Database Systems Laboratory Homework 2

정보통신공학과

2017112200 신현호

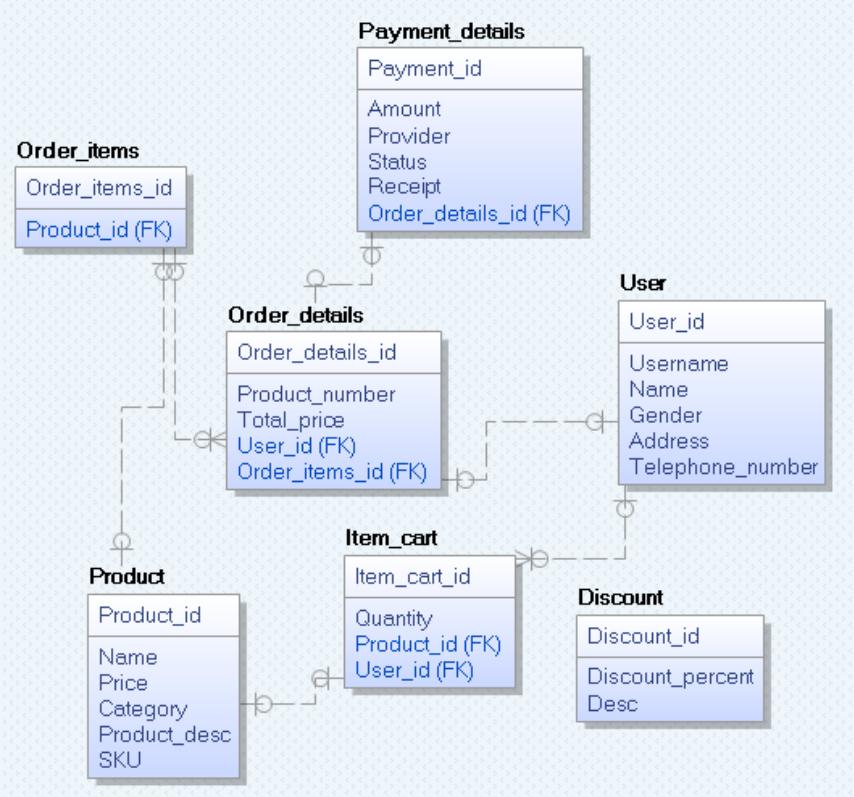
2017112207 이건우

1. Convert the E-R diagram of your HW 1 into database tables. Your database tables must be based exactly on your HW 1 (no exception). You can never change the subject, company, or the application of the HW1 (no exception). However, it is allowed to change/modify a little bit of the E-R such as the attributes, entities, relationships, etc. If you change the E-R (no matter how small the changes are), you should hand in the revised version of ER along with HW2.
    a. Using MySQL software, create your own database tables based on your E-R diagram(HW 1). Show the SQL commands you used for creating the tables.

```
create database hmdb;
use hmdb;

create table product(
   product_id int primary key,
   product_name varchar(20),
   price int,
   category varchar(20),
   product_desc varchar(20),
   sku varchar(20)
);

create table discount(
   discount_id int,
   discount_percent int,
   discount_desc varchar(20),
   primary key(discount_id)
);

create table order_items(
   order_items_id int primary key,
   product_id int,
   foreign key(product_id) references product(product_id)
);

create table user(
   user_id int primary key,
   username varchar(20),
   name varchar(20),
   gender varchar(5),
   address varchar(30),
   telephone_number varchar(13)
);


create table item_cart(
   item_cart_id int primary key,
```

```
    quantity int,
    product_id int,
    user_id int,
    foreign key(product_id) references product(product_id),
    foreign key(user_id) references user(user_id)
);

create table order_details(
    order_details_id int primary key,
    product_number int,
    total_price int,
    user_id int,
    order_items_id int,
    foreign key(user_id) references user(user_id),
    foreign key(order_items_id) references order_items(order_items_id)
);

create table payment_details(
    payment_id int primary key,
    amount int,
    provider varchar(20),
    status varchar(20),
    receipt varchar(20),
    order_details_id int,
    foreign key(order_details_id) references order_details(order_details_id)
);
```

b. Show the schema diagram of your database tables. The schema diagram ought to include table schema, attribute, primary key, foreign key, constraints etc.



2. Implement the following tasks using MySQL software.
    a. Select some tables (more than 3), and materialize them(insert some tuples into the tables). Try to keep the tuple size moderate (not too many tuples). Show the SQL commands and the contents of tables before and after the command.

```
-- discount table
insert into discount values (00001, 13, "shoes");
insert into discount values (00242, 22, "bags");
insert into discount values (28489, 04, "shorts");
insert into discount values (33902, 30, "wallets");
insert into discount values (45354, 63, "hoodies");

-- product table
insert into product values (45924, "red shoes", 30000, "shoes", "for women", "K898290");
insert into product values (34892, "students bag", 40000, "bags", "for students", "B939309");
insert into product values (88392, "khaki shorts", 20000, "shorts", "military style", "S839329");
insert into product values (00372, "black wallet", 100000, "wallets", "for business men", "W384248");
insert into product values (74022, "micky mouse hoodie", 60000, "hoodies", "disney merch", "H383024");

-- user table
insert into user values ('001','hhshin','신현호','M','경기도 수원시 영통구','01000000000');
insert into user values ('010','gunnoo','이건우','M','서울특별시 마포구', '01000000001');
insert into user values ('100','Kindess','박친절','F','대구광역시 달성구','01000000011');
insert into user values ('011','James98','호현호','M','서울특별시 중구','01000000111');
insert into user values ('111','hello02','우건우','M','서울특별시 성북구','01011000000');

-- item_cart table
insert into item_cart values ('200',1,45924,'001');
insert into item_cart values ('210',2,34892,'010');
insert into item_cart values ('220',3,88392,'001');
insert into item_cart values ('230',2,00372,'011');
insert into item_cart values ('240',7,74022,'111');
```

- discount table

discount_id	discount_percent	discount_desc
1	13	shoes
242	22	bags
28489	4	shorts
33902	30	wallets
45354	63	hoodies

- product table

product_id	product_name	price	category	product_desc	sku
372	black_wallet	100000	wallets	for business men	W384248
34892	students_bag	40000	bags	for students	B939309
45924	red_shoes	30000	shoes	for women	K898290
74022	micky_mouse_hoodie	60000	hoodies	disney merch	H383024
88392	khaki_shorts	20000	shorts	military style	S839329
NULL	NULL	NULL	NULL	NULL	NULL

- user table

	user_id	username	name	gender	address	telephone_number
▶	1	hhshin	신현호	M	경기도 수원시 영통구	01000000000
	10	gunnoo	이건우	M	서울특별시 마포구	01000000001
	11	James98	호현호	M	서울특별시 중구	01000000111
	100	Kindess	박친절	F	대구광역시 달성구	01000000011
	111	hello02	우건우	M	서울특별시 성북구	01011000000

- item_cart table

	item_cart_id	quantity	product_id	user_id
▶	200	1	45924	1
	210	2	34892	10
	220	3	88392	1
	230	2	372	11
	240	7	74022	111
*	NULL	NULL	NULL	NULL

b. Execute at least two SQL commands for each of the following clause (12 SQLs in total). For the single-person project, you can execute one SQL command per each clause. Show a) the query in plain English, (explain what this query is about), b) the corresponding SQL command, and c) the results of the command, executed using your database.
I. use WHERE

```
-- show the discount id which is 30% off coupon
select D.discount_id
from discount as D
where D.discount_percent = 30;
```

	discount_id
▶	33902
*	NULL

```
-- show the quantity of certain product id in the cart
select I.quantity
from item_cart as I
where I.product_id = '88392';
```

	quantity
▶	3

II. more than one tables in FROM

```
-- show products which are on sale
select product_id, product_name, product_desc
from product, discount
where product.category = discount.discount_desc;
```

	product_id	product_name	product_desc
▶	45924	red shoes	for women
	34892	students bag	for students
	88392	khaki shorts	military style
	372	black wallet	for business men
	74022	micky mouse hoodie	disney merch

```
-- estimate earnings if all the items in the carts are sold
select sum(price)
from item_cart, product
where item_cart.product_id = product.product_id
```

	sum(price)
▶	275000

III. use SET operation

```
-- showing product_id and product_name with item_cart_id in item_cart by using
-- left join function
select P.product_id, P.product_name, I.item_cart_id
from product as P
left join item_cart as I on P.product_id = I.product_id;
```

	product_id	product_name	item_cart_id
▶	372	black_wallet	230
	34892	students_bag	210
	45924	red_shoes	200
	74022	micky_mouse_hoodie	240
	88392	khaki_shorts	220

```
-- showing all the data in user table and product_id and user_id in item_cart table
-- by using join function
select *
from user as U
join item_cart as I on U.user_id = I.user_id;
```

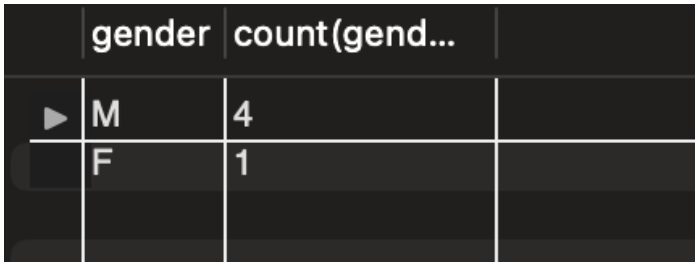
	user_id	username	name	gender	address	telephone_number	item_cart_id	quantity	product_id	user_id
▶	1	hhshin	신현호	M	경기도 수원시 영통구	01000000000	200	1	45924	1
	1	hhshin	신현호	M	경기도 수원시 영통구	01000000000	220	3	88392	1
	10	gunnoo	이건우	M	서울특별시 마포구	01000000001	210	2	34892	10
	11	James98	호현호	M	서울특별시 중구	01000000111	230	2	372	11
	111	hello02	우건우	M	서울특별시 성북구	01011000000	240	7	74022	111

IV. use aggregate function and/or GROUP BY

```
-- calculate sum of the price of each category
select category, sum(price)
from product
group by category;
```

	category	sum(price)
▶	wallets	110000
	bags	44000
	shoes	33000
	hoodies	66000
	shorts	22000

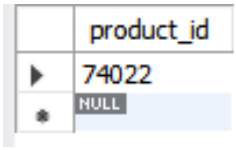
```
-- count the genders of the users
select gender, count(gender)
from user
group by gender;
```



	gender	count(gend...
▶	M	4
	F	1

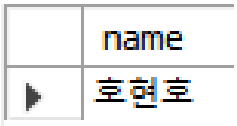
V. use SUBQUERY

```
-- showing product_id of which price is more than 35000 won and quantity is over two
select P.product_id
from product as P
where P.price > 35000 and product_id in (select I.product_id
                                         from item_cart as I
                                         where I.quantity > 2);
```



	product_id
▶	74022
*	NULL

```
-- showing user's name whose gender is male and who has a product in his item cart
-- and product's id is '372'
select U.name
from user as U
where U.gender = 'M' and U.user_id = (select I.user_id
                                      from item_cart as I
                                      where I.product_id = '372');
```



	name
▶	호현호

VI. use EXISTS or UNIQUE

```
-- if there is no over-priced product, then show all the products
select *
from product
where not exists
  (select product_id
   from product
   where price > 1000000000);
```

	product_id	product_name	price	category	product_desc	sku
▶	372	black wallet	110000	wallets	for business men	W384248
	34892	students bag	44000	bags	for students	B939309
	45924	red shoes	33000	shoes	for women	K898290
	74022	micky mouse hoodie	66000	hoodies	disney merch	H383024
	88392	khaki shorts	22000	shorts	military style	S839329

```
-- if there's on sale more than 50%, then show all the products
select *
from product
where exists
  (select *
   from discount
   where discount_percent > 50);
```

	product_id	product_name	price	category	product_desc	sku
▶	372	black wallet	110000	wallets	for business men	W384248
	34892	students bag	44000	bags	for students	B939309
	45924	red shoes	33000	shoes	for women	K898290
	74022	micky mouse hoodie	66000	hoodies	disney merch	H383024
	88392	khaki shorts	22000	shorts	military style	S839329

c. Execute at least TWO SQL commands that change the value(s) of some attributes using some conditions. Show a) the query in plain English, b) the corresponding SQL command, and c) the results of the command, executed using your database.

   i. 가격 변경

```
set SQL_SAFE_UPDATES = 0;

-- 1. 가격 변경
update product set price = price * 1.1;
```

price
110000
44000
33000
66000
22000

ii. 수량 변경

```
-- 2. 수량 변경
update item_cart set quantity = quantity*2 where product_id = '88392';
```

item_cart_id	quantity	product_id	user_id
200	1	45924	1
210	2	34892	10
220	6	88392	1
230	2	372	11
240	7	74022	111

d. Execute at least 5 SQL commands (3 SQL commands for single-person project) which you think is important or necessary for your application. Show a) the query in plain English, b) the corresponding SQL command, and c) the results of the command, executed using your database.

```
-- user can serach products by categories
select category, product_name, product_desc, price
from product
where category = "wallets";
```

category	product_name	product_desc	price
wallets	black_wallet	for business men	110000

```
-- user can search products by price
select product_id, price
from product
where price < 70000;
```

product_id	price
34892	44000
45924	33000
74022	66000
88392	22000
NULL	NULL

```
-- changed discount percent
update discount set discount_percent = discount_percent * 1.5 where discount_desc = 'shorts';
```

	discount_id	discount_percent	discount_desc
▶	1	13	shoes
	242	22	bags
	28489	6	shorts
	33902	30	wallets
	45354	63	hoodies

```
-- changed address
update user set address = '서울특별시 광진구' where name = '우건우';
```

	user_id	username	name	gender	address	telephone_number
▶	111	hello02	우건우	M	서울특별시 광진구	01011000000
*	NULL	NULL	NULL	NULL	NULL	NULL

```
-- changed phone_number
update user set telephone_number = '01048807780' where name = '신현호';
```

	user_id	username	name	gender	address	telephone_number
▶	1	hhshin	신현호	M	경기도 수원시 영통구	01048807780
*	NULL	NULL	NULL	NULL	NULL	NULL