾기)

INSERT INTO n\_mem VALUES(#{member\_id},#{planet\_id},#{nation\_id},’y’,’y’)

그룹멤버 추가 – 그룹 찾기에서 following

INSERT INTO n\_mem VALUES(#{member\_id},#{planet\_id},#{nation\_id},’y’,’y’)

그룹멤버 목록 보기

SELECT \* FROM n\_mem WHERE nation\_id=#{nation\_id}

그룹장 위임(사용자상태 status 수정)

UPDATE n\_mem SET member\_status=’y’ WHERE member\_id=#{id}

UPDATE n\_mem SET member\_status=’n’ WHERE member\_id=#{id}

그룹 탈퇴(사용자상태 status 수정)

UPDATE n\_mem SET use\_status=’n’ WHERE member\_id=#{id}

\*게시글

게시글 등록

1]일반게시글 - 댓글 컬럼 null

중요도의 default 값은 일반으로 놓기

# 글 작성 시 중요도 설정할 경우 해당 중요도 숫자로 설정하기

insert into ARTICLE VALUES(#{id}, ?, ?, ?, sysdate, null, ?, N, 중요도, N,#{id}, 1, 0)

2]댓글 - 중요도, 공지 상태 null

insert into ARTICLE VALUES(?, ?, ?, ?, sysdate, null, ?, N, null, null, 원글id, 순서, 1);

댓글 그룹은 원본글의 게시글 id

게시글 삭제(삭제상태status 수정, 첨부파일, 좋아요, 요청, 일정 status 수정, 댓글게시글도 삭제status 수정 - 댓글의 댓글그룹=삭제한 게시글ID 인 게시글(댓글)의 status 수정.)

update ARTICLE SET use\_status = ‘N’ where article\_id=삭제하려는 게시글 id

—>게시글에 연관된 파일, 일정, 요청, 좋아요 가 있는 경우

update PLAN\_ART SET use\_status = ‘N’ where article\_id = 삭제하려는 게시글 id

update REQ\_ART SET use\_status = ‘N’ where article\_id = 삭제하려는 게시글 id

update LIKE\_ART SET use\_status = ‘N’ where article\_id = 삭제하려는 게시글 id update FILE\_ART SET use \_status = ‘N’ where article\_id = 삭제하려는 게시글 id

— >게시글에 연관된 댓글 삭제 status 수정

삭제하려는 게시글의 id를 그룹으로 갖는 댓글 찾기

select article\_id from ARTICLE where grp = 삭제하려는 게시글 id;

if문으로 그러한 댓글이 존재할 경우 그러한 article\_id가 존재하는 경우

update ARTICLE SET use \_status = N where article\_id= 삭제하려는 댓글id;|

( 위에서 찾은 article\_id )

게시글 수정

1] 중요도 수정

update ARTICLE SET imp\_status = 변경하려는 중요도 where article\_id = 수정할 게시글id;

2] 공지사항 수정

update ARTICLE SET notice\_status = Y where article\_id = 공지할 게시글id;

\*\*\*중요도와 공지인 경우 수정날짜 변경안해줌

3] 일반 게시글 내용 수정

—>수정 날짜도 변경해 주어야한다.

update ARTICLE SET art\_contents = 변경내용 where article\_id = 수정할 게시글id;

update ARTICLE SET art\_edit\_date=sysdate where article\_id = 수정할 글 id;

게시글 조회(전체, 그룹별, id별)

1]전체

SELECT ( SELECT nation\_name FROM nation WHERE nation.nation\_id = art.nation\_id ) nation,

( SELECT member\_name FROM member WHERE member.member\_id = art.member\_id ) membername,

art.member\_id, art\_reg\_date, art\_contents,

plan\_contents, plan\_startdate, plan\_enddate,

rart.resp\_user\_id, req\_contents,

(SELECT count(\*) FROM LIKE\_ART lart WHERE art.article\_id = lart.article\_id AND use\_status='Y'),

grp, seq, lvl

FROM ARTICLE art left

JOIN PLAN\_ART part ON art.article\_id = part.article\_id left

JOIN REQ\_ART rart ON art.article\_id = rart.article\_id left

JOIN FILE\_ART fart ON art.article\_id = fart.article\_id

ORDER BY grp desc, seq asc;

2]그룹별

SELECT ( SELECT nation\_name FROM nation WHERE nation.nation\_id = art.nation\_id ) nation,

( SELECT member\_name FROM member WHERE member.member\_id = art.member\_id ) membername,

art.member\_id, art\_reg\_date, art\_contents,

plan\_contents, plan\_startdate, plan\_enddate,

rart.resp\_user\_id, req\_contents,

(SELECT count(\*) FROM LIKE\_ART lart WHERE art.article\_id = lart.article\_id AND use\_status='Y'),

grp, seq, lvl

FROM ARTICLE art left

JOIN PLAN\_ART part ON art.article\_id = part.article\_id left

JOIN REQ\_ART rart ON art.article\_id = rart.article\_id left

JOIN FILE\_ART fart ON art.article\_id = fart.article\_id

WHERE art.nation\_id=현재 그룹id

ORDER BY grp desc, seq asc;

3]id 별

SELECT ( SELECT nation\_name FROM nation WHERE nation.nation\_id = art.nation\_id ) nation,

( SELECT member\_name FROM member WHERE member.member\_id = art.member\_id ) membername,

art.member\_id, art\_reg\_date, art\_contents,

plan\_contents, plan\_startdate, plan\_enddate,

rart.resp\_user\_id, req\_contents,

(SELECT count(\*) FROM LIKE\_ART lart WHERE art.article\_id = lart.article\_id AND use\_status='Y'),

grp, seq, lvl

FROM ARTICLE art left

JOIN PLAN\_ART part ON art.article\_id = part.article\_id left

JOIN REQ\_ART rart ON art.article\_id = rart.article\_id left

JOIN FILE\_ART fart ON art.article\_id = fart.article\_id

WHERE art.member\_id=현재 회원id

ORDER BY grp desc, seq asc;

\*첨부파일

1)첨부파일 추가(첨부파일 명으로 저장)

insert into FILE\_ART VALUES (?,?,?,?, 첨부파일id, 첨부파일명, ‘Y’);

2)첨부파일 수정(파일 추가, 삭제)

글을 수정하며 첨부파일을 추가 하고 삭제

-첨부파일 추가

insert into FILE\_ART VALUES (?,?,?,?, 첨부파일id, 첨부파일명, ‘Y’);

-첨부파일 삭제 (이미 추가한 첨부파일을 ‘X’ 클릭으로 삭제할때)

첨부파일을 수정한 경우에는 아예 삭제할 것인가 상태만 바꿀것인가

\*\*\*\*첨부파일의 수정은 두가지 경우가 있음

글쓰기 도중에 추가했다가 삭제하는 경우

글쓰기 완료한 후에 삭제하는 경우

1. 상태만 바꾸기

UPDATE FILE\_ART SET use\_status=’N’ WHERE article\_id=#{id} AND file\_id=#{id}

~~//ii) db에서 아예 삭제 delete FROM FILE\_ART WHERE file\_name=삭제할 첨부파일의 id;~~

3)첨부파일 삭제(상태status 수정)

update FILE\_ART SET use\_status=’N’ WHERE article\_id=#{id} AND file\_id=#{id};

4)첨부파일 다운로드- 다운로드는 db에서 수행되는 것이 아닌것같은데…..

해당 첨부파일명을 클릭하면 해당파일이 저장된다.

5)첨부파일 조회

🡪그룹에 해당하는 모든 첨부파일 조회됨

SELECT file\_name, member\_id, art\_reg\_date

FROM FILE\_ART

JOIN article ON article.article\_id=file\_art.article\_id

WHERE nation\_id=현재 선택한 그룹;

\*좋아요

클릭했을 때 좋아요 사용자가 존재하는지 확인하고 없을 때 추가 있으면 상태변경

SELECT like\_member\_id FROM like\_art WHERE article\_id=#{id} AND like\_member\_id=#{id}

1)추가

INSERT INTO like\_art VALUES( #{article\_id}, #{누른사람id}, ‘Y’)

2)상태변경

UPDATE like\_art SET use\_status=’N’/’Y’ WHERE article\_id=#{id} AND like\_member\_id=#{id}

좋아요 한 사람 조회하기

SELECT like\_member\_id

FROM like\_art

WHERE article\_id=#{id} AND use\_status=’Y’

\*요청

요청 추가.(담당자는 그룹멤버 찾기)

SELECT member\_id FROM n\_mem WHERE member\_id=#{검색한id} AND nation\_id=#{id}

INSERT INTO req\_art VALUES(#{article\_id}, #{resp\_user\_id}, 0, #{req\_contents}, ‘Y’ )

요청 수정(내용변경, 담당자 변경(그룹 멤버 찾기) )

SELECT member\_id FROM n\_mem WHERE member\_id=#{검색한id} AND nation\_id=#{id}

UPDATE req\_art SET resp\_user\_id= #{resp\_user\_id, req\_contents= #{req\_contents}

WHERE article\_id=#{id}

요청 수정(상태만 수정)

UPDATE req\_art SET req\_status=#{status} WHERE article\_id=#{id}

요청 조회(ALL, ~~요청 보낸 것, 받은 것 세 가지 분류~~)

SELECT resp\_user\_id, req\_status, req\_contents,

member\_id, art\_reg\_date, art\_edit\_date

FROM req\_art

JOIN article ON article.article\_id=req\_art.article\_id

WHERE nation\_id=#{id}

요청 삭제(상태status 수정)

UPDATE req\_art SET use\_status=’N’ WHERE article\_id=#{id}

\*일정

일정 추가

INSERT INTO plan\_art VALUES( #{article\_id}, #{plan\_contents}, #{plane\_startdate}, #{plan\_enddate}, ‘Y’ )

일정 삭제(상태status 수정)

UPDATE plan\_art SET use\_status=’N’

WHERE article\_id=#{id}

일정 수정

UPDATE plan\_art SET plan\_contents=#{plan\_contents},

plan\_startdate=#{plan\_startdate},

plan\_enddate =#{plan\_enddate}

WHERE article\_id=#{id}

일정 조회(선택한 월 별로)

SELECT member\_id, plan\_contents, plan\_startdate, plan\_enddate

FROM plan\_art

JOIN article ON article.article\_id=plan\_art.article\_id

WHERE planet\_id=#{id} AND nation\_id=#{id}