1. 부질의

SQL : select stu_height
from student
where stu_name = '옥성우';

결과 :

STU_HEIGHT
172

2. 부질의

SQL : select stu_height
from student
where stu_height > 172;

결과 :

STU_HEIGHT
177
188
174

3. 부질의

SQL : select stu_no, stu_name, stu_height
from student
where stu_height >
(select stu_height
from student
where stu_name = '옥성우');

결과 :

STU_NO	STU_NAME	STU_HEIGHT
20213075	옥한빛	177
20212088	조민우	188
20201007	진현무	174

4. SelfJoin

SQL : select a.stu_no, a.stu_name, a.stu_height
from student a, student b
where a.stu_height > b.stu_height
and b.stu_name = '옥성우';

결과 :

STU_NO	STU_NAME	STU_HEIGHT
20213075	옥한빛	177
20212088	조민우	188
20201007	진현무	174

5. 단일열 부질의

SQL : select *
from student

where stu_weight =
(select stu_weight
from student
where stu_name = '박희철');

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20192003	박희철	전기전자	3	B	M		63
20191025	옥성우	컴퓨터정보	3	A	F	172	63

6. 단일열 부질의

SQL : select *
from student
where stu_weight =
(select stu_weight
from student
where stu_name = '박희철')
and stu_name <> '박희철';

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20191025	옥성우	컴퓨터정보	3	A	F	172	63

7. IN연산자 사용

SQL : select *
from student
where stu_class in
(select distinct stu_class
from student
where stu_dept = '컴퓨터정보')
and stu_dept <> '컴퓨터정보';

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20192003	박희철	전기전자	3	B	M		63
20202021	심수정	전기전자	2	A	F	168	45
20212088	조민우	전기전자	1	C	M	188	90
20203054	유가인	기계	2	C	F	154	47
20213088	이태연	기계	1	C	F	162	50
20213075	옥한빛	기계	1	C	M	177	80

8. 부질의

SQL : select *
from student
where stu_height >
(select avg(stu_height)
from student);

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	옥한빛	기계	1	C	M	177	80
20212088	조민우	전기전자	1	C	M	188	90
20201007	진현무	컴퓨터정보	2	A	M	174	64
20191025	옥성우	컴퓨터정보	3	A	F	172	63

9. 부질의

SQL : select *
from student
where stu_height > all
(select avg(stu_height)
from student
group by stu_dept);

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20212088	조민우	전기전자	1	C	M	188	90

10. 부질의

SQL : select stu_dept, min(stu_height)
from student
group by stu_dept having min(stu_height) >
(select min(stu_height)
from student
where stu_dept = '컴퓨터정보');

결과 :

STU_DEPT	MIN(STU_HEIGHT)
전기전자	168

11. 부질의와 Join

SQL : select *
from student
where stu_no in
(select stu_no
from enrol
where sub_no = 101);

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20191001	김종헌	컴퓨터정보	3	C	M		72
20191025	옥성우	컴퓨터정보	3	A	F	172	63

12. 부질의와 Join

SQL : select *
from student a, enrol b
where a.stu_no = b.stu_no and b.sub_no = 101;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT	SUB_NO	STU_NO_1	ENR_GRADE
20191001	김종헌	컴퓨터정보	3	C	M		72	101	20191001	80
20191025	옥성우	컴퓨터정보	3	A	F	172	63	101	20191025	65

13. 101번 수강 과목 학생 정보 검색(부질의 사용 불가능)

SQL : select a.stu_no, a.stu_name, b.enr_grade
from student a, enrol b
where a.stu_no = b.stu_no and b.sub_no = 101;

STU_NO	STU_NAME	ENR_GRADE
20191001	김종헌	80
20191025	옥성우	65

14. 복수열 부질의 준비

SQL : create table test(empno, ename, sal, comm, deptno)
as
select empno, ename, sal, comm, deptno
from emp
where deptno = 1;

결과 :

15. 테이블 검색

SQL : select *
from test;

EMPNO	ENAME	SAL	COMM	DEPTNO
11	apple	1000		30
12	banana	2000	100	30
13	cheese	1000	0	10
14	dog	2000		20
15	egg	1000	100	20

16. 복수열 부질의 결과값

SQL> : select *
from test
where (sal, nvl(comm, -1)) =
(select sal, nvl(comm,-1)
from test
where empno = 11);

결과 :

EMPNO	ENAME	SAL	COMM	DEPTNO
11	apple	1000		30

17. 복수열 복수행 부질의 결과값

SQL : select *
from test
where (sal , nvl(comm, -1)) in (select sal , nvl(comm, -1)
from test
where deptno = 30);

결과 :

EMPNO	ENAME	SAL	COMM	DEPTNO
11	apple	1000		30
12	banana	2000	100	30

18. From절 In-Line

SQL : select stu_no, stu_name, a.stu_dept, stu_height, avg_height
from student a, (select stu_dept, round(avg(stu_height),2) as avg_height
from student
group by stu_dept;
) b
where a.stu_dept = b.stu_dept and stu_height > avg_height;

결과 :

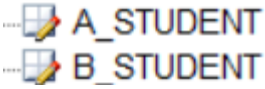
STU_NO	STU_NAME	STU_DEPT	STU_HEIGHT	AVG_HEIGHT
20213075	옥한빛	기계	177	164.33
20212088	조민우	전기전자	188	178
20201007	진현무	컴퓨터정보	174	170.67
20191025	옥성우	컴퓨터정보	172	170.67

19. 학생 테이블 2개 생성

SQL : create table a_student
as select *
from student
where stu_deptin ('기계', '전기전자');

SQL : create table b_student
as select *
from student
where stu_deptin ('전기전자', '컴퓨터정보');

결과 :



20. 내용 확인

SQL : select * from a_student;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50
20203054	유가인	기계	2	C	F	154	47
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63

SQL : select * from b_student;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63
20211062	김인중	컴퓨터정보	1	B	M	166	67
20201007	진현무	컴퓨터정보	2	A	M	174	64
20191001	김종헌	컴퓨터정보	3	C	M		72
20191025	옥성우	컴퓨터정보	3	A	F	172	63

21. Union

SQL : select * from a_student
union
select * from b_student;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20191001	김종헌	컴퓨터정보	3	C	M		72
20191025	옥성우	컴퓨터정보	3	A	F	172	63
20192003	박희철	전기전자	3	B	M		63
20201007	진현무	컴퓨터정보	2	A	M	174	64
20202021	심수정	전기전자	2	A	F	168	45
20203054	유가인	기계	2	C	F	154	47
20211062	김인중	컴퓨터정보	1	B	M	166	67
20212088	조민우	전기전자	1	C	M	188	90
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50

22. Union

SQL : select * from a_student
union all
select * from b_student;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50
20203054	유가인	기계	2	C	F	154	47
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63
20211062	김인중	컴퓨터정보	1	B	M	166	67
20201007	진현무	컴퓨터정보	2	A	M	174	64
20191001	김종헌	컴퓨터정보	3	C	M		72
20191025	옥성우	컴퓨터정보	3	A	F	172	63

23. Intersect

SQL : select * from a_student
intersect
select * from b_student;

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20192003	박희철	전기전자	3	B	M		63
20202021	심수정	전기전자	2	A	F	168	45
20212088	조민우	전기전자	1	C	M	188	90

24. Minus

SQL : select * from a_student
minus
select * from b_student;

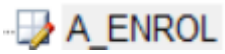
결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20203054	유가인	기계	2	C	F	154	47
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50

25. 단일 튜플 삽입

SQL : create table a_enrol
as select *
from enrol
where stu_no < 20210000;

결과 :



26. a_enrol 테이블의 구조(desc) 및 데이터 확인(select)

SQL : desc a_enrol;

결과 :

NAME	NULLABLE	TYPE	DEFAULT	COMMENT
SUB_NO		CHAR(3)		
STU_NO		NUMBER(9)		
ENR_GRADE		NUMBER(3)		

SQL : select *
from a_enrol;
결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41

27. a_enroll 테이블에 데이터 삽입
SQL : insert into a_enrol(sub_no, stu_no, enr_grade)
values (108, 20211062, 92);
SQL : insert into a_enrol
values (109, 20212088, 85);
추가 후 결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41
108	20211062	92
109	20212088	85

28. 일부 컬럼만 값이 존재하는 경우 생략 불가능
SQL : insert into a_enrol(sub_no, stu_no)
values (110, 20212088);
SQL : select *
from a_enrol;
결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41
108	20211062	92
109	20212088	85
110	20212088	

29. NULL값의 명시적 표현

SQL : insert into a_enrol
values(111, 20213075, null);

SQL : select *
from a_enrol;

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41
108	20211062	92
109	20212088	85
110	20212088	
111	20213075	

30. 복수 행 삽입

SQL : insert into a_enrol
select * from enrol
where stu_no like '2021%';

SQL : select * from a_enrol

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41
108	20211062	92
109	20212088	85
110	20212088	
111	20213075	
103	20212088	45
108	20211062	81
102	20213075	66
105	20213075	56
102	20213088	61
105	20213088	78

31. 전체 데이터에 대한 변경 (Update)

SQL : select * from a_enrol;

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	80
104	20191001	56
106	20192003	72
101	20191025	65
104	20191025	65
107	20203054	41
108	20211062	92
109	20212088	85
110	20212088	
111	20213075	
103	20212088	45
108	20211062	81
102	20213075	66
105	20213075	56
102	20213088	61
105	20213088	78

SQL : update a_enrol

set enr_grade = enr_grade + 5;

SQL select * from a_enrol;

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	61
106	20192003	77
101	20191025	70
104	20191025	70
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

32. 조건에 맞는 데이터 변경

SQL : select * from a_enrol;

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	61
106	20192003	77
101	20191025	70
104	20191025	70
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

SQL : update a_enrol
set enr_grade = enr_grade + 10
where sub_no = 104 ;

SQL : select * from a_enrol;

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	71
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

33. 부질의를 갖는 UPDATE문

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	71
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

```
SQL> update a_enrol
        set enr_grade = enr_grade + 10
        where sub_no = (select sub_no
        from subject
        where sub_name = '시스템 분석설계');
```

결과 :

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	71
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

34. 특정 튜플 삭제 (Delete)

SUB_NO	STU_NO	ENR_GRADE
101	20191001	85
104	20191001	71
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

SQL : delete from a_enrol
where stu_no = 20191001;

결과 :

SUB_NO	STU_NO	ENR_GRADE
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

35. 부질의를 갖는 삭제

SUB_NO	STU_NO	ENR_GRADE
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

SQL : delete from a_enrol
where stu_no =(select sub_no
from subject
where sub_name = '기계요소설계');

SQL : select * from a_enrol;

결과 :

SUB_NO	STU_NO	ENR_GRADE
106	20192003	77
101	20191025	70
104	20191025	80
107	20203054	46
108	20211062	97
109	20212088	90
110	20212088	
111	20213075	
103	20212088	50
108	20211062	86
102	20213075	71
105	20213075	61
102	20213088	66
105	20213088	83

36. 테이블내의 모든 데이터가 삭제

SQL : delete from a_enrol;

결과 : (SQLTools for Oracle)

SQL :❖drop table (테이블 제거는 DDL영역)

37. 학생(student) 테이블의 내용을 a_student로 수정 또는 삽입하는 MERGE 연산

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50
20203054	유가인	기계	2	C	F	154	47
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63

SQL : merge into a_student a
using student s
on (s.stu_no = a.stu_no)
when matched then
update set a.stu_weight = s.stu_weight
when not matched then
insert values(s.stu_no, s.stu_name, s.stu_dept, null,
null, null, null, s.stu_weight);

결과 :

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	옥한빛	기계	1	C	M	177	80
20213088	이태연	기계	1	C	F	162	50
20203054	유가인	기계	2	C	F	154	47
20212088	조민우	전기전자	1	C	M	188	90
20202021	심수정	전기전자	2	A	F	168	45
20192003	박희철	전기전자	3	B	M		63
20191001	김종현	컴퓨터정보					72
20191025	옥성우	컴퓨터정보					63
20201007	진현무	컴퓨터정보					64
20211062	김인중	컴퓨터정보					67