

한봄고등학교 [빅데이터 정보과]

정보처리산업기사 실기 과정평가형

2022. 11. 19(토)

Prepared by DaeKyeong Kim

Ph.D.

(주)엠아이티능력개발원



1

5장 : SQL 활용

5장: SQL 활용





44. SQL-DDL

1. DDL(Data Define Language, 데이터 정의어)

- DDL은 DB 구조, 데이터 형식, 접근 방식 등 DB를 구축하거나 수정할 목적으로 사용하는 언어이다.
- 번역한 결과가 데이터 사전(Data Dictionary)이라는 특별한 파일에 여러 개의 테이블로 저장된다.
- DDL의 3가지 유형

명령어	기능
<small>필기 20.6</small> CREATE	SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의함
<small>필기 20.6</small> ALTER	TABLE에 대한 정의를 변경하는 데 사용함
<small>필기 20.6</small> DROP	SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 삭제함



44. SQL-DDL

2. CREATE SCHEMA

CREATE SCHEMA는 스키마를 정의하는 명령문

- 스키마의 식별을 위해 스키마 이름과 소유권자나 허가권자를 정의함
- 표기 형식

```
CREATE SCHEMA 스키마명 AUTHORIZATION 사용자_id;
```

예제 소유권자의 사용자 ID가 '홍길동'인 스키마 '대학교'를 정의하는 SQL문은 다음과 같다.

3.CREATE DOMAIN

CREATE DOMAIN은 도메인을 정의하는 명령문

- 임의의 속성에서 취할 수 있는 값의 범위가 SQL에서 지원하는 전체 데이터 타입의 값이 아니고 일부분일 때, 사용자는 그 값의 범위를 도메인으로 정의 할 수 있음
- 정의된 도메인명은 일반적인 데이터 타입처럼 사용함
- 표기 형식

```
CREATE DOMAIN 도메인명 [AS] 데이터_타입  
[DEFAULT 기본값]  
[CONSTRAINT 제약조건명 CHECK (범위값)];
```

- 데이터 타입 : SQL에서 지원하는 데이터 타입
- 기본값 : 데이터를 입력하지 않았을 때 자동으로 입력되는 값



44. SQL-DDL

4. CREATE TABLE

CREATE TABLE은 테이블을 정의하는 명령문

- 표기 형식

```
CREATE TABLE 테이블명
    (속성명 데이터_타입 [DEFAULT 기본값] [NOT NULL], ...
    [, PRIMARY KEY(기본키_속성명, ...)]
    [, UNIQUE(대체키_속성명, ...)]
    [, FOREIGN KEY(외래키_속성명, ...)
        REFERENCES 참조테이블(기본키_속성명, ...)
        [ON DELETE 옵션]
        [ON UPDATE 옵션]
    [, CONSTRAINT 제약조건명] [CHECK (조건식)]);
```



44. SQL-DDL

5. CREATE VIEW

CREATE VIEW는 뷰(View)를 정의하는 명령문

- 표기 형식

```
CREATE VIEW 뷰명[(속성명[, 속성명, ...])]  
AS SELECT문;
```

- SELECT문을 서브 쿼리로 사용하여 SELECT문의 결과로서 뷰를 생성함
- 서브 쿼리인 SELECT문에는 UNION이나 ORDER BY절을 사용할 수 없음
- 속성명을 기술하지 않으면 SELECT문의 속성명이 자동으로 사용됨

6. CREATE INDEX

CREATE INDEX는 인덱스를 정의하는 명령문

• 표기 형식

```
CREATE [UNIQUE] INDEX 인덱스명  
ON 테이블명(속성명 [ASC | DESC] [,속성명 [ASC | DESC]])  
[CLUSTER];
```

• UNIQUE

- 사용된 경우 : 중복 값이 없는 속성으로 인덱스를 생성
- 생략된 경우 : 중복 값을 허용하는 속성으로 인덱스를 생성

• 정렬 여부 지정

- ASC : 오름차순 정렬
- DESC : 내림차순 정렬
- 생략된 경우 : 오름차순으로 정렬됨

7. ALTER TABLE

ALTER TABLE은 테이블에 대한 정의를 변경하는 명령문

- 표기 형식

```
ALTER TABLE 테이블명 ADD 속성명 데이터_타입 [DEFAULT '기본값'];
```

```
ALTER TABLE 테이블명 ALTER 속성명 [SET DEFAULT '기본값'];
```

```
ALTER TABLE 테이블명 DROP COLUMN 속성명 [CASCADE];
```

- **ADD** : 새로운 속성(열)을 추가할 때 사용
- **ALTER** : 특정 속성의 Default 값을 변경할 때 사용
- **DROP COLUMN** : 특정 속성을 삭제할 때 사용

```
CREATE TABLE 학생  
  (이름 VARCHAR(15) NOT NULL  
   학번 CHAR(8)  
   학년 CHAR(5))
```



44. SQL-DDL

8. DROP

DROP은 스키마, 도메인, 기본 테이블, 뷰 테이블, 인덱스, 제약 조건 등을 제거하는 명령문

- 표기 형식

```
DROP SCHEMA 스키마명 [CASCADE | RESTRICTED];
```

```
DROP DOMAIN 도메인명 [CASCADE | RESTRICTED];
```

```
DROP TABLE 테이블명 [CASCADE | RESTRICTED];
```

```
DROP VIEW 뷰명 [CASCADE | RESTRICTED];
```

```
DROP INDEX 인덱스명 [CASCADE | RESTRICTED];
```

```
DROP CONSTRAINT 제약조건명;
```

1. DCL(Data Control Language, 데이터 제어어)

- DCL은 데이터의 보안, 무결성, 회복, 병행 제어 등을 정의하는 데 사용하는 언어이다.
- DCL은 데이터베이스 관리자(DBA)가 데이터 관리를 목적으로 사용한다.
- DCL의 종류

명령어	기능
<small>필기 20.8</small> COMMIT	명령에 의해 수행된 결과를 실제 물리적 디스크로 저장하고, 데이터베이스 조작 작업이 정상적으로 완료되었음을 관리자에게 알려줌
<small>필기 20.8</small> ROLLBACK	데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래의 상태로 복구함
<small>필기 20.8</small> GRANT	데이터베이스 사용자에게 사용 권한을 부여함
REVOKE	데이터베이스 사용자의 사용 권한을 취소함

2. GRANT / REVOKE

데이터베이스 관리자가 데이터베이스 사용자에게 권한을 부여하거나 취소하기 위한 명령어

- GRANT : 권한 부여를 위한 명령어
- REVOKE : 권한 취소를 위한 명령어
- 사용자등급 지정 및 해제

- GRANT 사용자등급 TO 사용자_ID_리스트 [IDENTIFIED BY 암호];
- REVOKE 사용자등급 FROM 사용자_ID_리스트;

사용자 등급

- DBA : 데이터베이스 관리자
- RESOURCE : 데이터베이스 및 테이블 생성 가능자
- CONNECT : 단순 사용자

3. COMMIT

트랜잭션이 성공적으로 끝나면 데이터베이스가 새로운 일관성(Consistency) 상태를 가지기 위해 변경된 모든 내용을 데이터베이스에 반영하여야 하는데, 이때 사용하는 명령이 COMMIT 임

- COMMIT 명령을 실행하지 않아도 DML문이 성공적으로 완료되면 자동으로 COMMIT되고, DML이 실패하면 자동으로 ROLLBACK이 되도록 Auto Commit 기능을 설정할 수 있음

4. ROLLBACK

ROLLBACK은 아직 COMMIT되지 않은 변경된 모든 내용들을 취소하고 데이터베이스를 이전 상태로 되돌리는 명령어

- 트랜잭션 전체가 성공적으로 끝나지 못하면 일부 변경된 내용만 데이터베이스에 반영되는 비일관성(Inconsistency)인 상태를 가질 수 있기 때문에 일부 분만 완료된 트랜잭션은 롤백(Rollback)되어야함



5. SAVEPOINT

SAVEPOINT는 트랜잭션 내에 ROLLBACK 할 위치인 저장점을 지정하는 명령어

- 저장점을 지정할 때는 이름을 부여하며, ROLLBACK 시 지정된 저장점까지의 트랜잭션 처리 내용이 취소됨



5. SAVEPOINT

<사원>

사원번호	이름	부서
10	김기획	기획부
20	박인사	인사부
30	최재무	재무부
40	오영업	영업부

예제1 <사원> 테이블에서 '사원번호'가 40인 사원의 정보를 삭제한 후 COMMIT을 수행하시오.

예제2 '사원번호'가 30인 사원의 정보를 삭제하시오.

예제3 SAVEPOINT 'S1'을 설정하고 '사원번호'가 20인 사원의 정보를 삭제하시오.

예제4 SAVEPOINT 'S2'를 설정하고 '사원번호'가 10인 사원의 정보를 삭제하시오.

예제5 SAVEPOINT 'S2'까지 ROLLBACK을 수행하시오.

예제6 SAVEPOINT 'S1'까지 ROLLBACK을 수행하시오.

예제7 SAVEPOINT 없이 ROLLBACK을 수행하시오.

1. DML(Data Manipulation Language, 데이터 조작어)

DML(데이터 조작어)은 데이터베이스 사용자가 응용 프로그램이나 질의어를 통해 저장된 데이터를 실질적으로 관리하는데 사용되는 언어

- DML은 데이터베이스 사용자와 데이터베이스 관리 시스템 간의 인터페이스를 제공함
- DML의 유형

명령문	기능
SELECT	테이블에서 튜플을 검색함
INSERT	테이블에 새로운 튜플을 삽입함
DELETE	테이블에서 튜플을 삭제함
UPDATE	테이블에서 튜플의 내용을 갱신함



2. 삽입문(INSERT INTO)

삽입문은 기본 테이블에 새로운 튜플을 삽입할 때 사용함

• 일반 형식

```
INSERT INTO 테이블명([속성명1, 속성명2,...])  
VALUES (데이터1, 데이터2,... );
```

- 대응하는 속성과 데이터는 개수와 데이터 유형이 일치해야함
- 기본 테이블의 모든 속성을 사용할 때는 속성명을 생략할 수 있음
- SELECT문을 사용하여 다른 테이블의 검색 결과를 삽입할 수 있음



46. SQL-DML

3. 삭제문(DELETE FROM)

삭제문은 기본 테이블에 있는 튜플들 중에서 특정 튜플(행)을 삭제할 때 사용함

• 일반 형식

```
DELETE  
FROM 테이블명  
[WHERE 조건];
```

〈사원〉

이름	부서	생일	주소	기본급
홍길동	기획	04/05/61	망원동	120
임꺽정	인터넷	01/09/69	성산동	80
황진이	편집	07/21/75	연희동	100

- 모든 레코드를 삭제할 때는 WHERE절을 생략함
- 모든 레코드를 삭제하더라도 테이블 구조는 남아 있기 때문에 디스크에서 테이블을 완전히 제거하는 DROP과는 다름



4. 갱신문(UPDATE SET)

갱신문은 기본 테이블에 있는 튜플들 중에서 특정 튜플의 내용을 변경할 때 사용함

- 일반 형식

```
UPDATE 테이블명  
SET 속성명 = 데이터[, 속성명=데이터, ...]  
[WHERE 조건];
```



47. DML-SELECT-1

1. 일반 형식


```
SELECT [PREDICATE] [테이블명.]속성명 [AS 별칭][, [테이블명.]속성명, ...]  
[, 그룹함수(속성명) [AS 별칭]]  
[, Window함수 OVER (PARTITION BY 속성명1, 속성명2, ...  
ORDER BY 속성명3, 속성명4, ...)]  
FROM 테이블명[, 테이블명, ...]  
[WHERE 조건]  
[GROUP BY 속성명, 속성명, ...]  
[HAVING 조건]  
[ORDER BY 속성명 [ASC | DESC]];
```



47. DML-SELECT-1

1. 일반 형식

• SELECT절

 PREDICATE : 불러올 튜플 수를 제한할 명령어를 기술함

▶ ALL : 모든 튜플을 검색할 때 지정하는 것으로, 주로 생략함

▶ DISTINCT : 중복된 튜플이 있으면 그 중 첫 번째 한 개만 검색함

▶ DISTINCTROW : 중복된 튜플을 제거하고 한 개만 검색하지만 선택된 속성의 값이 아닌, 튜플 전체를 대상으로함

- 속성명 : 검색하여 불러올 속성(열) 또는 속성을 이용한 수식을 지정함

▶ 기본 테이블을 구성하는 모든 속성을 지정할 때는 '*'를 기술함

▶ 두 개 이상의 테이블을 대상으로 검색할 때는 '테이블명.속성명'으로 표현함

- AS : 속성 및 연산의 이름을 다른 제목으로 표시하기 위해 사용됨



47. DML-SELECT-1

1. 일반 형식

- FROM절 : 질의에 의해 검색될 데이터들을 포함하는 테이블명을 기술함
- WHERE절 : 검색할 조건을 기술함
- ORDER BY절 : 특정 속성을 기준으로 정렬하여 검색할 때 사용함
 - 속성명 : 정렬의 기준이 되는 속성명을 기술함
 - [ASC | DESC] : 정렬 방식으로서 'ASC'는 오름차순, 'DESC'는 내림차순임. 생략하면 오름차순으로 지정됨



47. DML-SELECT-1

2.조건 연산자

- 비교 연산자 ✓

연산자	=	< >	>	<	> =	< =
의미	같다	같지 않다	크다	작다	크거나 같다	작거나 같다

- 논리 연산자 : NOT, AND, OR
- LIKE 연산자 : 대표 문자를 이용해 지정된 속성의 값이 문자 패턴과 일치하는 튜플을 검색하기 위해 사용된다.

대표 문자	%	_	#
의미	모든 문자를 대표함	문자 하나를 대표함	숫자 하나를 대표함



47. DML-SELECT-1

3. 기본 검색

```
SELECT * FROM 사원;  
SELECT 사원.* FROM 사원;  
SELECT 이름, 부서, 생일, 주소, 기본급 FROM 사원;  
SELECT 사원.이름, 사원.부서, 사원.생일, 사원.주소, 사원.기본급 FROM 사원;
```

4. 조건 지정 검색

```
SELECT *  
FROM 사원  
WHERE 부서 = '기획';
```



47. DML-SELECT-1

5. 정렬 검색

```
SELECT *  
FROM 사원  
ORDER BY 주소 DESC;
```

6. 하위 질의

```
SELECT 이름, 주소  
FROM 사원  
WHERE 이름 = (SELECT 이름 FROM 여가활동 WHERE 취미 = '나이트댄스');
```



47. DML-SELECT-1

7. 복수 테이블 검색

```
SELECT 사원.이름, 사원.부서, 여가활동.취미, 여가활동.경력  
FROM 사원, 여가활동  
WHERE 여가활동.경력 >= 10 AND 사원.이름 = 여가활동.이름;
```



48. DML-SELECT-2

1. 일반 형식

```
SELECT [PREDICATE] [테이블명.]속성명 [AS 별칭][, [테이블명.]속성명, ...]  
[, 그룹함수(속성명) [AS 별칭]]  
[, Window함수 OVER (PARTITION BY 속성명1, 속성명2, ...  
ORDER BY 속성명3, 속성명4, ...)]  
FROM 테이블명[, 테이블명, ...]  
[WHERE 조건]  
[GROUP BY 속성명, 속성명, ...]  
[HAVING 조건]  
[ORDER BY 속성명 [ASC | DESC]];
```



1. 일반 형식

- **그룹함수** : GROUP BY절에 지정된 그룹별로 속성의 값을 집계할 함수를 기술함
- **WINDOW 함수** : GROUP BY절을 이용하지 않고 속성의 값을 집계할 함수를 기술함
 - PARTITION BY : WINDOW 함수가 적용될 범위로 사용할 속성을 지정함
 - ORDER BY : PARTITION 안에서 정렬 기준으로 사용할 속성을 지정함
- **GROUP BY절** : 특정 속성을 기준으로 그룹화하여 검색할 때 사용함. 일반적으로 GROUP BY절은 그룹 함수와 함께 사용됨
- **HAVING절** : GROUP BY와 함께 사용되며, 그룹에 대한 조건을 지정함



48. DML-SELECT-2

2. 그룹 함수

GROUP BY절에 지정된 그룹별로 속성의 값을 집계할 때 사용됨

- COUNT(속성명) : 그룹별 튜플 수를 구하는 함수
- SUM(속성명) : 그룹별 합계를 구하는 함수
- AVG(속성명) : 그룹별 평균을 구하는 함수
- MAX(속성명) : 그룹별 최대값을 구하는 함수
- MIN(속성명) : 그룹별 최소값을 구하는 함수
- STDDEV(속성명) : 그룹별 표준편차를 구하는 함수
- VARIANCE(속성명) : 그룹별 분산을 구하는 함수



48. DML-SELECT-2

2. 그룹 함수

- ROLLUP(속성명, 속성명, ...)



- 인수로 주어진 속성을 대상으로 그룹별 소계를 구하는 함수
- 속성의 개수가 n 개이면, $n+1$ 레벨까지, 하위 레벨에서 상위 레벨 순으로 데이터가 집계됨

- CUBE(속성명, 속성명, ...)

- ROLLUP과 유사한 형태이나 CUBE는 인수로 주어진 속성을 대상으로 모든 조합의 그룹별 소계를 구함
- 속성의 개수가 n 개이면, n^2 레벨까지, 상위 레벨에서 하위 레벨 순으로 데이터가 집계됨



3. Window 함수

- GROUP BY절을 이용하지 않고 함수의 인수로 지정한 속성을 범위로 하여 속성의 값을 집계함
- 함수의 인수로 지정한 속성이 대상 레코드의 범위가 되는데, 이를 윈도우(WINDOW)라고 부름
- WINDOW 함수
 - ROW_NUMBER() : 윈도우별로 각 레코드에 대한 일련 번호를 반환함
 - RANK() : 윈도우별로 순위를 반환하며, 공동 순위를 반영함
 - DENSE_RANK() : 윈도우별로 순위를 반환하며, 공동 순위를 무시하고 순위를 부여함

예제 <상여금> 테이블에서 '상여내역'별로 '상여금'에 대한 일련 번호를 구하시오. (단 순서는 내림차순이며 속성명은 'NO'로 할 것)

```
SELECT 상여내역, 상여금,  
       ROW_NUMBER() OVER (PARTITION BY 상여내역 ORDER BY 상여금 DESC) AS NO  
FROM 상여금;
```



48. DML-SELECT-2

4. 그룹 지정 검색

```
SELECT 부서, COUNT(*) AS 사원수  
FROM 상여금  
GROUP BY 부서;
```



48. DML-SELECT-2

5. 집합 연산자를 이용한 통합 질의

집합 연산자를 사용하여 2개 이상의 테이블의 데이터를 하나로 통합한다.

표기 형식

```
SELECT 속성명1, 속성명2, ...  
FROM 테이블명  
UNION | UNION ALL | INTERSECT | EXCEPT  
SELECT 속성명1, 속성명2, ...  
FROM 테이블명  
[ORDER BY 속성명 [ASC | DESC]];
```

- 두 개의 SELECT문에 기술한 속성들은 개수와 데이터 유형이 서로 동일해야 한다.



48. DML-SELECT-2

5. 집합 연산자를 이용한 통합 질의

- 집합 연산자의 종류(통합 질의의 종류)

집합 연산자	설명	집합 종류
UNION	<ul style="list-style-type: none">• 두 SELECT문의 조회 결과를 통합하여 모두 출력함• 중복된 행은 한 번만 출력함	합집합 ✓
UNION ALL	<ul style="list-style-type: none">• 두 SELECT문의 조회 결과를 통합하여 모두 출력함• 중복된 행도 그대로 출력함	합집합
INTERSECT	두 SELECT문의 조회 결과 중 공통된 행만 출력함	교집합
EXCEPT	첫 번째 SELECT문의 조회 결과에서 두 번째 SELECT문의 조회 결과를 제외한 행을 출력함	차집합



49. DML-JOIN

1. JOIN

- JOIN은 2개의 릴레이션에서 연관된 튜플들을 결합하여, 하나의 새로운 릴레이션을 반환한다.
- JOIN은 일반적으로 FROM절에 기술하지만, 릴레이션이 사용되는 곳 어디에나 사용할 수 있다.
- JOIN은 크게 INNER JOIN과 OUTER JOIN으로 구분된다.



2. INNER JOIN

INNER JOIN은 일반적으로 EQUI JOIN과 NON-EQUI JOIN으로 구분된다.

- 조건이 없는 INNER JOIN을 수행하면 CROSS JOIN*과 동일한 결과를 얻을 수 있다.
- EQUI JOIN
 - EQUI JOIN은 JOIN 대상 테이블에서 공통 속성을 기준으로 '='(equal) 비교에 의해 같은 값을 가지는 행을 연결하여 결과를 생성하는 JOIN 방법이다.
 - EQUI JOIN에서 JOIN 조건이 '='일 때 동일한 속성이 두 번 나타나게 되는데, 이 중 중복된 속성을 제거하여 같은 속성을 한 번만 표기하는 방법을 NATURAL JOIN이라고 한다.
 - EQUI JOIN에서 연결 고리가 되는 공통 속성을 JOIN 속성이라고 한다.



49. DML-JOIN

2. INNER JOIN

- WHERE절을 이용한 EQUI JOIN의 표기 형식

```
SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...  
FROM 테이블명1, 테이블명2, ...  
WHERE 테이블명1.속성명 = 테이블명2.속성명;
```

- NATURAL JOIN절을 이용한 EQUI JOIN의 표기 형식

```
SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...  
FROM 테이블명1 NATURAL JOIN 테이블명2;
```

- JOIN ~ USING절을 이용한 EQUI JOIN의 표기 형식

```
SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...  
FROM 테이블명1 JOIN 테이블명2 USING(속성명);
```



49. DML-JOIN

2. INNER JOIN

- NON-EQUI JOIN

- NON-EQUI JOIN은 JOIN 조건에 '=' 조건이 아닌 나머지 비교 연산자, 즉 >, <, <>, >=, <= 연산자를 사용하는 JOIN 방법이다.

- 표기 형식

```
SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...  
FROM 테이블명1, 테이블명2, ...  
WHERE (NON-EQUI JOIN 조건);
```




49. DML-JOIN

3. OUTER JOIN

OUTER JOIN은 릴레이션에서 JOIN 조건에 만족하지 않는 튜플도 결과로 출력하기 위한 JOIN 방법으로, LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTER JOIN이 있다.

- LEFT OUTER JOIN : INNER JOIN의 결과를 구한 후, 우측 항 릴레이션의 어떤 튜플과도 맞지 않는 좌측 항의 릴레이션에 있는 튜플들에 NULL 값을 붙여서 INNER JOIN의 결과에 추가한다.

- 표기 형식

- SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...
FROM 테이블명1 LEFT OUTER JOIN 테이블명2
ON 테이블명1.속성명 = 테이블명2.속성명;
- SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...
FROM 테이블명1, 테이블명2
WHERE 테이블명1.속성명 = 테이블명2.속성명(+);



49. DML-JOIN

3. OUTER JOIN

- RIGHT OUTER JOIN : INNER JOIN의 결과를 구한 후, 좌측 항 릴레이션의 어떤 튜플과도 맞지 않는 우측 항의 릴레이션에 있는 튜플들에 NULL 값을 붙여서 INNER JOIN의 결과에 추가한다
 - 표기 형식

- SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...
FROM 테이블명1 RIGHT OUTER JOIN 테이블명2
ON 테이블명1.속성명 = 테이블명2.속성명;
- SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...
FROM 테이블명1, 테이블명2
WHERE 테이블명1.속성명(+) = 테이블명2.속성명;



49. DML-JOIN

3. OUTER JOIN

- FULL OUTER JOIN
 - LEFT OUTER JOIN과 RIGHT OUTER JOIN을 합쳐 놓은 것이다.
 - INNER JOIN의 결과를 구한 후, 좌측 항의 릴레이션의 튜플들에 대해 우측 항의 릴레이션의 어떤 튜플과도 맞지 않는 튜플들에 NULL 값을 붙여서 INNER JOIN의 결과에 추가한다.
그리고 유사하게 우측 항의 릴레이션의 튜플들에 대해 좌측 항의 릴레이션의 어떤 튜플과도 맞지 않는 튜플들에 NULL 값을 붙여서 INNER JOIN의 결과에 추가한다.
 - 표기 형식

```
SELECT [테이블명1.]속성명, [테이블명2.]속성명, ...  
FROM 테이블명1 FULL OUTER JOIN 테이블명2  
ON 테이블명1.속성명 = 테이블명2.속성명;
```

1. 데이터 사전

데이터 사전은 데이터베이스에 저장되어 있는 모든 데이터 개체들에 대한 정보의 집합체.

- 내용 변경 권한은 데이터베이스 관리 시스템(DBMS)이 가지며, 사용자는 단순 조회만 가능함.
- 메타 데이터로 구성되어 있으며, 시스템 카탈로그(SystemCatalog)라고도 한다.
- 데이터 사전은 테이블 형태로 저장되어 있으므로 SELECT문을 이용하여 조회할 수 있음.
- 대표적인 데이터 사전 테이블

테이블명	메타 데이터	DBMS
TABLES	테이블에 대한 정보	My-SQL, Oracle
VIEWS	뷰에 대한 정보	My-SQL, Oracle
COLUMNS	속성에 대한 정보	My-SQL
TAB_COLUMNS		Oracle
TRIGGERS	트리거에 대한 정보	My-SQL, Oracle

1. 데이터 사전

※ 메타 데이터(Meta Data)

- 메타 데이터는 데이터 관리를 위한 데이터, 즉 데이터를 위한 데이터를 말함.
- 메타 데이터의 유형
 - 데이터베이스 객체 정보 : 테이블(Table), 인덱스(Index), 뷰(View) 등의 구조 및 통계 정보
 - 사용자 정보 : 아이디, 비밀번호, 접근 권한 등
 - 테이블의 무결성 제약 조건 정보 : 기본키, 외래키, NULL 값 허용 여부 등
 - 함수, 프로시저, 트리거 등에 대한 정보

2. My-SQL의 데이터 사전 조회

My-SQL에서 데이터 사전 테이블을 조회하려면 먼저 테이블들이 모여있는 INFORMATION_SCHEMA 스키마로 이동해야 하며, SHOW 명령으로 스키마 내의 테이블 목록을 확인할 수 있음.

```
USE INFORMATION_SCHEMA;  
SHOW TABLES;  
SELECT * FROM [테이블];
```

- **USE INFORMATION_SCHEMA;** : 'INFORMATION_SCHEMA'로 이동함
- **SHOW TABLES;**
 - 'INFORMATION_SCHEMA'에 저장된 데이터 사전 테이블들의 목록을 조회함.
 - <TABLES>, <VIEWS>, <COLUMNS> 등 메타 데이터를 저장하고 있는 데이터 사전 테이블이 조회됨.
- **SELECT * FROM [테이블];** : [테이블]에 있는 모든 속성을 조회함
 - [테이블] : SHOW TABLES ; 를 통해 확인한 테이블 중 조회할 메타 데이터가 저장된 테이블을 입력함

2. My-SQL의 데이터 사전 조회

예제 My-SQL에서 현재 위치한 스키마가 'sys'일 때 데이터베이스에 존재하는 모든 테이블에 대한 데이터 사전 정보를 조회하는 SQL문을 작성하시오.

```
USE INFORMATION_SCHEMA;  
SELECT * FROM TABLES;
```

[결과]

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE	TABLE_ROWS	AVG_ROW_LENGTH	...
def	sys	강의	BASE TABLE	43	8192	...
def	sys	학생	BASE TABLE	24	2048	...
def	sys	채용	BASE TABLE	11	8192	...
def	sys	직원	BASE TABLE	21	8192	...
def	sys	emp_tbl	VIEW	9	5461	...
def	sys	shop	BASE TABLE	7	3276	...

3. Oracle의 데이터 사전 조회

Oracle에서는 데이터 사전 테이블 앞에 '영역'을 지정하여 정보를 조회함.

```
SELECT * FROM [영역]_[테이블];
```

- SELECT * : 모든 속성을 표시함
- FROM [영역]_[테이블] : <[영역]_[테이블]>에 해당하는 정보를 조회함
 - [영역] : DBA, ALL, USER 중 필요한 접근 권한을 입력함
 - ▶ DBA : 데이터베이스의 모든 객체에 대해 조회 가능
 - ▶ ALL : DBA의 권한 없이 접근할 수 있는 모든 객체에 대해 조회 가능
 - ▶ USER : 현재 자신의 계정이 소유한 객체에 대해 조회 가능
 - [테이블] : <TABLES>, <INDEXES>, <VIEWS>등 조회할 메타 데이터가 저장된 테이블을 입력함

3. Oracle의 데이터 사전 조회

예제 Oracle에서 일반 사용자 계정으로 접근할 수 있는 뷰에 대한 데이터 사전 정보를 조회하는 SQL문을 작성하시오.

```
SELECT * FROM ALL_VIEWS;
```

[결과]

OWNER	VIEW_NAME	TEXT_LENGTH	TEXT	TYPE_TEXT	SUPERVIEW_NAME	...
SYS	V_POLICY	359	[뷰를 정의하는 SELECT문]	(null)	(null)	...
SYS	V_ICY	729	[뷰를 정의하는 SELECT문]	(null)	(null)	...
SYS	V_EMP	330	[뷰를 정의하는 SELECT문]	(null)	(null)	...
SYS	V_S_LIST	275	[뷰를 정의하는 SELECT문]	(null)	(null)	...
SYS	DASH_V	163	[뷰를 정의하는 SELECT문]	(null)	(null)	...
SYS	DASH_T	344	[뷰를 정의하는 SELECT문]	(null)	(null)	...

Unit A

참고자료



문헌

1. <http://www.ncs.go.kr>
2. 2022 시나공 정보처리산업기사 실기, 김정준, 강윤석, 길벗알앤디 저 | 길벗 | 2022년 03월 14일

실습 자료

1. <https://github.com/looker2zip/info.git>

감사합니다.

❖ Mobile: 010-9591-1401

❖ E-mail: onlooker2zip@naver.com / dkkim@lklab.org

