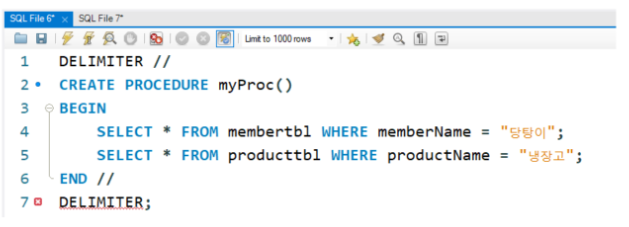
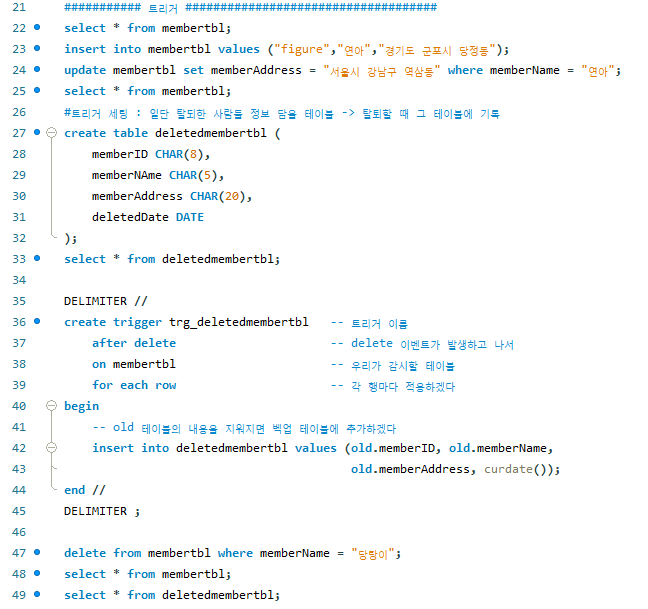
22-02-22

My SQL

* 파워셀 관리자 권한으로 실행 -> use employees; show tables; select \* from salaries;
* use 데이터베이스; select 내가 볼 컬럼 from 어느 테이블, where 조건; (전체 컬럼 보려면 : \*)
* 데이터 베이스에서 쿼리문을 직접 작성해서 테이블 추가 하기 : create table `이름` (id INT); 🡪 이름에 공란 있다면 백틱( ` ` ) 이용하기
* 테이블 제거하는 경우 : drop table `제거하려는 테이블 이름`;
* idxtbl을 만들어서 ­­­­­­­­­인덱스를 사용하면 속도가 더 빨라짐
* 뷰(가상의 테이블- 접근에 제한을 둠) : create view uv\_membertbl as select memberName,memberAddress from membertbl; select \* from un\_membertbl 🡪 uv\_membertbl은 Name과 Address만 보이게 됨
* 스토어드 프로시저(SQL문을 하나로 묶어 사용) :  🡪 gui로 만드는 방법도 있음
* 트리거(Trigger)- insert, delete, update발생하면 실행되는 코드 : 
* 백업(Data Export)과 복원(Data Import) 가능함
* 테이블의 디테일한 정보 보려면 : show table status;
* 테이블의 대략적인 정보 보려면 : describe table; ( describe 약어 desc로 사용가능) 🡪 pandas의 df.info( )랑 비슷 pandas의 describe 랑 다름 주의!
* DB 만들고 테이블 만들고 집어넣기 create datebases DB이름; create table테이블이름( 컬럼이름 char(8) not null primary key, 컬럼이름 int not null, 컬럼이름 date, … ); insert into 테이블 이름 values(‘~~’,’~~~’,’~~’,… );
  1. 🡪 타입이 안 맞거나 개수 안 맞으면 에러 남
  2. 컬럼 순서 변경해서 넣으려면 : insert into 테이블이름 (컬럼3, 컬럼1, 컬럼2) values (값3, 값1, 값2);
  3. 기존값과 중첩되서 에러나면 그 얘 빼고 남은애들 실행하려면 : ignore insert ignore membertbl values ("HYK", "현주협", "경기");
* 보고자 하는 컬럼을 합쳐서 그룹화 하기 : EX) select userID, name, concat(mobile1,"-",mobile2) as phonenumber from usertbl where addr = "서울" and mobile1 = 011;
* between A and B : A이상 B이하 🡪 부등호 대신 이용 가능
* 묶어 주기 위해 in ( , ) 이용 EX) select \* from from tbl where name in (“A”, “B”)
* 문자열 검색
  1. 첫 글자만 알 때 : “조%” 🡪 뒤에 글자수 제한 없음 EX) select name, addr from usertbl where name like "조%";
  2. 앞글자 모르고 뒤에만 알 때: “\_종신” EX) select name, addr from usertbl where name like "\_종신";
  3. 두번째 글자만 알 때 : "\_용%" EX) select name, addr from usertbl where name like "\_용%";
  4. 어디 있는지 모르는 글자 찾을 때: “%김%” EX) select name, addr from usertbl where name like "%김%";
* 서브 쿼리 : where 뒤에 조건에 쿼리 넣기 EX) select name, height from usertbl where height > (select height from usertbl where name = "이승기");
* 서브 쿼리에서 any (한 가지만 만족해고 가능) / all(모두 만족) 사용 EX) select name, height from usertbl where > any( select height from usertbl where addr = “서울”);

EX) select name, height from usertbl where > all( select height from usertbl where addr = “서울”);

* order by asc(오름차순 -디폴트) / desc(내림차순) 🡪 여러 개 쓰면 같을 때 다음걸로 순서 배열 EX) select name, height from usertbl order by height desc, name asc;
* 유니크한 값들 보여줌 : distinct 🡪가로줄 수 줄어듬 EX) select distinct addr from usertbl;
* n개만 보기 : limit n 🡪 order by와 같이 사용하면 top10 top5.. 알 수 있음 EX) select emp\_no, hire\_date from employees order by hire\_date desc limit 5;
* group by : 그룹별로 묶어서 보여줌 EX) select userID, sum(amount) as “총개수”from buytbl group by userID order by userID;
*  ‘
* 집계함수에 조건 걸 때 where 말고 having 써야함 : EX) select userID, sum(price\*amount) from buytbl group by userID having sum(price\*amount) > 1000 order by sum(price\*amount); 🡪group by 다음에 having 써야함 (순서 주의!)
* 총합 또는 중간 합계 낼 때 : rollup EX) select groupName, sum(price\*amount) as "비용" from buytbl group by groupName with rollup;
* update : 데이터 값을 변경하거나 지울 때
  1. 데이터 변경 :update testtbl4 set Lname = "a" where Fname = "100";
  2. 데이터 삭제
     1. delete는 데이터 기반으로 데이터만 사라짐 tbl은 유지 : delete from bigtbl1;
     2. drop은 DB, tbl 다 사라짐 : drop table bigtbl2;
     3. truncate은 데이터만 날리고 그대로 : truncate table bigtbl3;
* with : 쿼리문이 복잡할 때 단순화 시키기 위해 만드는 임시테이블 EX) with adhtbl (addr, maxHeight) as (select addr, max(height) from usertbl group by addr) select avg(maxHeight) as "지역최장신평균" from adhtbl;
* 형변환 : cast, convert
  1. cast - select cast(avg(amount) as signed integer) as "평균판매수" from buytbl;
  2. convert - select convert(avg(amount), signed integer) as "평균판매수" from buytbl;
* 조건문 if EX) select if (100>200, "true", "false");
* 아스키코드, 문자 변환
  1. 문자 -> 아스키코드 : select ascii("A");
  2. 아스키코드 -> 문자 : select char("65");
* concat\_ws : 구분자 🡪 날짜, 시간대… EX) select concat\_ws("/", "2022","02","25"); 🡪 concat으로만 쓰면 select concat("2022","/","02","/","25");
* pivot table create table pivotTest (uName char(3), season char(2), amount int); insert into pivotTest values("김범수", "겨울", 10), ("김범수","가을",25) … ; select \* from pivotTest;
  1. 가로 : 소비자 / 세로 : 계절별 합계 select uName, sum(if (season = "겨울", amount, 0)) as "겨울", … from pivotTest group by uName;
  2. 가로 : 계절별 / 세로 : 사람별로 select season, sum(if (uName = "김범수", amount, 0)) as "김범수", … from pivotTest group by season order by season;
* join
  1. corss join EX) select \* from members cross join movies; 🡪 간략적으로 쓸 때, select \* from members, movies;
  2. inner join EX) select \* from members ME inner join movies M on ME.movie\_id = M.id;
  3. outer join (left join 🡪 right join은 잘 쓰이지 않음) EX) select \* from movies M left join members ME on M.id = ME.movie\_id;
     1. 만약 테이블이 3개 라면 EX) select \* from 테이블1 left join 테이블2 on 테이블1.id = 테이블2.id left join 테이블3 on 테이블2.name = 테이블3.name;
  4. full join 🡪 union, union all EX) select \* from members ME left join movies M on ME.movie\_id = M.id union select \* from members ME right join movies M on ME.movie\_id = M.id;
* case when 조건1 then 결과1 when 조건2 then 결과2 else 결과3 end : 조건에 따른 다른 값을 출력하고 싶을 때
  1. Select 지역, case when 지역 in (“서울“, “경기“) then “수도권“ else “비수도권” end as 수도권구분 from 테이블
  2. 위의 식에다가 + group by case when addr in ("서울", "경기") then "수도권" else "비수도권" end; OR group by 1;
  3. sum(case when addr = "서울" then 1 else 0 end) 🡪 카운트로 사용 가능
* 서브 쿼리 🡪 select/where/from 뒤에 select 쿼리문 적기 EX) select \* from (select \* from usertbl) A; \*뒤에 A처럼 약어 써야함
* 순위 정렬
  1. Rank : 중복만큼 skip하고 건너 뜀 EX) select \*, rank( ) over (order by id desc) as Rnk from 테이블
  2. DENSE\_RANK : 중복순위하고 그 다음은 다음 수 skip없음 EX) select \*, dense\_rank( ) over (order by id desc) as Rnk from 테이블
  3. ROW\_NUMBER : 중복값들 에대해서도 순차적인 순위 EX) select \*, row\_rank( ) over (order by id desc) as Rnk from 테이블
  4. PARTITION BY : 특정 속성별로 구분하고자 할 때 EX) select \*, row\_rank( ) over (partition by id order by name desc) as Rnk from 테이블
* 순위 정렬에서 top 10만 보고 싶을 때 EX) select \* from (위의 보이는 테이블) where Rnk <= 10 order by Name;
* floor(숫자) : 값보다 작은 정수 중 가장 큰 정수 🡪 소수점 이하 버림