FROM book where bookname like '\_구%';--2번째 문자에 '구' 가 들어가는 북네임

- \_ : 1개의 문자와 일치(

- %구% : 앞뒤에 0개 이상의 문자열과 일치

별칭 부여하기(as)

Select stu\_no as Id, stu\_name as name from student;

|| 두 개 의 속성을 붙여서 출력한다.

SELECT stu\_dept || stu\_name FROM student

SELECT stu\_dept||'과 '||stu\_name||' 입니다' FROM student;

속성 사이에 문자열 포함.

Order by 정렬

오름차순으로 정렬할 경우 생략가능(ASC)

내림차순으로 정렬할 경우 DESC를 뒤에 붙인다.

SELECT \* FROM Book ORDER BY price DESC, publisher ASC;

SELECT

stu\_no,stu\_weight as target

FROM student

order by target;

SELECT

stu\_no,stu\_weight as target

FROM student

order by 2; 타겟 대신 타겟의 열 번호를 써도 됨.

select UPPER('korea') from dual;

select initcap('korea') from dual;--첫글자만 대문자.

select LOWER('KOREA')from dual;

select CONCAT('korea','japan')from dual;

select SUBSTR('korea',2,3)from dual;

select sysdate from dual; 오늘 날짜 반환

alter session set nls\_date\_format='mm-dd-yyyy' 날짜 반환 형식 변환

select next\_day(sysdate,2) from dual ->다음주 n요일(1=일 2=월 3=화…)

select round(sysdate,'MON') from dual

====변환 함수=====

To\_number() = 문자를 숫자로

To\_date() = 문자를 날짜로

To\_char() = 숫자와 날짜를 문자로.

NVL(인수1 ,인수2)

:인수값이 NULL인지 체크하여 연산이 가능한 값으로 변환하여 반환하는 함수

Ex)인수1이 널이면 인수2

인수2가 널이면 인수1을 반환한다.

Select nvl(stu\_height,0) from student

Stu\_height가 널이면 0을 반환.

SELECT count(stu\_class) FROM student;

SELECT count(\*) FROM student;

Count(\*) : 전체 레코드(튜플 )의 수

SELECT

stu\_gender,avg(stu\_weight)

FROM student

group by stu\_gender;

그룹핑 젠더(F와 M)을 기준으로 묶는다.

SELECT stu\_grade,

avg(stu\_height)

FROM student

where stu\_dept='기계'

group by stu\_grade

having avg(stu\_height)>=120;

그룹후 조건문은 where이 아니라 having

SELECT

rpad(stu\_name,10,'&')

FROM student

Ex)옥성우&&&&

조인

SELECT \* FROM customer,orders

where orders.custid = customer.custid;

커스터머와 오더의 id를 조인

SELECT

student.stu\_no,stu\_name,enr\_grade

FROM student,enrol

where student.stu\_no=enrol.stu\_no;

다른 테이블의 같은 속성을 합친다.

\*\*이퀴조인(equi join)

where절에서 ‘=’ 연산자 사용

\*\*natural join

Select stu\_no, stu\_name, enr\_grade

From student natural join enroll

=가 필요 없음 알아서 조인 시켜줌