

Database Management Essentials

Basic Query Formulation with SQL

Summarized by
github.com/dgkim5360
dgkim5360.tistory.com

SQL Overview

Structured Query Language

A computer language
containing statements of database
definition / control / manipulation

Context

- Stand-alone: Workbench
- Embedded: Django App

Weakness

표준화가 엉망이다: Higher switching cost

Major SQL Statements

Statement	Statement Type
CREATE TABLE	Definition, Control
CREATE VIEW	Definition
CREATE TYPE	Definition
SELECT	Manipulation
INSERT	Manipulation
UPDATE	Manipulation
DELETE	Manipulation
COMMIT	Manipulation
ROLLBACK	Manipulation
CREATE TRIGGER	Control, Manipulation
GRANT	Control
REVOKE	Control

SELECT Statement

Introduction

```
SELECT <list-of-column-expression>  
FROM <list-of-tables-and-join-expression>  
WHERE <list-of-logical-expression-for-rows>  
ORDER BY <list-of-sorting-specifications>  
;
```

Additional Keywords

- **AND, OR**
- **DISTINCT:** Remove duplicated rows
- **AS:** Rename a column
- **LIKE:** Inexact string matching
DATE type에는 사용하지 말자
Treat DATE as numeric, not text
- **BETWEEN AND**
- **IS NULL:** Testing a lack of value

Portability Issue

- e.g. Different date functions in
- **Oracle:** to_char, to_string
 - **MySQL:** date_format

JOIN Operator

- Builds a new table by combining rows from two tables that match on a join condition
- Unqualified Columns
 - Qualified Column? 특정 테이블에 속한 column
 - e.g. PK, FK
- EQUI-JOIN (Equality Join)
 - 같은 unqualified column을 join column으로 맞춘다
 - 보통 PK - FK 간의 연결을 주로 볼 수 있음

Using JOIN in SELECT

Cross-Product Style

```
SELECT
  OfferNo,
  CourseNo,
  FacFirstName,
  FacLastName
FROM Offering, Faculty
WHERE OffTerm = 'FALL'
  AND OffYear = 2016
  AND FacRank = 'ASST'
  AND CourseNo LIKE 'IS%'
  AND Faculty.FacNo = Offering.FacNo
;
```

Join Operator Style

```
SELECT
  OfferNo,
  CourseNo,
  FacFirstName,
  FacLastName
FROM Offering
  INNER JOIN Faculty
    ON Faculty.FacNo = Offering.FacNo
WHERE OffTerm = 'FALL'
  AND OffYear = 2016
  AND FacRank = 'ASST'
  AND CourseNo LIKE 'IS%'
```

- N-1 JOIN conditions for N-table JOIN
- 이를 위반하게 되면 심각한 오류가 발생한다.
 - 정확하지 않은 결과
 - 높은 수준의 자원 소비
- Name Qualification 필요

GROUP BY Clause

Row Summaries

```
SELECT StdMajor, AVG(StdGPA) as AvgGPA
FROM Student
WHERE StdClass IN ('JR', 'SR')
GROUP BY StdMajor
HAVING AVG(StdGPA) > 3.1
;
```

Query Clause Evaluation Order

- Row Operations
 - FROM
 - WHERE
- Group Operations (오직 한 번만 일어남)
 - GROUP BY
 - HAVING
- Final Operations for Results
 - ORDER BY
 - SELECT

작은 테이블로 연습하세요