

# 프로젝트 보고서\_3주차 (개념)

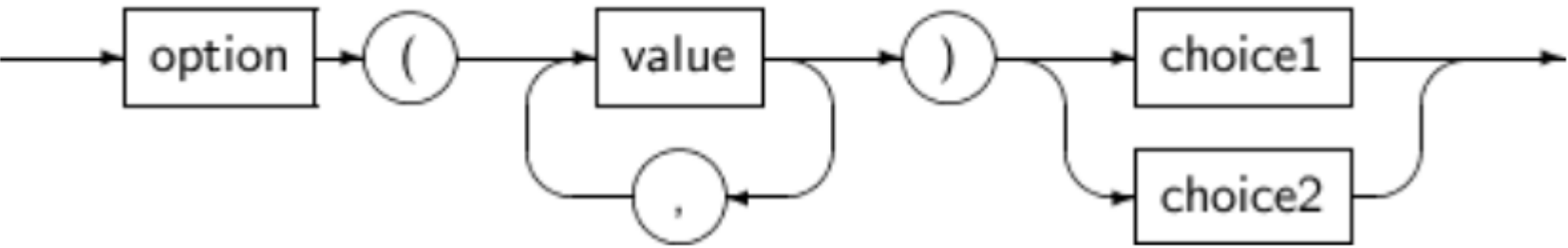
## 제3장. SQL 연산

### 3.1. 개요

#### 연산식

- 구조 : 연산자(Operator), 피연산자(Operand), 함수(Function)으로 구성
- 종류
  - 일반 연산식 : 논리/비교 연산자를 제외한 모든 연산식
  - 조건식 : 논리/비교 연산자
- 문법

[그림 3.1] 연산식 문법  
example



### 3.2. 연산자

#### 3.2.1. 일반 연산자

- 산술 연산자 : 사칙연산을 수행하기 위한 연산 (덧셈, 뺄셈, 곱셈, 나눗셈, 양수, 음수)
- 문자열 연산자 : 접한 연산자( || )
- 집합 연산자 : 두 개의 질의 결과에 대한 연산자 → 결과: 하나의 테이블

연산자	연산 결과
UNION	두 질의의 결과를 하나로 합친다. (중복X) → 교환법칙 성립
UNION ALL	두 질의의 결과를 하나로 합친다. (중복O) → 교환법칙 성립
INTERSECT	두 질의의 결과를 공통적으로 포함된 로우를 추출(중복X) → 교환법칙 성립
MINUS	앞 질의로부터 뒤 질의의 결과에 포함된 모든 로우를 제거 (중복X) → 교환법칙 성립 X

#### 3.2.2. 조건식에 포함되는 연산자

- 논리 연산자(피연산자) : 조건식 → NOT(단항 연산자), AND, OR (이항 연산자)
- 결과 : TRUE, FALSE, UNKNOWN
  - NOT 연산자의 진리표

NO	TRUE	FALSE	UNKNOWN
	FALSE	TRUE	UNKNOWN

- AND 연산자의 진리표

AND	TRUE	FALSE	UNKNOWN
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

- OR 연산자의 진리표

OR	TRUE	FALSE	UNKNOWN
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	UNKNOWN
UNKNOWN	TRUE	UNKNOWN	UNKNOWN

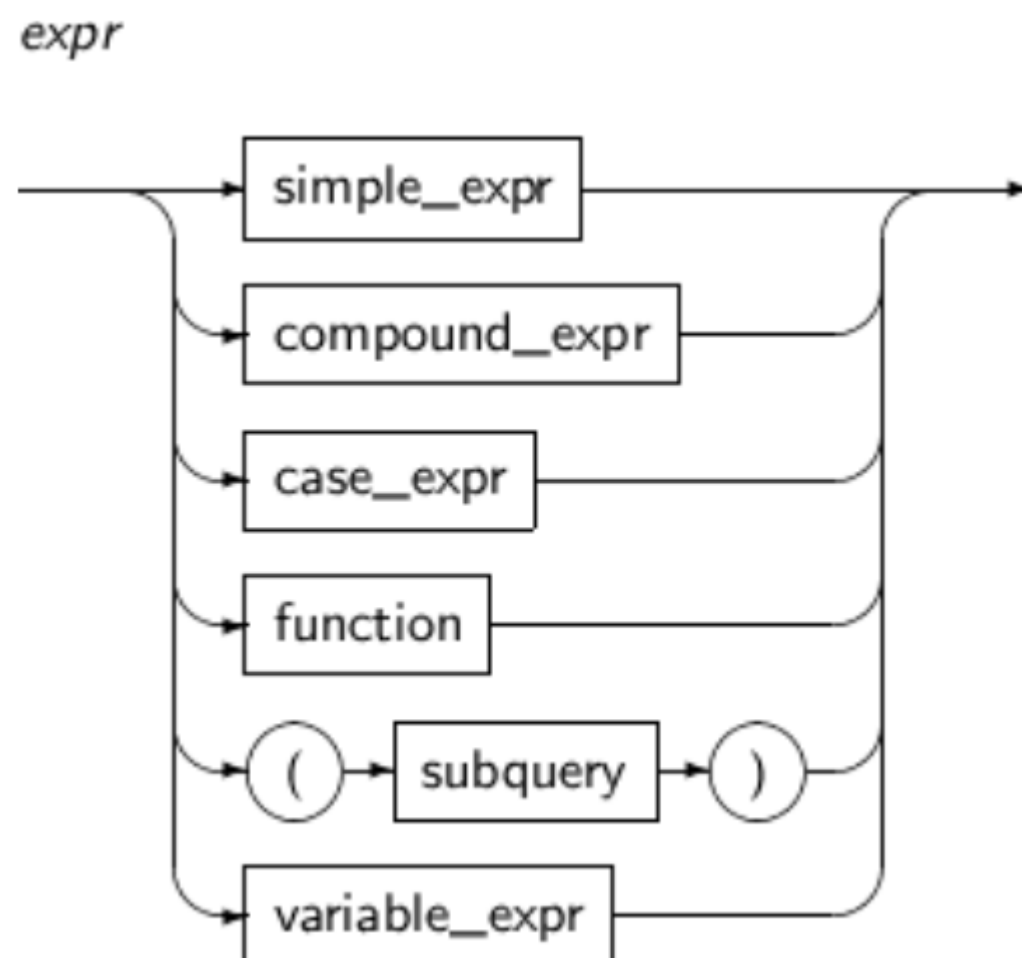
- 비교 연산자(피연산자) : 임의의 산술 값이나 문자열, 리스트 등이 올 수 있다.

연산자	설명
=	두 값이 서로 같은지
!=, ^=, ~=, <>	두 값이 서로 다른지
>, <	두 값의 대소비교를 통해 크거나 작는지
>=, ≤	두 값의 대소비교를 통해 크거나 같은지 또는 작거나 같은지

- 연산자 우선순위
1. +, - (숫자부호, 단항 연산자)
  2. \*, / (산술 연산자)
  3. +, - (산술 연산자)
  4. =, !=, ^=, ~=, <>, <, >, <=, ≥ (비교 연산자)
  5. LIKE, BETWEEN, IN, EXISTS, IS NULL (비교 연산자)
  6. NOT (논리 연산자)
  7. AND (논리 연산자)
  8. OR (논리 연산자)

### 3.3. 연산식

- 문법

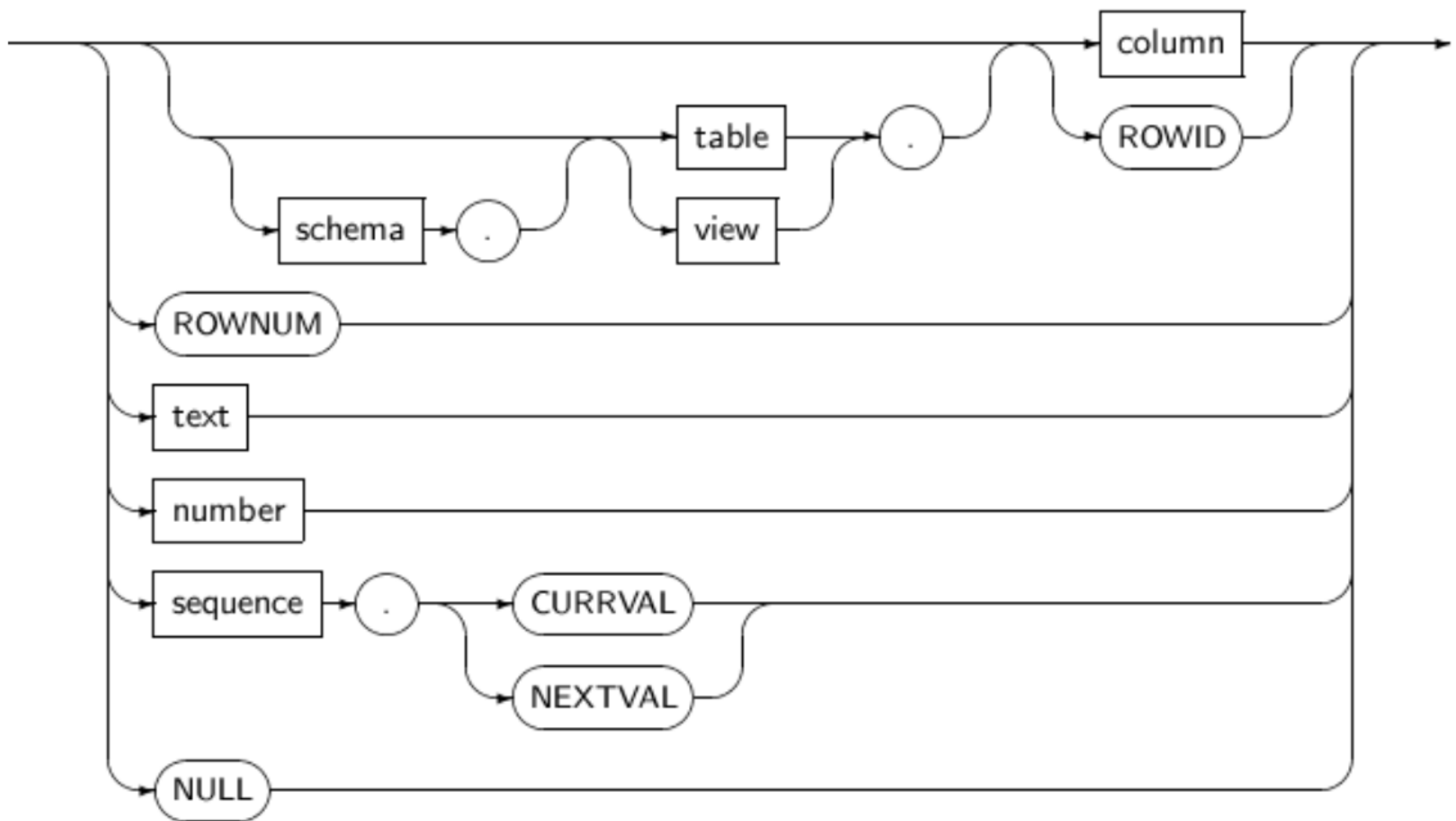


#### 3.3.1. 단순 연산식

- 하나의 피연산자만으로 이루어진 연산식
- EX) 특정 테이블 칼럼, ROWID 타입, 문자열 또는 수치값, NULL값

- 문법

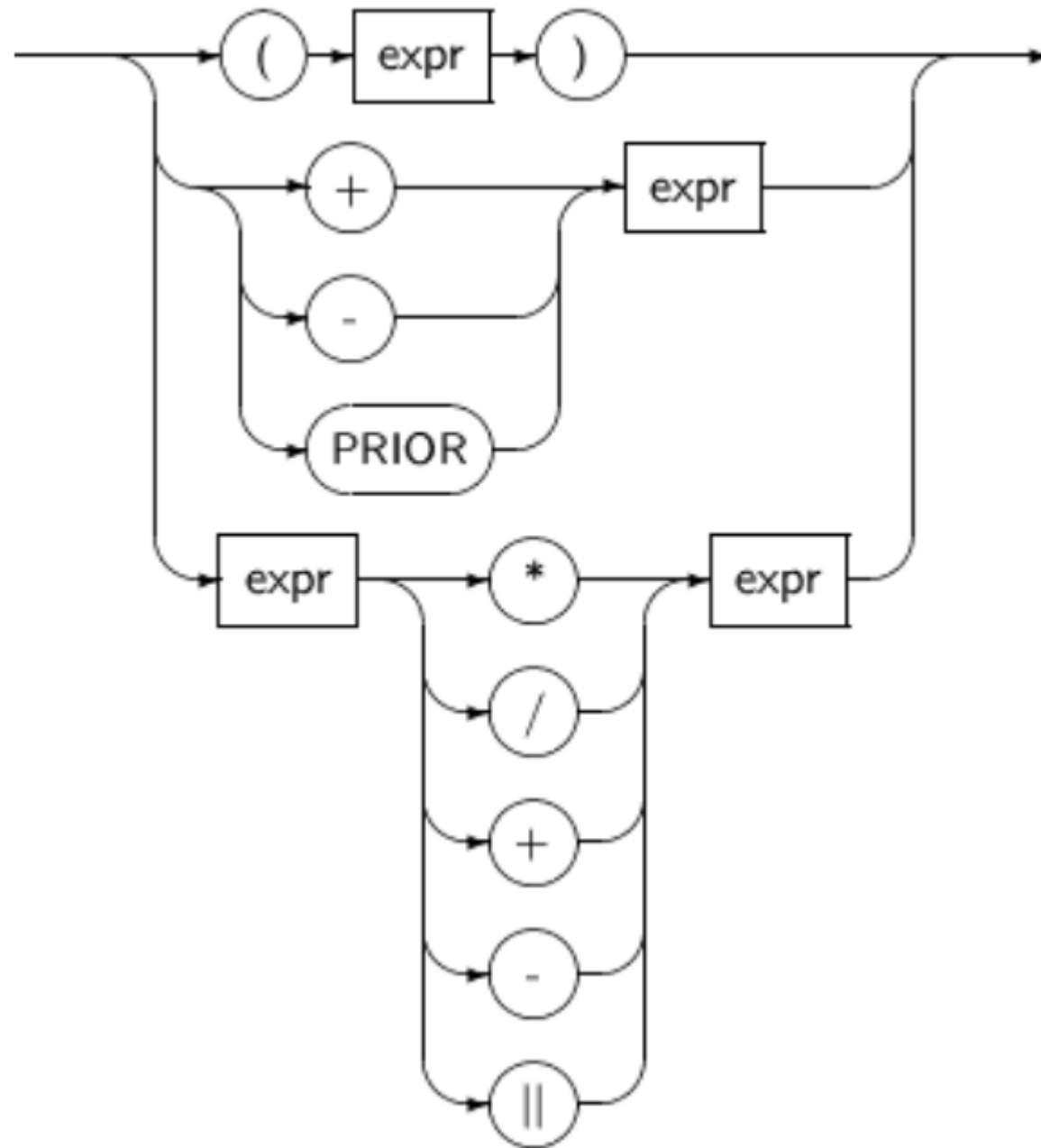
*simple\_expr*



### 3.3.2. 복합 연산식

- 하나 이상의 연산식을 결합한 복잡한 형태의 연산식
- 문법

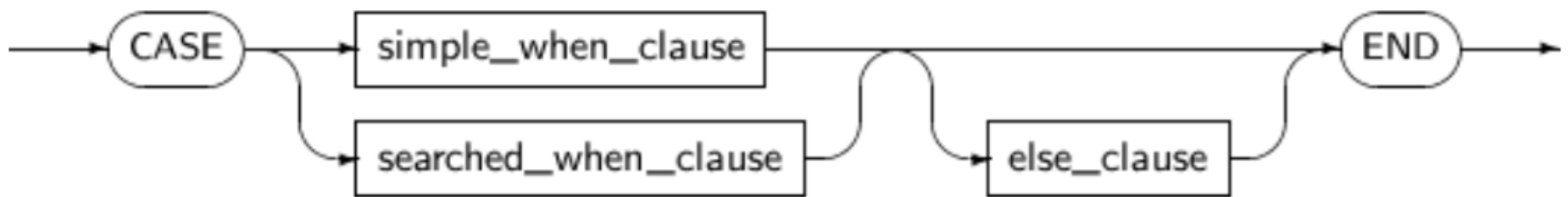
*compound\_expr*



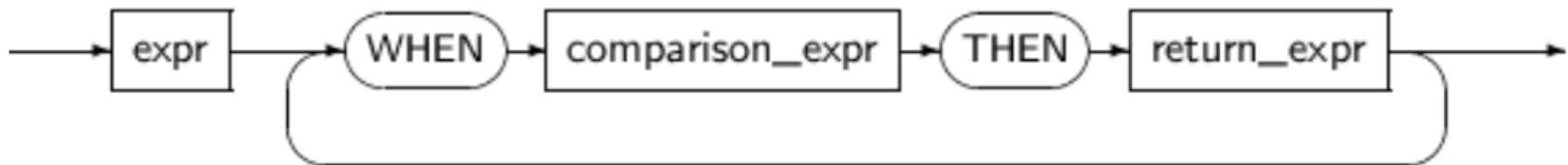
### 3.3.3. CASE 연산식

- IF...THEN...ELSE 로직 표현
- 최대 65535까지 표현 가능
- 문법

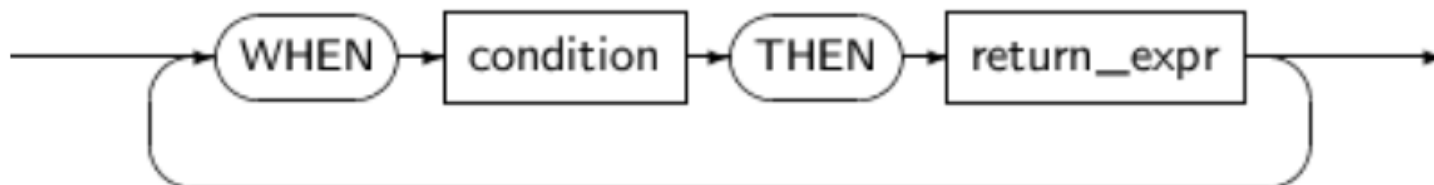
*case\_expr*



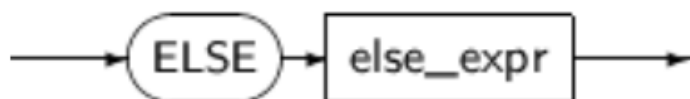
*simple\_when\_clause*



*searched\_when\_clause*



*else\_clause*



### 3.3.4. 함수

- 0개 이상의 파라미터로 구성
- 단일 로우 함수 & 집단 함수
- ROUND(숫자,소수점), LENGTH(), SYSDATE, AVG()

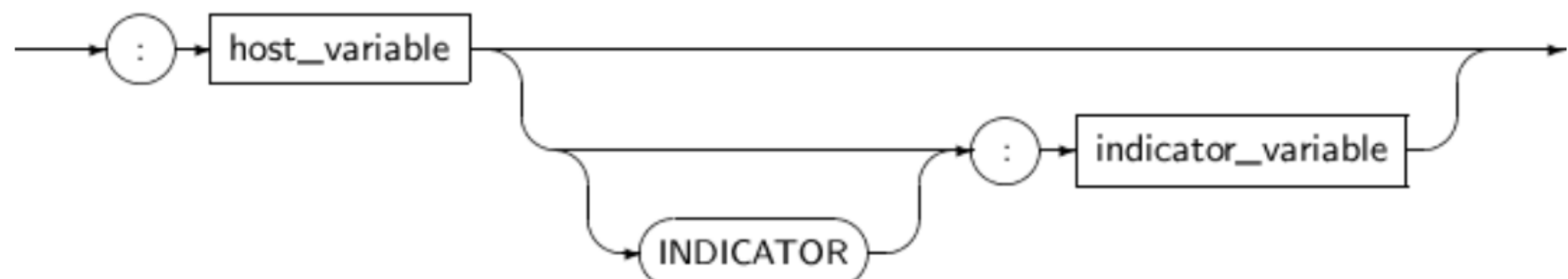
### 3.3.5. 부질의 연산식

- 연산식 내에 SELECT문이 포함된 것 → 반드시 1개의 칼럼 값을 갖는 1개의 로우
- 부질의 연산식 올 수 없는 위치
  - SELECT문의 GROUP BY절
  - 칼럼의 디폴트값
  - CHECK 제약조건의 조건식

### 3.3.6. 변수

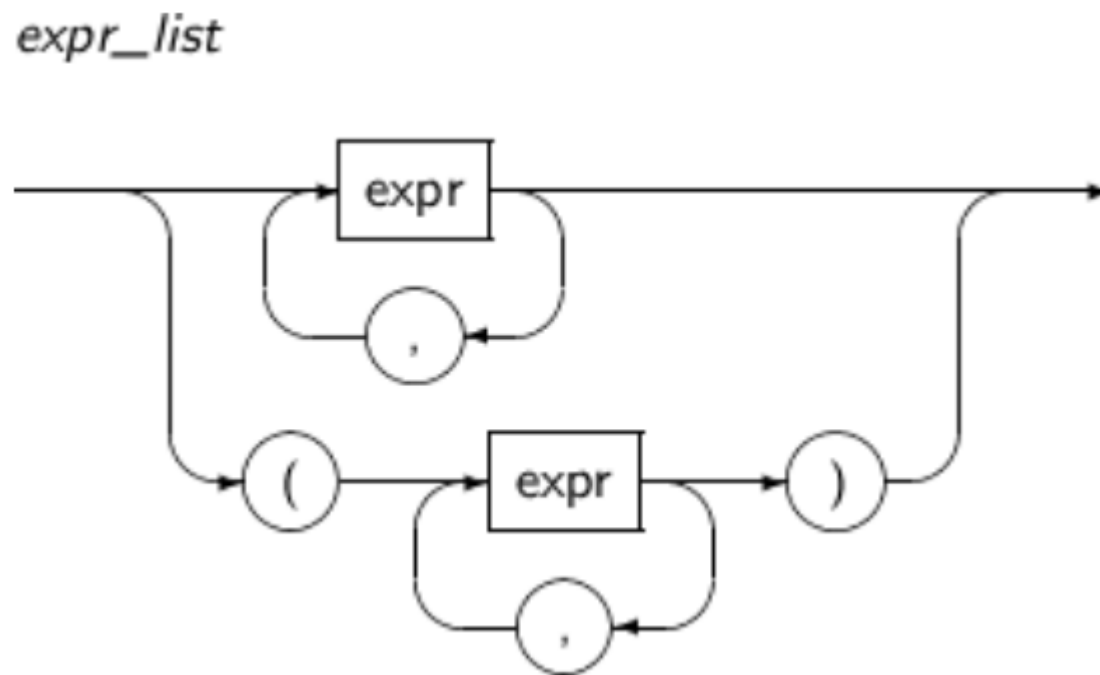
- 호스트 변수, 지시자, 지시자 변수로 구성
- tSQL 문장에 사용하여 프로그램의 입출력에 사용
- 문법

*variable\_expr*



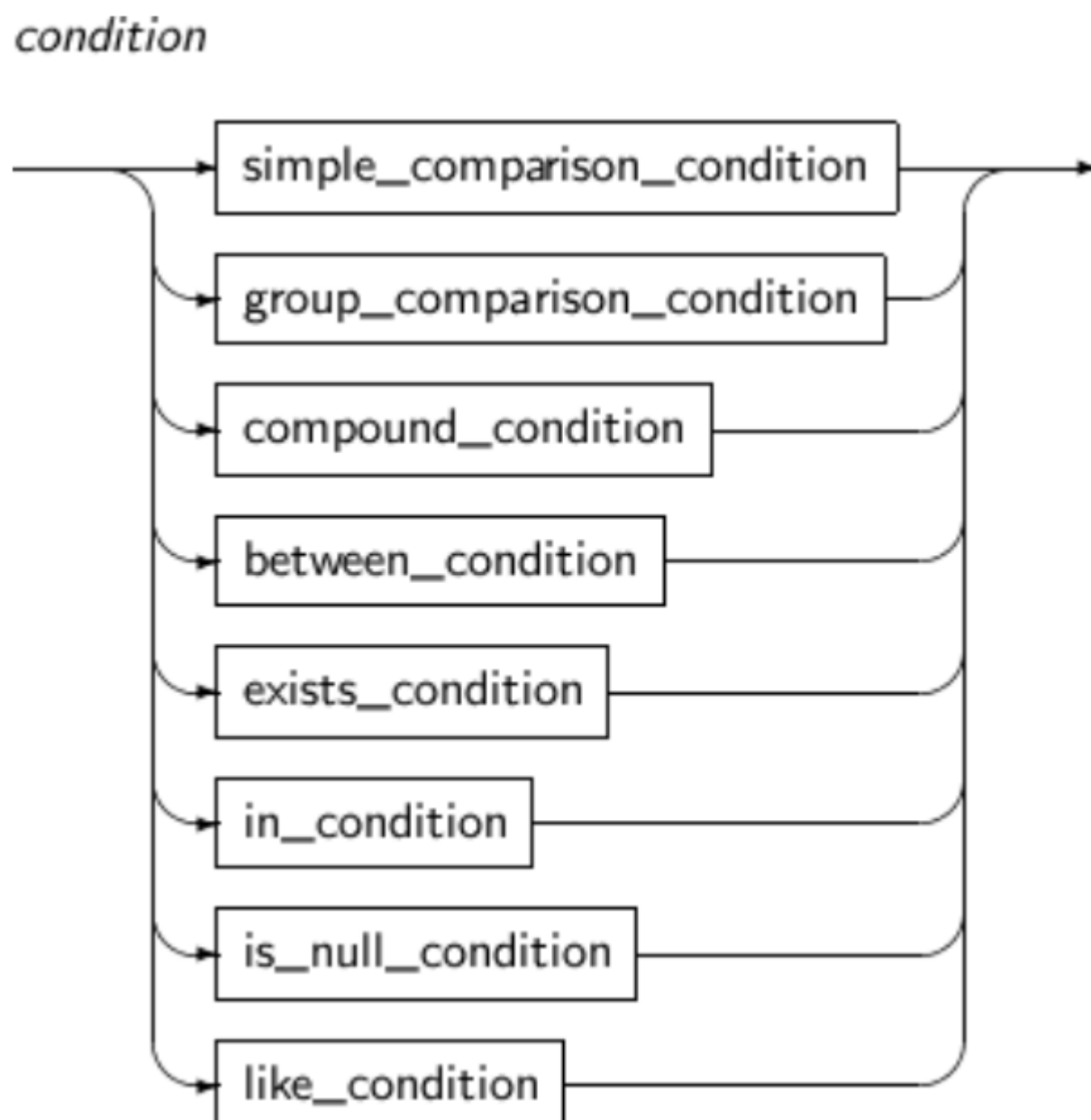
### 3.3.7. 리스트

- 리스트는 하나 이상의 연산식을 쉼표(,)로 구분하여 나열
- 문법



### 3.4.조건식

- 위치
  - SELECT,UPDATE,DELETE문장의 WHERE절
  - SELECT문의 HAVING절
- 종류
  - 단순 조건식, 그룹 조건식, 복합 조건식,비교 조건식
- 문법



### 3.4.1. 단순 조건식

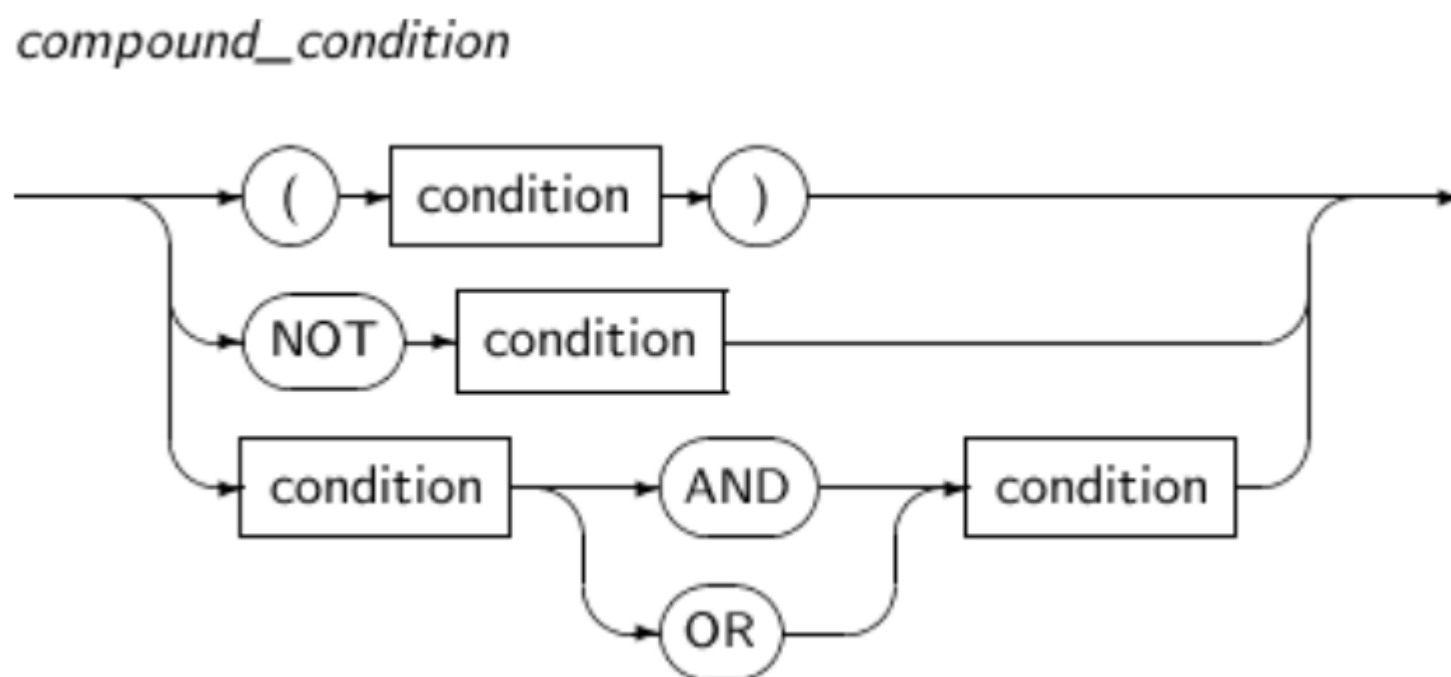
- 단순 조건식은 두 개의 데이터 값을 비교하는 조건식
- 피연산자 중의 하나라도 NULL이 포함된 단순 조건식은 항상 UNKNOWN을 반환
- 비교 연산자 : =, !=, ^=, ~=, <>, <, >, >=, ≤

### 3.4.2. 그룹 조건식

- 하나의 데이터 값과 리스트에 포함된 값을 비교하는 연산 수행
- ANY(), ALL(), SOME()

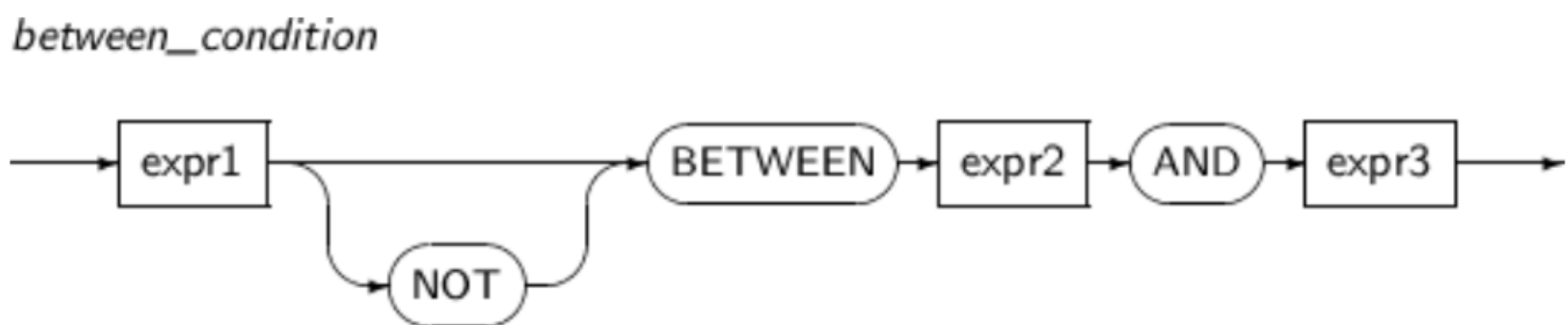
### 3.4.3. 복합 조건식

- 하나 이상의 조건식을 결합한 복잡한 형태의 조건식
- 괄호로 묶은 조건식, NOT 연산자를 붙인 연산식, 논리 연산자(AND, OR)로 결합한 조건식
- 문법



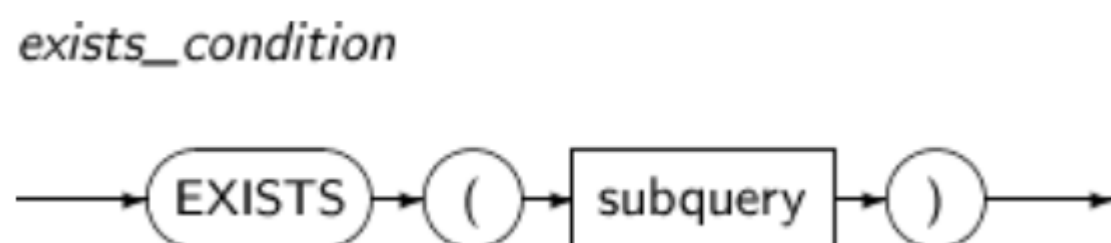
### 3.4.4. BETWEEN 조건식

- 문법



### 3.4.5. EXISTS 조건식

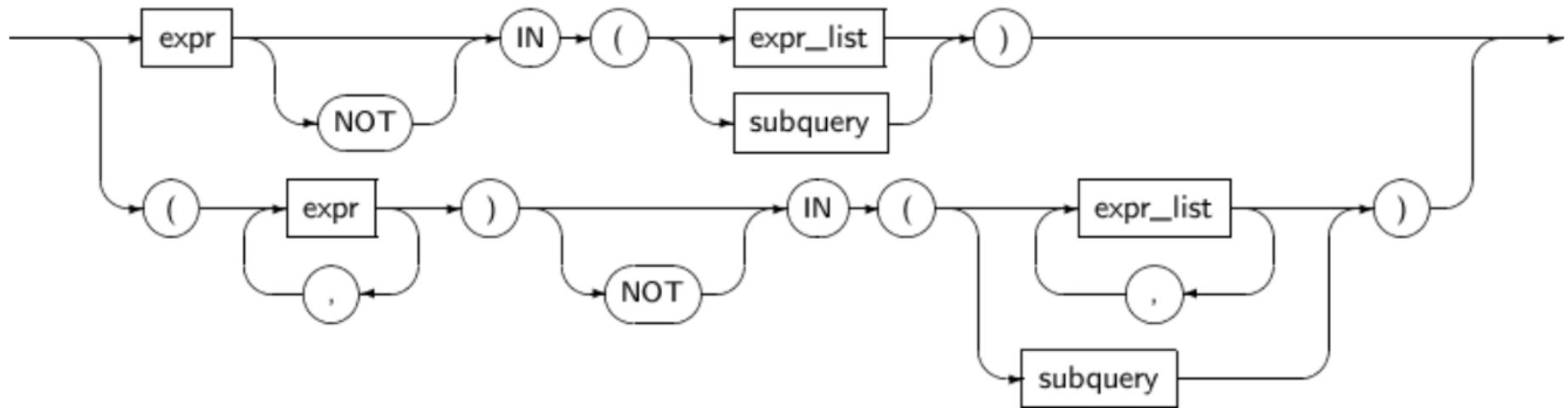
- 오른쪽 부질의를 실행한 결과가 하나 이상의 로우를 반환하면 True, 그렇지 않으면 False
- 문법



### 3.4.6. IN 조건식

- 왼쪽의 데이터 값이 오른쪽의 리스트 내에 포함되어 있는지 비교하는 연산자
- 문법

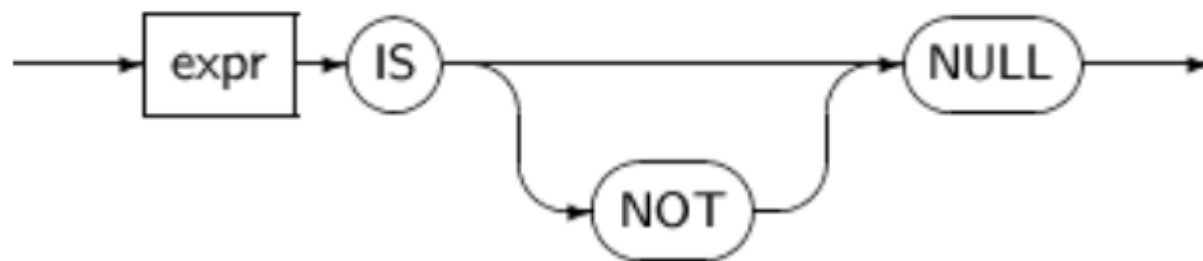
*in\_condition*



### 3.4.7. IS NULL (IS NOT NULL)조건식

- 왼쪽의 연산식의 결과가 NULL이면 True, 그렇지 않으면 False를 반환
- 문법

*is\_null\_condition*



### 3.4.8. LIKE 조건식

- 문법

*like\_condition*



### 3.4.8. REGEXP\_LIKE 조건식

- 정규표현식으로 된 문자열 패턴을 비교한다는 점을 제외하고는 LIKE와 동일

## 제 4장. 함수

### 4.1. 개요

#### 4.1.1. 단일 로우 함수

- 하나의 로우로부터 컬럼 값을 파라미터로 입력 받은 함수

#### 4.1.2. 집단 함수

- 하나 이상의 로우로부터 컬럼 값을 파라미터로 입력받는 함수



### 4.1.3. 분석 함수

- 특정 로우 그룹에 대한 집계 값을 구하는데 사용
- 각각의 로우에 대해 로우 그룹이 별개로 정의되며 별개로 각각 자신의 로우 그룹에 대한 집계 값을 갖게 된다.
  - OVER analytic\_clause
  - partition by
  - order\_by\_clause
  - window\_clause