

# *Writing Basic SQL*

## *SELECT Statements*

---

기본 **SELECT** 문

산술식

열명(Alias)정의

연결 연산자

리터럴 문자열

중복행 제거

실습

# 기본 SELECT 문

```
SELECT * | {[DISTINCT] column|expression [alias],...}  
FROM    table;
```

SELECT : 표시할 대상 열을 지정한다.

FROM : 대상 열을 포함하는 해당 테이블을 지정한다.

```
SELECT department_id, location_id  
FROM departments;
```

수식 연산자를 사용하여 **number** 및 **date** 데이터 상의 표현식을 작성한다.

Operator	Description
+	Add
-	Subtract
*	Multiply
/	Divide

```
SELECT last_name, salary, 12*salary+100  
FROM employees;
```

Null 값을 포함한 산술식의 결과는 null이다.

```
SELECT last_name, 12*salary*commission_pct  
FROM employees;
```

# 별명(Alias) 정의

- 컬럼의 이름을 재명명한다.
- 계산에 유용하다.
- 컬럼 명 바로 뒤에 별명을 정의하여야 하며, 컬럼 명과 별명 사이의 키워드 **AS**는 써도 되고 안 써도 된다.
- 만약 빈 칸 또는 특수 문자가 있거나 대소문자를 구별한다면, “ ”가 필요하다.

```
SELECT last_name "Name", salary*12 "Annual Salary"  
FROM employees;
```

# 연결 연산자, 리터럴 문자열

- 컬럼이나 문자열을 다른 컬럼에 연결한다.
- 두 개의 버티컬바(II)로 표시한다.
- 결합된 컬럼이 문자 형태로 나타난다.

```
SELECT last_name || job_id AS "Employees"  
FROM   employees;
```

- 리터럴은 **SELECT** 절에 포함된 문자, 수식, 또는 숫자 등을 말한다.
- 날짜 그리고 문자 리터럴 값은 ' ' 안에 들어가야 한다.
- 각 행에 대하여 리터럴 문자열이 적용된다.

```
SELECT last_name || ' is a ' || job_id AS "Employee Details"  
FROM   employees;
```

# 중복행 제거

- 질의는 기본적으로 중복된 행을 포함하여 모든 결과를 보여준다.
- **SELECT** 절에 **DISTINCT**라는 키워드를 사용하여 중복된 행을 제거한다.

```
SELECT DISTINCT department_id  
FROM employees;
```

- 테이블의 구조를 표시하기 위하여 **DESCRIBE** 명령어를 사용한다.

```
DESCRIBE employees
```

# 실습 - 1

1. 다음 명령문의 오류는 무엇인가?

```
SELECT  employee_id, last_name sal x 12 ANNUAL SALARY  
FROM    employees ;
```

2. DEPARTMENTS 테이블의 구조를 표시하고 테이블의 모든 데이터를 검색한다.
3. EMPLOYEES 테이블의 내용을 검색시 사원번호, 사원이름, 업무코드, 입사일  
일이 오도록 질의를 작성한다. HIRE\_DATE 열에 STARTDATE라는 별칭을  
지정한다.
4. 이름과 JOB\_ID를 연결한 다음 쉼표 및 공백으로 구분하여 표시하고, 열이름  
을 Employee and Title로 지정한다.