문제의 풀이는 정답이 아닐 수 있으니 유의할 것!

/\*다음의 설명을 신중하게 읽고 문제의 요청한 대로 답변하도록 하세요.

문제의 답변은 별도의 제시된 파일에 작성하되 파일 명은 자신 이름으로 저장합니다. 홍길동학생의 경우, 홍길동.sql로 작성합니다. 다른 이름으로 작성된 파일의 경우 평가하지 않습니다.

\*/

-- A. DB 정의

/\*1) classicmodels 데이터베이스로부터 일부 데이터를 추출하여 데이터베이스를 생성한다. 데이터베이스는 '학생이름DB'으로 정한다.

즉, 홍길동 학생의 경우, '홍길동DB'로 표현한다. 이를 지키지 않을 경우는 더이상 평가하지 않습니다.\*/

drop database if exists 이재원DB;

create database if not exists 이재원DB;

use 이재원DB ;

/\*2) 홍길동DB에는 다음의 E-R 다이어그램에서 제시한 것처럼 테이블과 필드를 정의한다. 테이블 명의 앞에는 자신의 이름을, 필드 명의 끝에는 학번의 마지막 번호를 포함한다.

예를 들어 학번이 123456인 홍길동의 경우, 홍길동\_customers, 홍길동\_orders, 홍길동\_orderDetails,

홍길동\_products의 테이블들을 생성하고 테이블의 모든 필드의 이름은 6으로 끝나도록 정의한다.

\*/

drop table if exists 이재원\_customers;

create table 이재원\_customers(

customerNo0 int primary key,

customerName0 varchar(50),

salesRepEmployeeLastName0 varchar(50)

);

drop table if exists 이재원\_orders;

create table 이재원\_orders(

orderNo0 int primary key,

orderDate0 date,

amount0 dec(10,2) default 0,

customerNo0 int,

foreign key(customerNo0) references 이재원\_customers(customerNo0)

);

drop table if exists 이재원\_orderDetails;

create table 이재원\_orderDetails(

orderNo0 int,

productCode0 varchar(15),

qtyOrdered0 int,

priceEach0 dec(10,2),

primary key(orderNo0,productCode0),

foreign key(orderNo0) references 이재원\_orders(orderNo0),

foreign key(productCode0) references 이재원\_products(productCode0)

);

drop table if exists 이재원\_products;

create table 이재원\_products(

productCode0 varchar(15) primary key,

productName0 varchar(70),

qtyInStock0 smallint,

MSRP0 decimal(10,2)

);

/\*B. 데이터 생성

1) 기본 데이터의 입력

Classicmodels의 테이블들을 조회 결과를 홍길동DB의 테이블들에 데이터를 입력한다. 이때 insert <테이블> select ... 형식의 명령문을 사용한다.\*/

select \* from classicmodels.orders;

select \* from classicmodels.orderdetails;

select \* from classicmodels.customers;

select \* from classicmodels.employees;

select \* from classicmodels.products;

insert into 이재원\_orders(orderNo0,orderDate0,customerNo0)

select orderNumber, orderDate,customerNumber from classicmodels.orders;

insert into 이재원\_orderDetails(orderNo0, productCode0,qtyOrdered0,priceEach0)

select orderNumber, productCode, quantityOrdered,priceEach from classicmodels.orderdetails;

-- 2) 홍길동\_customers 테이블에 학생 자신의 정보를 고객으로 입력한다(insert문).

insert into 이재원\_customers(customerNo0,customerName0,salesRepEmployeeLastName0)

select c.customerNumber, '이재원', e.lastName from classicmodels.customers c,classicmodels.employees e

where c.salesRepEmployeeNumber = e.employeeNumber ;

select \* from 이재원\_customers;

-- 3) 홍길동\_products 테이블에 좋아하는 차량을 등록한다(insert문).

insert into 이재원\_products(productCode0,productName0,qtyInStock0,MSRP0)

select productCode, '아우디',quantityInStock,MSRP from classicmodels.products;

select \* from 이재원\_customers;

select \* from 이재원\_products;

select \* from 이재원\_orders;

select \* from 이재원\_orderDetails;

-- C. 주요 개념 이해도 확인

-- 1) 홍길동\_orders의 amount6는 orderDetails에서 해당 주문번호의 주문 금액을 나타낸다. 모든 주문에 대한 주문금액을 계산하는 명령문(update)을 쓰시오.

update 이재원\_orders o

set amount0 = (select sum(qtyOrdered0 \*qtyOrdered0) from 이재원\_orderDetails od where o.orderNo0 = od.orderNo0);

select \* from 이재원\_orders;

-- 2) (뷰) 2004년도의 제품별 제품명, 총 주문수량, 평균 주문가격을 구하는 view를 정의하시오.

create view get\_products(name, qty, avg\_amount) as

select p.productName0, sum(qtyOrdered0), avg(o.amount0)

from 이재원\_orders o, 이재원\_orderDetails od, 이재원\_products p

where year(o.orderDate0) = 2004 and od.productCode0 = p.productCode0

and od.orderNo0 = o.orderNo0

group by od.productCode0;

select \* from get\_products;

-- 3) (뷰) view 갱신 여부를 보이는 테이블을 사용하여 위의 view의 갱신 여부를 조회하는 명령문을 쓰시오.

select table\_name, is\_updatable

from information\_schema.views

where table\_schema = '이재원DB';

-- 4) 홍길동\_orders 테이블에 B.2),3)에서 입력한 정보를 기반으로 새로운 주문 정보를 입력한다.

select \* from get\_products;

select \* from 이재원\_orders;

insert into values

-- 5) (트리거) 홍길동\_orderDetails 테이블에 트리거를 생성하여 한 행이 삽입될 때마다 홍길동\_products 테이블의 관련 제품의 재고 수량(qtyInStock)을 주문 수량만큼 감소시키려 한다.

-- 이 내용을 홍길동\_orderDetails\_Insert 트리거로 구현하시오.

delimiter //

create trigger 이재원\_orderDetails\_Insert

before insert on 이재원\_orderDetails for each row

begin

update 이재원\_products p

set qtyInStock0 = qtyInStock0 - 1

where p.productCode0 = new.productCode0;

end //

delimiter ;

-- 6) (저장프로시저) 홍길동\_orderDetails 테이블에 레코드를 삽입하는 저장 프로시저(홍길동\_addRecord\_to\_orderDetails)를 구현한다. \*/

delimiter //

create procedure 이재원\_addRecord\_to\_orderDetails(

in porderNo0 char(10), in productCode0 char(10), in pqtyOrdered0 int, in ppriceEach0 float)

begin

insert into 이재원\_orderDetails values

(porderNo0,pproductCode0,pqtyOrdered0 , ppriceEach0);

end //

delimiter ;