오라클 개인 프로젝트

서종태

목차

DB 설계 목적

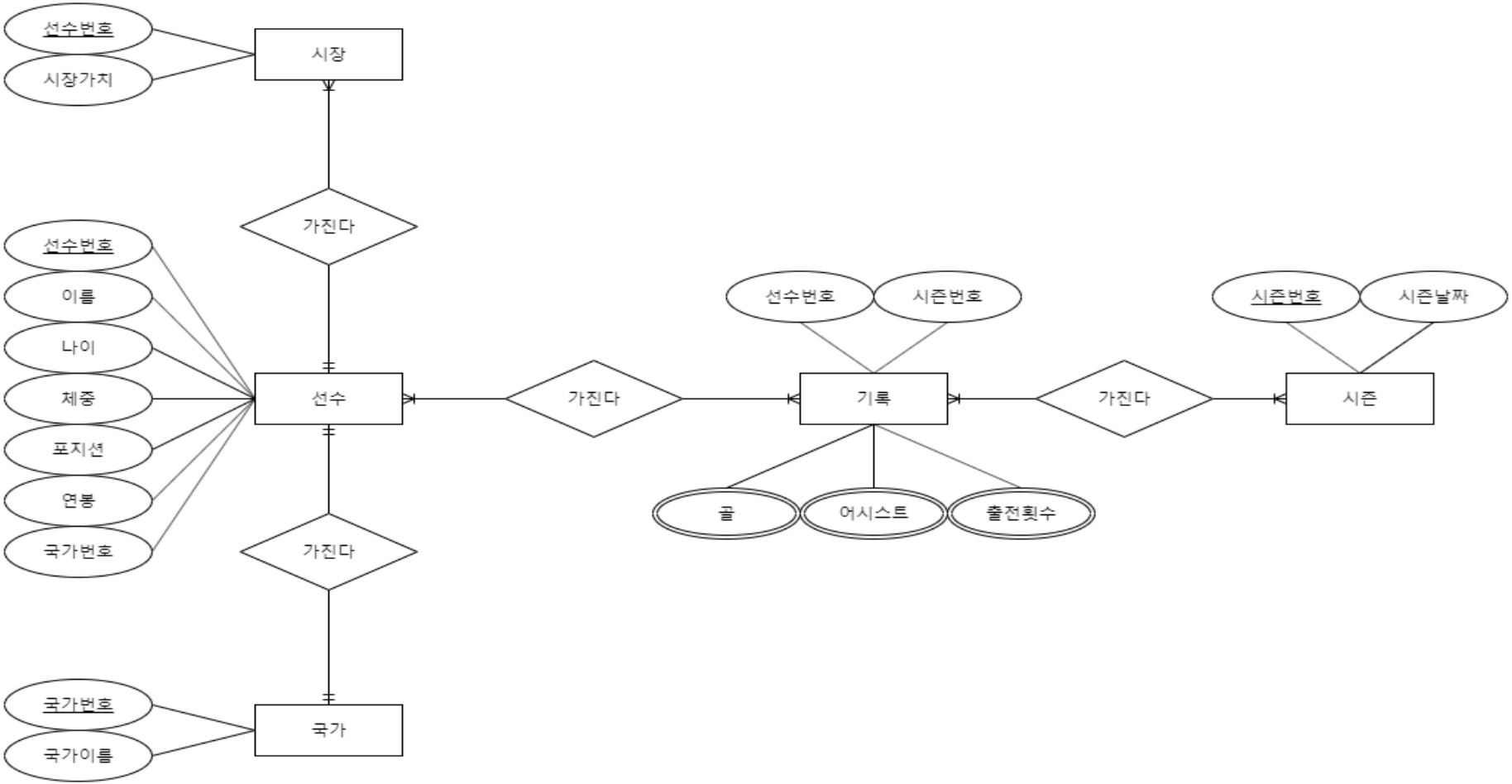
영국 프리미어리그 토트넘 축구팀에서 뛰는 선수들의 정보들을 쉽게 알수있게 하기 위해서 설계함

요구 사항 분석 및 엔티티 도출

1. 토트넘 팀 선수들의 정보를 쉽게 알수있음 (이름,나이 ,몸무게,포지션,연봉)
2. 토트넘 팀 선수들의 국가정보를 파악할수 있음
3. 토트넘 선수의 2019~2020 시즌의 정보를 파악할수 있음
4. 토트넘 선수의 2018~2019 시즌의 정보를 파악할수 있음
5. 토트넘 선수의 시장가치를 파악할수 있음
6. 토트넘 선수들의 기록을 파악할수 있음(골,어시스트,출전수)

빨간색 : 엔티티 도출

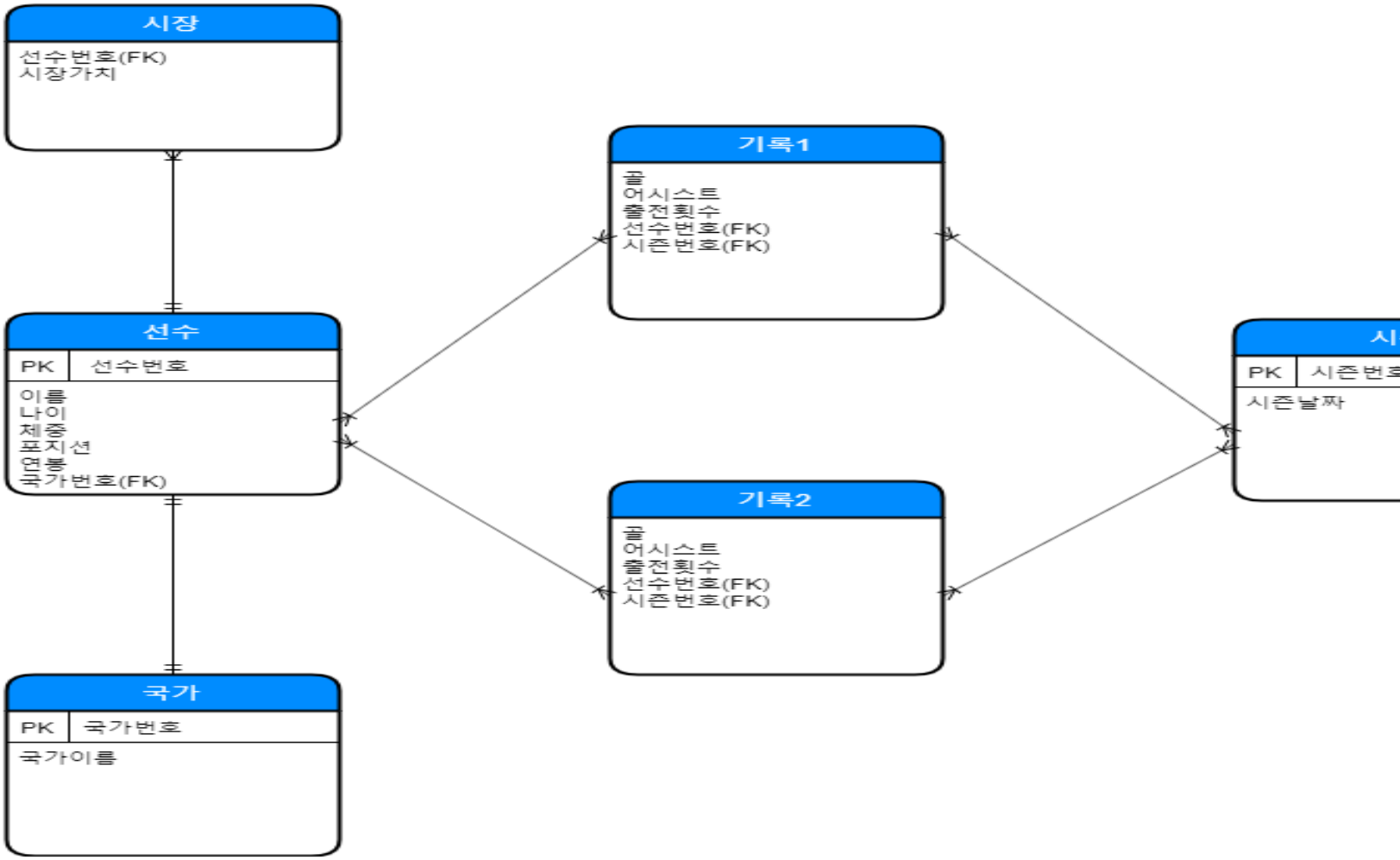
ERD 모델


선수번호
시장가치
시장
가진다
선수번호
이름
나이
체중
포지션
연봉
국가번호
선수
가진다
가진다
선수번호
시즌번호
기록
골
어시스트
출전횟수
가진다
시즌번호
시즌날짜
시즌
국가번호
국가이름
국가

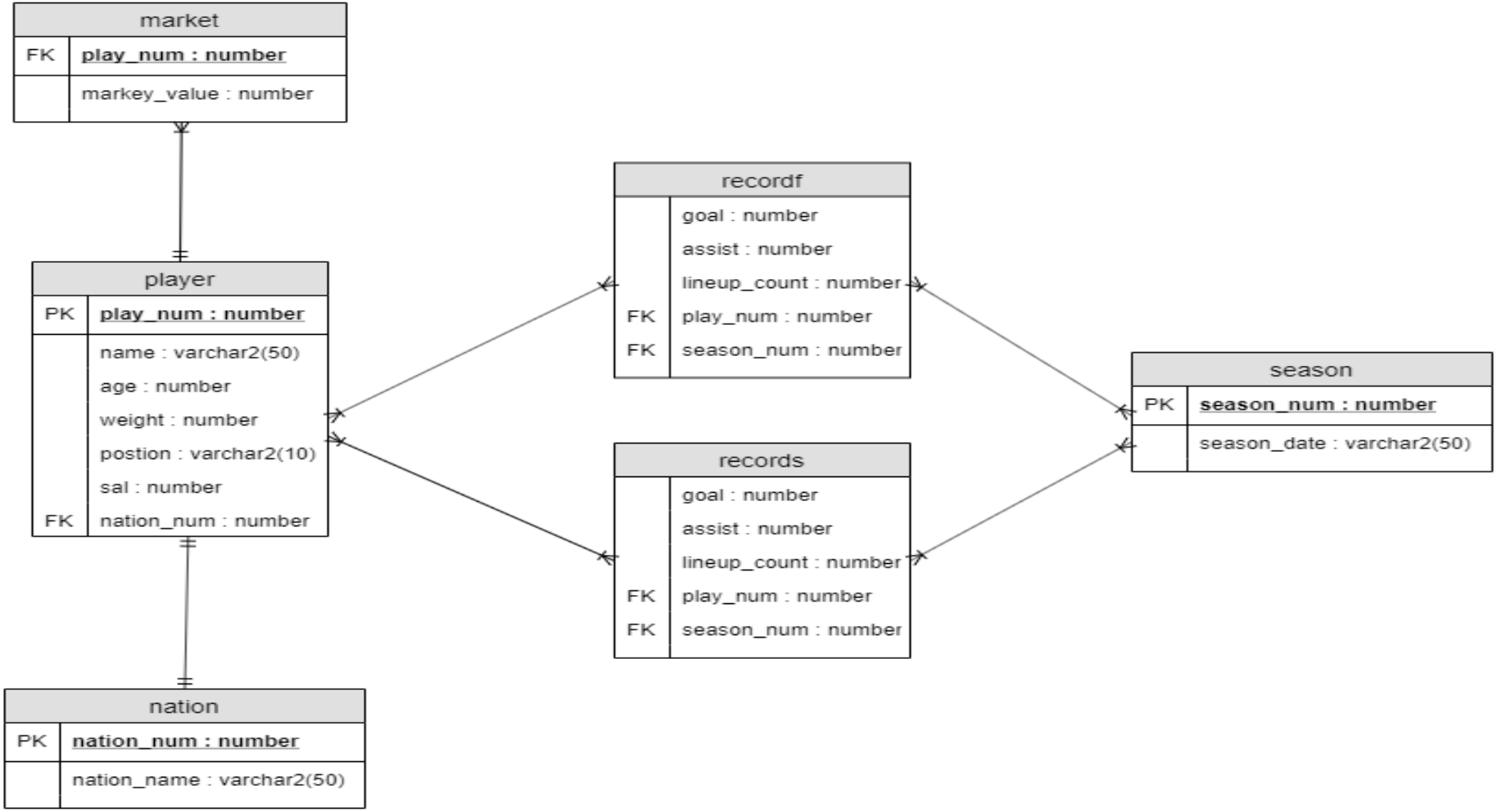
테이블 기술서

PDF 파일 별도 첨부

논리적 모델링


시장
선수번호(FK)
시장가치
기록1
골
어시스트
출전횟수
선수번호(FK)
시즌번호(FK)
선수
PK
선수번호
이름
나이
체중
포지션
연봉
국가번호(FK)
기록2
골
어시스트
출전횟수
선수번호(FK)
시즌번호(FK)
PK
시즌날짜
국가
PK
국가번호
국가이름

물리적 모델링


market
FK
play_num : number
markey_value : number
player
PK
play_num : number
name : varchar2(50)
age : number
weight : number
postion : varchar2(10)
sal : number
FK
nation_num : number
recordf
goal : number
assist : number
lineup_count : number
FK
play_num : number
FK
season_num : number
records
goal : number
assist : number
lineup_count : number
FK
play_num : number
FK
season_num : number
season
PK
season_num : number
season_date : varchar2(50)
nation
PK
nation_num : number
nation_name : varchar2(50)

SQL문 쿼리 작성

1. 테이블 생성

선수 테이블 생성

```
create table player(
play_num number primary key,
name varchar2(50) not null,
age number not null,
weight number not null,
position varchar2(10) not null,
sal number not null,
nation_num number,
constraint player_nation_num_fk
foreign key(nation_num) references nation(nation_num));
```

국가 테이블 생성

```
create table nation(
nation_num number primary key,
nation_name varchar2(50) not null);
```

시장 테이블 생성

```
create table market(
play_num number,
market_value number,
constraint market_play_num_fk
foreign key(play_num) references
player(play_num));
```

시즌 테이블 생성

```
create table season(
season_num number primary key,
season_date varchar2(50) not null);
```

기록(2019~2020 시즌 기록) 테이블 생성
```
create table recordf(
goal number,
assist number,
lineup_count number not null,
play_num number,
season_num number,
constraint recordf_play_num_fk foreign key(play_num)
references player(play_num),
constraint recordf_season_num_fk foreign key(season_num)
references season(season_num));
```

기록(2018~2019 시즌 기록) 테이블 생성
```
create table records(
goal number,
assist number,
lineup_count number not null,
play_num number,
season_num number,
constraint records_play_num_fk foreign key(play_num)
references player(play_num),
constraint records_season_num_fk foreign key(season_num)
references season(season_num));
```

2. 시퀀스 생성

선수번호(PK)에 시퀀스 주기
```
create sequence player_seq
start with 1
increment by 1;
```

국가번호(PK)에 시퀀스 주기
```
create sequence nation_num_seq
start with 10
increment by 10
maxvalue 120;
```

시즌번호(PK)에 시퀀스 주기
```
create sequence season_num_seq
start with 1
increment by 1
maxvalue 5;
```

3.트리거 사용

player 테이블의 pk인 play_num에 trigger를 생성함

```
create or replace trigger player_trigger
before insert
on player
for each row
begin
select player_seq.nextval into :new.play_num from dual;
end;
/
```

4번 insert 문
국가테이블에 insert

```
insert into nation
values(10,'한국');

insert into nation
values(20,'영국');

insert into nation
values(30,'프랑스');

insert into nation
values(40,'덴마크');

insert into nation
values(50,'아르헨티나');

insert into nation
values(60,'브라질');

insert into nation
values(70,'벨기에');

insert into nation
values(80,'콜롬비아');

insert into nation
values(90,'코트디부아르');
```

시즌테이블에 insert

```
insert into season
values(season_num_seq.nextval,'2019~2020');
```

```
insert into season
values(season_num_seq.nextval,'2018~2019');
```

선수 테이블에 INSERT

```
insert into player values(player_seq.nextval,'손흥민',27,77,'F',108,10);
insert into player values(player_seq.currval,'케인',26,86,'F',155,20);
insert into player values(player_seq.currval,'요리스',32,78,'G',77,30);
insert into player values(player_seq.currval,'에릭센',27,76,'M',58,40);
insert into player values(player_seq.currval,'알리',23,80,'M',116,20);
insert into player values(player_seq.currval,'로셀소',23,68,'M',60,50);
insert into player values(player_seq.currval,'모우라',27,72,'F',62,60);
insert into player values(player_seq.currval,'은돔벨레',22,76,'M',120,30);
insert into player values(player_seq.currval,'베르통언',32,88,'D',77,70);
insert into player values(player_seq.currval,'가자니가',27,90,'G',40,50);
insert into player values(player_seq.currval,'라멜라',27,79,'F',62,50);
insert into player values(player_seq.currval,'알더베이럴트',30,91,'D',72,70);
insert into player values(player_seq.currval,'오리에',26,76,'D',54,90);
insert into player values(player_seq.currval,'세세뇽',19,70,'D',55,20);
insert into player values(player_seq.currval,'산체스',23,81,'D',50,80);
insert into player values(player_seq.currval,'로즈',29,76,'D',46,20);
insert into player values(player_seq.currval,'시소코',30,90,'M',62,30);
insert into player values(player_seq.currval,'다이어',25,90,'M',46,20);
insert into player values(player_seq.currval,'윙크스',23,65,'M',67,20);
insert into player values(player_seq.currval,'피터스',22,62,'D',46,20);
insert into player values(player_seq.currval,'포이스',21,69,'D',42,50);
insert into player values(player_seq.currval,'데이비스',26,76,'D',47,20);
```

recordf 테이블에 insert

```
insert into recordf values(4,6,13,2,1);
insert into recordf values(7,1,14,3,1);
insert into recordf values(0,0,7,4,1);
insert into recordf values(1,1,12,5,1);
insert into recordf values(5,1,8,6,1);
insert into recordf values(0,0,8,7,1);
insert into recordf values(2,1,13,8,1);
insert into recordf values(2,1,12,9,1);
insert into recordf values(0,0,8,10,1);
insert into recordf values(0,0,3,11,1);
insert into recordf values(1,0,3,12,1);
insert into recordf values(0,2,14,13,1);
insert into recordf values(0,2,10,14,1);
insert into recordf values(0,0,1,15,1);
insert into recordf values(0,0,11,16,1);
insert into recordf values(0,0,10,17,1);
insert into recordf values(1,1,15,18,1);
insert into recordf values(0,0,5,19,1);
insert into recordf values(0,0,14,20,1);
insert into recordf values(0,0,3,21,1);
insert into recordf values(0,0,2,22,1);
insert into recordf values(0,0,6,23,1);
```

records 테이블에 insert

```
insert into records values(12,6,31,2,2);
insert into records values(17,4,28,3,2);
insert into records values(0,0,33,4,2);
insert into records values(8,12,35,5,2);
insert into records values(5,3,25,6,2);
insert into records values(0,0,0,7,2);
insert into records values(10,0,32,8,2);
insert into records values(0,0,0,9,2);
insert into records values(1,0,22,10,2);
insert into records values(0,0,9,11,2);
insert into records values(1,0,9,12,2);
insert into records values(0,0,34,13,2);
insert into records values(0,2,8,14,2);
insert into records values(0,0,0,15,2);
insert into records values(1,1,23,16,2);
insert into records values(0,3,26,17,2);
insert into records values(0,3,29,18,2);
insert into records values(3,0,20,19,2);
insert into records values(1,0,26,20,2);
insert into records values(0,3,6,21,2);
insert into records values(1,0,12,22,2);
insert into records values(0,0,27,23,2);
```

market 테이블에 insert

```
insert into market values(19,393);
insert into market values(15,458);
insert into market values(20,524);
insert into market values(8,524);
insert into market values(13,524);
insert into market values(7,655);
insert into market values(16,721);
insert into market values(9,850);
insert into market values(2,1051);
insert into market values(6,1181);
insert into market values(5,1312);
insert into market values(3,2000);
insert into market values(11,200);
```

5번 alter 문

player의 name 칼럼 데이터 타입 수정
alter table player modify name varchar2(20);

player테이블의 sal 칼럼명바꾸기

alter table player rename column sal to sal_million;

6번 update 문

알더베이럴트의 급여를 업데이트로 수정
update market
set market_value = 300
where play_num =13;

7번 delete 문

가자니가를 market 테이블에서 삭제
delete from market
where play_num = 11;

8번 조인

선수의 국가와 2018~2019시즌의 정보를 보여줌

```
select p.name,p.position,n.nation_name,rs.goal,rs.assist,
rs.lineup_count,s.season_date
from player p , nation n , records rs ,season s
where p.nation_num = n.nation_num
and p.play_num =rs.play_num
and rs.season_num = s.season_num;
```

선수의 시장 가치를 보여줌

```
select p.name,p.age,p.position,m.market_value
from player p inner join market m on p.play_num = m.play_num;
```

9번 뷰생성

선수정보와 시장가치를 합한 뷰
```
create view v_player_complex
as
select p.name,p.age,p.position,m.market_value
from player p inner join market m
on p.play_num = m.play_num;

select * from v_player_complex;
```

2018~19시즌의 한시즌의 선수의 정보를 보여주는 뷰
```
create view v_player_complex01
as
select p.name,p.position,n.nation_name,rs.goal,
rs.assist,rs.lineup_count,s.season_date
from player p , nation n ,records rs, season s
where p.nation_num = n.nation_num
and p.play_num = rs.play_num
and rs.season_num = s.season_num;
```

10번 커서

선수테이블의 정보를 모두 출력

```
set serveroutput on
declare
v_player player%rowtype;
cursor c1
is
select * from player;
begin
dbms_output.put_line('선수번호     이름     나이     체중     포지션     연봉     국가번호');
dbms_output.put_line('----------------------------------------------------------------------------------
----------------------');
for v_player in c1 loop
exit when c1%notfound;
dbms_output.put_line(v_player.play_num||'          '||v_player.name||'          '||v_player.age||'
'||v_player.weight||'          '||v_player.position||'          '||v_player.sal_million||'          '||v_player.nation_num);
end loop;
end;
/
```

11 번 프로시저

손흥민의 급여 구하기

```
create procedure son_sal
is
v_sal player.sal_million%type;
begin
select sal_million into v_sal
from player
where name='손흥민';
dbms_output.put_line
('손흥민의 급여는'||v_sal);
end;
/

execute son_sal;
```

토트넘 선수이름으로 특정선수의 급여조회

```
create procedure t_player_name
(v_name in player.name%type)
is
v_sal player.sal_million%type;
begin
select sal_million into v_sal
from player
where name = v_name;
dbms_output.put_line(v_name || '의 급여는' || v_sal);
end;
/

execute t_player_name('케인');
```

토트넘 선수의 이름을 조회해서 선수의 급여를 얻어옴

```
create procedure t_player_name2
(v_name in player.name%type,
v_sal out player.sal_million%type)
is
begin
select sal_million into v_sal
from player
where name = v_name;
end;
/

variable v_sal varchar2(14);
execute t_player_name2('에릭센',:v_sal);
print v_sal;
```

12번 사용자 정의함수

선수의 이름을 입력해서 포지션알아보기

```
create or replace function fn_name_position
(v_name in player.name%type)
return varchar2
is
v_position player.position%type;
begin
select position into v_position from player
where name = v_name;
return v_position;
end;
/

variable v_position varchar2;
execute :v_position := fn_name_position('손흥민');
print v_position;
```

감사합니다