

SQL 연습 레포트	
1535030 경제학과 이동희	
<문제1>	
1. 지난 시간에 다운 받은 create.sql을 지난 시간에 생성한 각자의 사용자 계정에서 실행한다.	
<pre>SQL*Plus: Release 11.2.0.1.0 Production on 목 4월 30 19:03:48 2020 Copyright (c) 1982, 2010, Oracle. All rights reserved.  사용자명 입력: donghee 비밀번호 입력:  다음에 접속됨: Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production With the Partitioning, OLAP, Data Mining and Real Application Testing options  SQL&gt; @C:\Users\movelight\Desktop\create.sql</pre>	
2. student 테이블에 ('본인의학번', '본인의주민번호', '본인의이름', 본인의학년, '서울', 920)을 삽입한다.	
<pre>SQL&gt; insert into student(stu_id, resident_id, name, year, address, dept_id) 2 values('1535030', '941012-1002478', '이동희', 4, '서울', 920);  1 개의 행이 만들어졌습니다.</pre>	
3. 아래의 문제를 SQL로 작성한다.	
가. 과목명에 '구조'가 들어 있는 과목번호와 과목명을 찾아라.	
<pre>SQL&gt; select course_id, title 2 from course 3 where title like '%구조%';  COURSE_ID TITLE ----- C102      자료구조 C302      컴퓨터구조</pre>	
나. 2020년도 1학기에 강의가 없는 교수의 이름을 찾아라.	
<pre>SQL&gt; select name 2 from professor 3 where prof_id not in 4 (select distinct prof_id from class where year = 2012 and semester = 1);  NAME ----- 김태석 고희석</pre>	
다. 2012년도 1학기에 한 과목도 수강하지 않은 학생의 학번과 이름, 학과명을 찾아라	
<pre>SQL&gt; select stu_id, name, dept_name 2 from student s, department d 3 where s.dept_id = d.dept_id and stu_id not in (select distinct stu_id from takes);  STU_ID NAME DEPT_NAME ----- 1535030 이동희 컴퓨터공학과 1292501 박철수 전자공학과 1292502 백태성 전자공학과 1292305 김우주 산업공학과</pre>	

라. 학과별 학생수를 찾아라.(학과명, 학생수)

```
SQL> select dept_name, count(*)
  2   from student s, department d
  3   where s.dept_id = d.dept_id
  4   group by dept_name;
```

DEPT_NAME	COUNT(*)
전자공학과	2
컴퓨터공학과	4
산업공학과	3

마. 학번별 수강과목 수를 찾아라.(학번, 수강과목 수)

```
SQL> select stu_id, count(class_id)
  2   from student left outer join takes
  3   using (stu_id)
  4   group by stu_id;
```

STU_ID	COUNT(CLASS_ID)
1292305	0
1292501	0
1292001	3
1292502	0
1292002	3
1292301	1
1292303	3
1292003	2
1535030	0

9 개의 행이 선택되었습니다.

바. 가장 최근에 임용된 교수의 이름과 재직연수를 찾아라

```
SQL> select name, year_emp
  2   from professor
  3   where year_emp >= all (select year_emp from professor);
```

NAME	YEAR_EMP
박철재	2007

사. 같은 학과, 같은 주소를 갖는 학생 이름을 쌍을 찾아라

```
SQL> select s1.name, s2.name
  2   from student s1, student s2
  3   where s1.dept_id = s2.dept_id and s1.address = s2.address and s1.stu_id > s2.stu_id;
```

NAME	NAME
이동희	김광식
김정현	김광식
이동희	김정현

#### <문제2>

(1) 모든 고객의 계좌번호, 이름, 그리고 예금 잔액을 검색하라.

```
SQL> select deposit_num, name, balance
2  from client, deposit
3  where client.ssn = deposit.ssn;
```

DEPOSIT_NUM	NAME	BALANCE
100	김기식	330000
101	강지선	120000
102	이기상	2300000
103	홍순태	560000
104	박지성	870000
105	차두리	9000
106	김기식	110000
107	황현희	1900000
108	홍순태	320000
109	박선희	560000
110	홍순태	1200000

DEPOSIT_NUM	NAME	BALANCE
111	정성태	900000

12 개의 행이 선택되었습니다.

(2) 이름이 '박지성'인 고객의 전화번호와 주민등록번호를 검색하라.

```
SQL> select phone, ssn
2  from client
3  where name = '박지성';
```

PHONE	SSN
010-5910-2312	900402-123571

(3) 지점 이름이 '성남지점'인 지점을 통해 개설된 모든 예금의 잔액을 검색하라.

```
SQL> select balance
2  from deposit
3  where branch_name = '성남지점';
```

BALANCE
2300000
870000
560000

(4) 지점장 이름이 '고희경'인 지점의 이름과 주소를 검색하라.

```
SQL> select branch_name, address
2  from branch
3  where branch_head = '고희경';
```

BRANCH_NAME	ADDRESS
제주지점	제주

(5) 지점 이름이 '광주지점'인 지점의 지점장 이름과 주소를 검색하라.

```
SQL> select branch_head, address
2   from branch
3   where branch_name = '광주지점';
```

BRANCH_HEAD	ADDRESS
김기백	광주

(6) 이름이 '김광식'인 고객이 소유한 예금의 계좌번호, 개설지점의 이름, 잔액을 검색하라.

```
SQL> select deposit_num, branch_name, balance
2   from deposit d, client c
3   where d.ssn = c.ssn and c.name = '김기식';
```

DEPOSIT_NUM	BRANCH_NAME	BALANCE
100	서울지점	330000
106	대구지점	110000

(7) '성남지점'에서 계좌를 개설한 고객의 이름과 주소, 그리고 예금 잔액을 검색하라

```
SQL> select name, address, balance
2   from client, deposit
3   where client.ssn = deposit.ssn and deposit.branch_name = '서울지점';
```

NAME	ADDRESS	BALANCE
김기식	서울	330000
황현희	부산	1900000
홍순태	서울	1200000

(8) '성남지점'에서 계좌를 개설한 고객 중 김씨 성을 가진 고객의 이름과 예금 잔액을 검색하라

```
SQL> select name, balance
2   from deposit, client
3   where deposit.ssn = client.ssn and name like '김%' and branch_name = '성남지점';
```

선택된 레코드가 없습니다.

(9) 예금 잔액이 10만원 이상인 계좌를 소유한 고객의 이름을 검색하라.

```
SQL> select distinct(name)
2   from deposit, client
3   where deposit.ssn = client.ssn and balance >= 100000;
```

NAME
황현희
홍순태
박지성
정성태
강지선
이기상
박선희
김기식

(10) 예금 잔액이 10만원 이상인 계좌가 개설된 지점의 이름과 지점장 이름을 검색하라.

```
SQL> select distinct b.branch_name, branch_head
  2  from branch b, deposit d
  3  where b.branch_name = d.branch_name and d.balance >= 100000;
```

BRANCH_NAME	BRANCH_HEAD
대구지점	김기식
성남지점	박찬주
대전지점	이연희
제주지점	고희경
서울지점	강동희
광주지점	김기백

6 개의 행이 선택되었습니다.

(11) 예금을 개설한 지점의 지점장과 이름이 같은 고객이 소유한 예금의 계좌번호, 잔액, 그리고 개설지점 이름을 검색하라.

```
SQL> select deposit_num, balance, d.branch_name
  2  from client c, deposit d, branch b
  3  where c.ssn = d.ssn and d.branch_name = b.branch_name and c.name = b.branch_head;
```

DEPOSIT_NUM	BALANCE	BRANCH_NAME
106	110000	대구지점

(12) '서울지점'에서 계좌를 개설한 고객들 중에서 남자 고객의 이름과 예금 잔액을 검색하라.

```
SQL> select name, balance
  2  from client, deposit
  3  where client.ssn = deposit.ssn and deposit.branch_name = '서울지점' and deposit.ssn like '%-1%';
```

NAME	BALANCE
김기식	330000
홍순태	1200000

(13) 주민등록번호상의 생일이 3월인 모든 고객의 이름과 소유한 예금의 계좌번호를 검색하라.

```
SQL> select name, deposit_num
  2  from client, deposit
  3  where deposit.ssn = client.ssn and deposit.ssn like '__03%';
```

NAME	DEPOSIT_NUM
홍순태	103
홍순태	108
홍순태	110
정성태	111

(14) 자신의 주소와 같은 지점에 계좌를 소유하고 있는 고객의 이름과 예금 잔액을 검색하라.

```
SQL> select name, balance
2  from client c, deposit d, branch b
3  where c.ssn = d.ssn and d.branch_name = b.branch_name and c.address = b.address;
```

NAME	BALANCE
김기식	330000
홍순태	1200000

(15) '성남지점'과 거래하고 있는 고객의 숫자를 검색하라.

```
SQL> select count(*)
2  from deposit
3  group by branch_name
4  having branch_name = '성남지점';
```

COUNT(*)
3

(16) 각 지점별 잔액의 총합을 검색하라.

```
SQL> select branch_name, sum(balance)
2  from deposit
3  group by branch_name;
```

BRANCH_NAME	SUM(BALANCE)
제주지점	900000
성남지점	3730000
대구지점	119000
서울지점	3430000
대전지점	120000
광주지점	880000

6 개의 행이 선택되었습니다.

(17) 고객 이름별 예금 잔액의 총합을 검색하라.

```
SQL> select name, sum(balance)
2  from client c, deposit d
3  where c.ssn = d.ssn
4  group by name;
```

NAME	SUM(BALANCE)
황현희	1900000
홍순태	2080000
박지성	870000
정성태	900000
강지선	120000
이기상	2300000
박선희	560000
김기식	440000
차두리	9000

9 개의 행이 선택되었습니다.

(18) 잔액의 합이 100만원 이상인 지점 이름과 잔액의 합을 검색하라.

```
SQL> select branch_name, sum(balance)
  2  from deposit
  3  group by branch_name
  4  having sum(balance) >= 1000000;
```

BRANCH_NAME	SUM(BALANCE)
성남지점	3730000
서울지점	3430000

(19) 지점별로 예금 잔액이 100만원 이상인 고객의 숫자를 검색하라.

```
SQL> create or replace view br_ssn as
  2  select branch_name, ssn
  3  from deposit
  4  group by branch_name, ssn
  5  having sum(balance) >= 1000000;
```

뷰가 생성되었습니다.

```
SQL> select branch_name, count(ssn)
  2  from br_ssn
  3  group by branch_name;
```

BRANCH_NAME	COUNT(SSN)
성남지점	1
서울지점	2

(20) 예금 계좌를 소유하고 있지 않은 고객의 이름과 전화번호를 검색하라.

```
SQL> select name, phone
  2  from client
  3  where ssn not in (select ssn from deposit);
```

NAME	PHONE
이영순	010-9876-2323