[로그인]

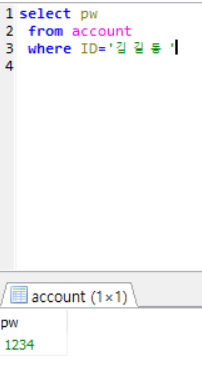
Login(input\_ID,input\_qw)

{ if(input\_pw=(select pw from account where ID=input\_ID))

Ex\_Login

};

로그인은 사용자가 입력한 pw가 DataBase에 저장되어있는 비밀번호와 일치하면 로그인이 되도록 하였다. select pw from account where ID=input\_ID 부분만 실행해보면 다음과 같다.



이처럼 ID가 ‘김길동’인 유저의 PW는 1234라고 DB에 저장

되어 있으며, 이와 유저가 입력한 PW가 일치할 때 로그인

을 허용해 주는 것이다.

[키워드 검색]

<SQL>

select res\_id, name, category, location

from Restaurant

where name like '%Keyword%';

<예시>에는 필요한 부분을 임의로 넣어서 진행하였다.

select res\_id, name, category, location

from Restaurant

where name like '%우리%';

<예시 결과>



[정렬(이름순)]

<SQL>

select res\_id, name, category, location

From Restaurant

order by name

<결과>



[정렬(평점순)]

<SQL>

select res\_id, name, category, location, avg(Review.rate)

From Restaurant natural join Review

group by res\_id

order by avg(Review.rate) desc

<결과>



[정렬(거리순)]

<SQL>

select res\_id, name, category, location

From Restaurant

group by res\_id

order by location

<결과>



[정렬(분류별)]

<SQL>

select res\_id, name, category, location

From Restaurant

where category='input\_category'

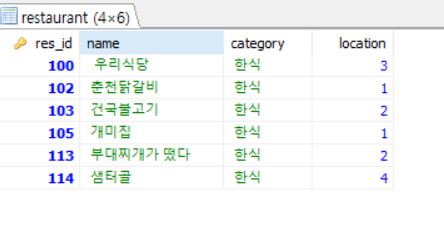
<예시>

select res\_id, name, category, location

From Restaurant

where category='한식'

<예시 결과>



[리뷰 작성]

<SQL>

INSERT INTO Review VALUES(rv\_id,'input\_rate','input\_reivew',res\_id,user\_id);

<예시>

INSERT INTO Review VALUES(100,1,'진짜 돈이 아까웠어요.',102,5);

<예시 결과>



-들어간 것은 [식당 정보]의 예시결과에서 확인

[식당 정보]

<SQL>

select account.ID, `review`, rate

from Review natural join Users

where res\_id='input\_res\_id'

<예시>

select account.ID ,`review` ,rate

from Review natural join account

where res\_id=102

<예시 결과>



[쿠폰 정보 탐색]

<SQL>

select cp\_id, Restaurant.name, discount, `point`

from Coupon natural join Restaurant

where res\_id='input\_res\_id'

<예시>

select cp\_id, Restaurant.name, discount, `point`

from Coupon natural join Restaurant

where res\_id=100

<예시 결과>



[쿠폰 구매]

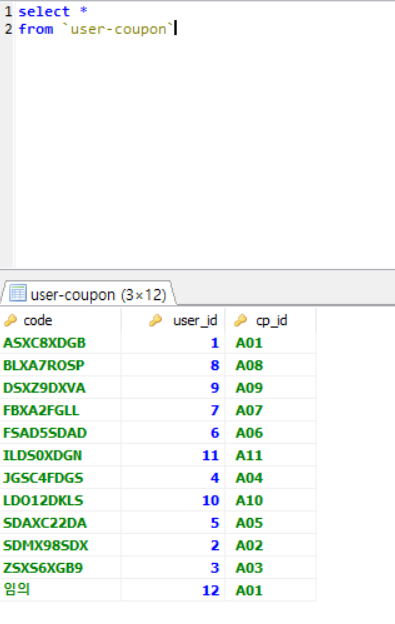
<SQL>

insert into `User-Coupon` values('해당user\_id','구매한 cp\_id','code');

<예시>

INSERT INTO `User-Coupon`(code,user\_id,cp\_id) VALUES('임의',12,'A01');

<예시 결과>



[쿠폰 사용]

<SQL>

delete from `User-Coupon`

where cp\_id='Used\_cp\_id';

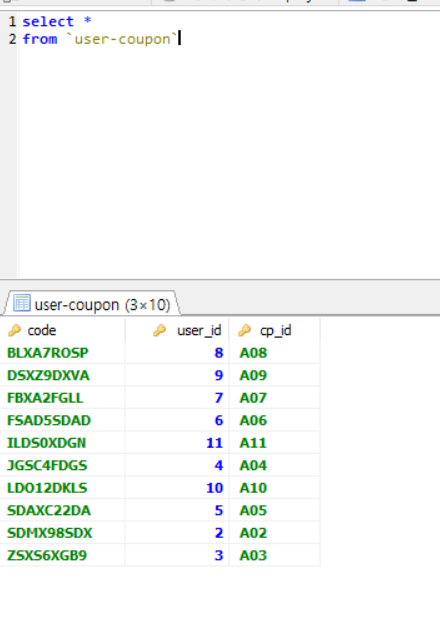
<예시>

delete from `User-Coupon`

where cp\_id='A01';

<예시 결과>





[식당 예약]

<SQL>

insert into Reservation

values(`rsv\_id`, 'user\_id', 'res\_id', 'number\_of\_people', 'time');

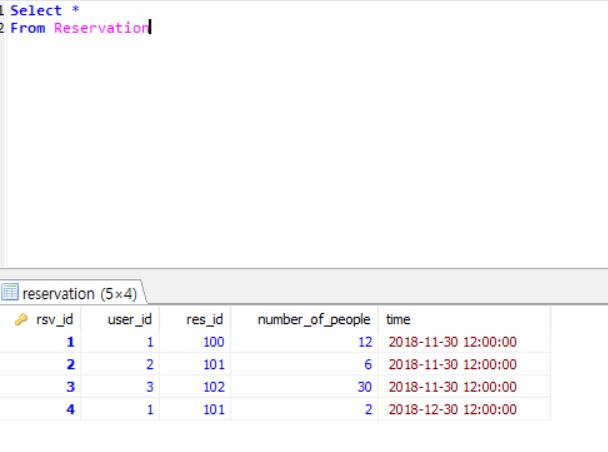
<예시>

insert into Reservation

values(4,1,101,2,'2018-12-30 12:00:00');

<예시 결과>





[본인 식당 피드백]

<SQL>

select name,avg(rate), review

from Restaurant natural join Review

where res\_id='owner\_res\_id'

<예시>

select name,avg(rate), review

from Restaurant natural join Review

where res\_id=101

<예시 결과>



[관리자-식당 추가/제거]

<SQL>

추가: insert into Restaurant values (res\_id, name, category, location);

제거: delete from Restaurant where res\_id='input\_res\_id'

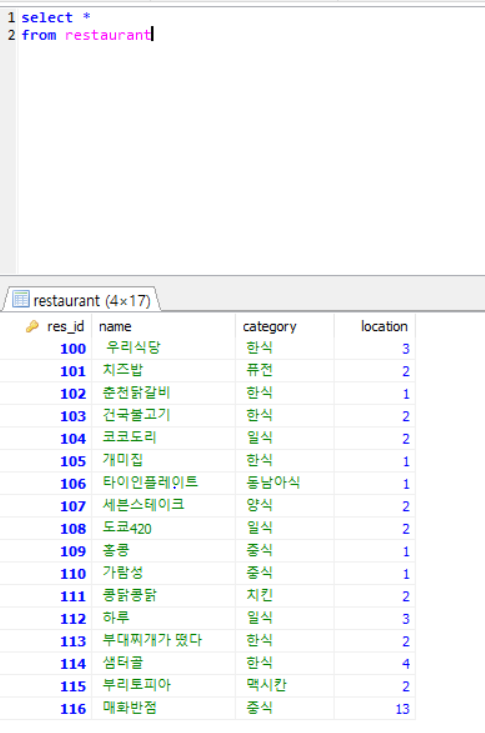
<예시>

insert into Restaurant values (116, ’매화반점’, ’중식’, 13);

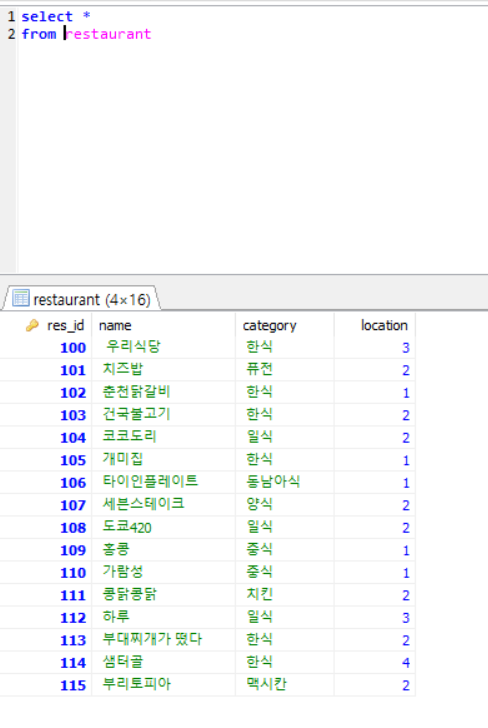
delete from Restaurant where res\_id=116

<예시 결과>









[관리자-메뉴 추가/제거]

<SQL>

INSERT INTO `Menu-Res`(res\_id,menu\_name,price) VALUES('res\_id', 'Menu\_name', 'price')

delete from `Menu-Res` where res\_id='input\_res\_id' and Menu\_name='input\_manu\_name'

<예시>

INSERT INTO `Menu-Res`(res\_id,menu\_name,price) VALUES('115','치킨취킨','77777')

delete from `Menu-Res` where res\_id=115 and Menu\_name='치킨취킨'

<예시 결과>



