

● SQL활용

1-1. DDL활용 (데이터 정의어) Data Definition Language

○ DDL 개요

DDL 은 데이터를 정의하는 언어 / 데이터를 담는 그릇 / DBMS에서는 오브젝트라고 함

○DDL 대상

- **스키마** : DBMS특성과 구현환경을 감안한 데이터 구조 / 직관적으로 하나의 데이터 베이스로 이해 가능
- **도메인** : 속성과 데이터 타입과 크기, 제약조건 등을 지정한 정보 / 속성이 가질 수 있는 값의 범위로 이해가능
- **테이블** : 데이터저장공간
- **뷰**: 하나이상의 물리테이블에서 유도되는 가상의 논리테이블
- **인덱스**: 검색을 빠르게 하기 위한 데이터 구조

○ DDL조작방법

- **생성(Create)** : 데이터베이스 오브젝트 생성
- **변경(Alter)**: 데이터베이스 오브젝트 변경
- **삭제(Drop)**: 데이터베이스 오브젝트 삭제
- **삭제(Truncate)**:데이터베이스 **오브젝트 내용** 삭제

○DDL활용

데이터베이스를 구축하기 위해 스키마, 테이블, 도메인, 인덱스, 뷰와 같은 오브젝트에 대한 DDL적용이 필요함.

1. 테이블 생성

```
create table 테이블 이름(  
열이름 데이터 타입 [default값] [not null]  
{,열이름 데이터 타입[default값] [not null]}*  
[primary key(열 리스트),]  
[{foreign key(열 리스트) references 테이블 이름[(열이름)]  
[on delete 옵션]  
[on update옵션]],}*  
[check(조건식 | unique(열이름))];
```

다른 테이블 정보를 이용한 테이블 생성

```
create table 테이블 이름 as select 문;
```

2. 테이블 변경:alter를 이용, 테이블 구조를 변경하는 문법

열추가:ALTER TABLE 테이블 이름 ADD 열이름 데이터타입[DEFAULT값]

열 데이터 타입 변경 :ALTER TABLE 테이블이름 MODIFY 열이름 데이터타입[DEFAULT값]

열 삭제 :ALTER TABLE 테이블이름 DROP열이름

3. 테이블삭제, 절단, 이름변경

테이블 삭제: DROP TABLE 테이블이름

테이블 내용 삭제: TRUNCATE TABLE 테이블이름

테이블 이름 변경 : RENAME TABLE 이전 테이블 이름 TO 새로운 테이블 이름

ALTER TABLE 이전테이블 이름 RENAME 새로운테이블 이름

○제약조건 유형

PRIMARY KEY : 테이블의 기본키를 정의 / 기본으로 NOT NULL, UNIQUE 제약이 포함

FOREIGN KEY : 외래키를 정의함. 참조 대상을 테이블이름(열이름)으로 명시해야함. 참조 무결성 위배 상황 발생 시 처리방법으로 옵션지정 가능

UNIQUE:테이블 내에서 열은 유일한 값을 가져야 함./테이블 내에서 동일한 값을 가져서는 안되는 항목에 지정함

NOT NULL:테이블내에서 관련 열의 값은 NULL일 수 없음 / 필수 입력 항목에 대해 제약 조건으로 설정함

CHECK:개발자가 정의하는 제약조건 / 상황에 따라 다양한 조건 설정가능

1-2.DML활용 : Data Manipulation Language

○DML의미 : 데이터를 조작하는 명령어/ 데이터 관점에서 생명주기 제어

○DML유형

-데이터생성 **INSERT**

-데이터조회 **SELECT**

-데이터변경 **UPDATE**

-데이터 삭제 **DELETE**

1-3. DCL활용 : Data Control Language

○ **DCL**유형 : 데이터베이스에서 데이터 이외의 오브젝트에 대해 조작

-사용자권한 :접근통제

사용자를 등록하고, 사용자에게 특정 데이터베이스를 사용할 수 있는 **권리를 부여**하는 작업

- 트랜잭션: 안전한 거래보장

동시에 다수의 작업을 독립적으로 안전하게 처리하기 위한 상호 작용 단위

○트랜잭션 제어를 위한 명령어 TCL(Transaction Control Language)

TCL과 DCL은 대상이 달라 서로 별개의 개념으로 분류할 수 있으나, 제어기능의 공통점으로 DCL의 일부로 분류하기도 한다.

○DCL 명령어

GRANT 데이터베이스 사용자 **권한 부여**

REVOKE 데이터베이스 사용자 **권한 회수**

TCL명령어

COMMIT 트랜잭션 확정 / **ROLLBACK** 트랜잭션 취소 / **CHECKPOINT** 복귀지점 설정

○ DCL활용

1. 사용자 권한 부여

시스템권한 : GRANT 권한1, 권한2 TO 사용자계정

객체권한: GRANT 권한1, 권한2 ON 객체명 TO 사용자계정

2. 사용자 권한 회수

시스템권한: REVOKE권한1, 권한2 FROM 사용자계정

객체권한: REVOKE 권한1, 권한2 ON 객체명 FROM 사용자계정

○ DCL이론적 배경인 접근통제

개념

보안정책에 따라 접근객체(시스템 자원, 통신 자원 등)에 대한 접근주체(사용자, 프로세스 등)의 접근권한확인 및 이를 기반으로 한 접근 제어를 통해 자원에 대한 비인가된 사용을 방지하는 정보 보호 기능

○접근 통제 정책에 따른 유형

- **임의 접근 통제 : DAC**

시스템 객체에 대한 접근을 사용자 개인 또는 그룹의 식별자를 기반으로 제한하는 방법

여기서는 임의적이라는 말은 어떤 종류의 접근 권한을 갖는 사용자는 다른 사용자에게 자신의 판단에 의해서 권한을 줄수 있다는 것, 주체와 객체의 신분 및 임의적 접근 통제 규칙에 기초하여 객체에 대한 주체의 접근을 통제하는 기능

(**통제 권한이 주체에 있음**)(주체가 임의적으로 접근 통제 권한을 배분하며 제어할 수 있음)

- **강제 접근 통제 : MAC**

정보시스템 내에서 어떤 주체가 특정 객체에 접근하려 할 때 양쪽의 보안 레이블에 기초하여 높은 보안 수준을 요구하는 정보(객체)가 낮은 보안 수준의 주체에게 노출되지 않도록 접근을 제한하는 통제방법

(통제 권한이 제3자에게 있음) (주체는 접근 통제 권한과 무관함)

○접근통제와 DCL 관계

강제 접근 통제의 경우, 제 3자의 종류에 따라 보다 세분화된 정책이 존재

접근통제 정책의 두가지 가운데 데이터베이스관리시스템

○ TCL활용방법

○트랜잭션 개념 : 일처리 단위

- 트랜잭션은 논리적 연산단위

-한개 이상의 데이터베이스 조작

-하나 이상의 SQL문장이 포함

-트랜잭션은 거래이다

-이때 거래 결과가 모두 반영되거나 모두 취소되어야 한다

-데이터베이스에서 트랜잭션은 특별한 엄격한 거래를 의미

-분할할 수 없는 최소단위

○트랜잭션 제어

트랜잭션의 결괏값 수용하거나 취소하는 것 의미

TCL명령어

COMMIT 거래내역을 확정함

ROLLBACK거래내역을 취소함

CHECKPOINT저장점 설정 / ROLLBACK할 위치를 지정함

○ 트랜잭션의 특성

원자성 : 트랜잭션 안에 정의된 연산은 모두 실행되거나 모두 실행되지 않아야 함

일관성: 트랜잭션 실행전과 후 동일하게 오류가 없어야 함

고립성: 트랜잭션 실행 중 다른 트랜잭션에 영향을 받지 않아야 함

지속성:트랜잭션 결과는 항상 보존됨

○메모리 관점에서 트랜잭션 조작확인

트랜잭션의 구현 원리

DBMS의 모든 정보는 하드디스크에 저장

DBMS에서 이루어지는 모든 조작 또는 연산은 메모리에서 이루어짐

하드디스크에 있는 정보를 메모리로 옮겨서 연산을 수행

적당한 시점에 메모리 정보를 하드디스크로 옮김

COMMIT 메모리의 내용을 하드디스크에 저장

ROLLBACK메모리의 내용을 하드에 저장않고 버린다

CHECKPOINT ROLLBACK범위 설정을 위해 메모리 상에 경계설정

1-4데이터 사전 검색

○데이터사전

데이터베이스의 데이터를 제외한 모든정보

데이터사전의 내용을 변경하는 권한은 시스템이 가지며 사용자에게는 읽기전용테이블 형태로 제공

단순 조회만 가능

데이터를 제외한 모든정보라는 것은 데이터의 데이터를 의미

따라서 데이터 사전은 메타데이터로 구성되어 있다.

○데이터 사전 내용

-사용자정보(아이디, 패스워드 및 권한 등)

-데이터베이스 객체 정보(테이블, 뷰, 인덱스 등)

-무결성 제약 정보

-함수, 프로시저 및 트리거 등

○데이터 사전 용도

TO 사용자 : 단순 조회 대상

데이터베이스 엔진을 이루는 컴파일러, 옵티마이저 등과 같은 구성요소에 데이터 사전은 작업을 수행하는 데 필요한 참조 정보일 뿐 아니라 작업의 대상

○데이터 사전 검색

1. 오라클

뷰로 데이터 사전에 접근할 수 있다.

데이터 사전과 관련된 뷰는 세가지영역이 있으며, 이때 오브젝트에 접근할 수 있는 사용자 권한에 따라 구분

DBA_>ALL_>USER_

영역 지시자 뒤에 오브젝트명을 붙이는 형태로 뷰의 이름이 결정된다

테이블, 뷰, 인덱스 와 같은 것 의미

○DBA : 데이터베이스의 모든 객체 조회 가능 (시스템 접근 권한 의미)

○ALL : 자신의 계정으로 접근 가능한 객체와 타 계정의 접근 권한을 가진 모든 객체 조회 가능

○USER : 현재 자신의 계정이 소유한 객체 조회 가능

○MYSQL에서 데이터 사전 검색

테이블 형태로 구성 / 해당 테이블의 위치와 이름을 정확히알고 있어야 한다.

information_schema라는 데이터베이스 안에 존재

○use Information_schema : --이동 ○show tables: --테이블 목록 보기

○SQL개요

DDL, DML, DCL 각 용도에 따라 SQL명령문 사용법 확인

용어: 구조적 질의 언어 -구조적 개념에 대한 이해 필요

용도: 데이터베이스 조작 언어 - 데이터베이스에 대한 이해 필요

형식: ISO/ANSI 표준으로 정의됨 - 표준의 변화에 대한 이해 필요

○ 용어 관점에서의 SQL

구조적 질의어 = 구조체 =비절차적 질의어 : 절차적으로 세세하게 묘사하는 것이 아니다.

○표준관점에서의 SQL

SQL은 데이터베이스를 조작하는 언어의 표준이다.

SQL-86 : ANSI에 의한 최초의 표준화

SQL-89 : 폭발적 전성기 시대의 표준, 벤더별 문법, 용어 차이로 상호 호환성에 문제 있음

SQL:1999 객체지향지원 , SQL3최초 상용화 버전은 오라클8i

SQL:2003 XML관련 특징 도입

현재 출시 제품, 최소한의 데이터베이스 표준

○용도관점에서의 SQL

SQL은 데이터베이스를 조작하는 언어다.

데이터베이스 : 연관된 데이터의 모음 / 데이터를 일정한 형태로 저장해 놓은 것

저장데이터: 컴퓨터를 통해 접근 가능한 저장 매체에 저장된 데이터

통합데이터: 중복이 최소화된 데이터

공유데이터: 여러 응용프로그램이 공동으로 사용하는 데이터

운영데이터: 조직의 목적을 위해 존재가치가 확실하고 반드시 필요한 데이터

○데이터베이스 관리 시스템: 조직의 목적을 위해 존재하는 운영 데이터를 통합 저장하여 공동으로 사용 가능하도록 관리하는 시스템 -DBMS 인스턴스 구조: 대용량 공유 메모리와 프로세스로 구성

SQL은 DBMS를 이용하여 DB를 조작하기 위한 언어 형태를 가진 수단.

○인덱스 : 데이터를 빠르게 찾을 수 있는 수단. 테이블에 대한 조회 속도를 높여주는 자료구조. 자동으로 생성X

○PK컬럼: PK를 생성할 때 자동으로 인덱스가 생성

인덱스 생성: CREATE[UNIQUE] INDEX<index_name> ON <table_name>(<column(s)>);

인덱스 삭제: DROP INDEX<index_name>;

인덱스 변경: ALTER [UNIQUE] INDEX<index_name> ON<table name> (<coloumn(s)>);

○인덱스 설계과정

접근경로 수집/ 분포도 조사를 통한 후보 컬럼 선정/ 접근 경로 결정/ 컬럼 조합 및 순서 결정/ 적용시험

○뷰 : 논리테이블로서 사용자에게는 테이블과 동일. 단순한 질의어 사용할 수 있기 때문에 사용.

장점 : 논리적 독립성 제공 / 사용자 데이터 관리용이 / 데이터 보안 용이

단점 : 뷰 자체 인덱스 불가 / 뷰 정의 변경 불가/ 데이터 변경 제약 존재

뷰 생성: CREATE VIEW <뷰이름>(컬럼목록) AS <뷰를 통해 보여줄 데이터 조회용 쿼리문>

뷰삭제 및 변경: 뷰 정의 자체를 변경하는 것은 불가능. 삭제와 재생성으로 변경 가능 DROP VIEW <View Name>;

뷰 내용 변경 : 뷰를 통해 접근 가능한 데이터에 대한 변경 가능하지만, 모든 경우가 아니라 일부 제약이 존재하고 insert는 불가.

다중테이블

○조인: 교집합 결과를 가지는 결합. 두 테이블의 공통값을 이용하여 컬럼을 조합. 세 개 이상의 테이블에 대한 결합은 두 개의 테이블을 우선 결합하고 그 결과와 나머지 한 개의 테이블을 다시 결합.

○유형(대분류)

논리적 조인: 사용자의 SQL문에 표현되는 테이블 결합 방식

물리적 조인: 데이터베이스의 옵티마이저에 의해 내부적으로 발생하는 테이블 결합방식

○유형(세분류)

내부조인(Inner Join): 두테이블에 공통으로 존재하는 컬럼을 이용(공통 컬럼 기반)(동등조인 / 자연 조인 / 교차 조인)

동등조인(EQUI JOIN) : 공통 존재 컬럼의 값이 같은 경우를 추출

자연조인(Natural Join): 두테이블의 모든 컬럼을 비교하여같은 컬럼명을 가진 모든 컬럼 값이 같은 경우를 추출

교차조인(Cross Join): 조인 조건이 없는 모든 데이터의 조합을 추출

외부조인(Outer Join): 특정 테이블의 모든 데이터를 기준으로 다른 테이블의 정보를 추출(다른 테이블에 값이 없어도 출력 됨)

(왼쪽 외부 조인 / 오른쪽 외부 조인 / 완전 외부 조인)

왼쪽 외부 조인(Left Outer Join) :왼쪽 테이블의 모든 데이터와 오른쪽 테이블의 동일 데이터 추출

오른쪽 외부 조인(Right Outer Join): 오른쪽 테이블의 모든 데이터와 왼쪽 테이블의 동일 데이터를 추출

완전 외부 조인(Full Outer Join): 양쪽의 모든 데이터 추출

○서브쿼리: SQL문 안에 포함된 또다른 SQL문을 의미 / 알려지지 않은 기준을 위한 검색을 위해 사용

○ 집합연산: 테이블을 집합 개념으로 보고, 두 테이블 연산에 집합 연산자를 사용하는 방식.

집합 연산자는 여러 질의 결과를 연결하여 하나로 결합하는 방식을 사용.

집합 연산자는 2개 이상의 질의 결과를 하나의 결과로 만들어준다.

UNION :여러SQL문의 결과에 대한 합집합(중복 행 제거O), UNION ALL: 여러 SQL문의 결과에 대한 합집합(중복행제거X)

INTERSECTION:여러 SQL문의 결과에 대한 교집합(중복행제거O), EXCEPT(MINUS): 앞의 SQL문의 결과와 뒤의 SQL문의 결과 사이이 차집합(중복행제거, 일부 제품의 경우 MINUS사용)