

DB_SQL_1일차_04.02

데이터 베이스

체계적인 데이터 모음

데이터

저장이나 처리에 효율적인 형태로 변환된 정보

기존 데이터 저장 방식

파일 (File) & 스프레드 시트 (Spreadsheet)

파일을 이용한 데이터 관리

장점 : 어디에서나 쉽게 사용 가능

단점 : 데이터를 구조적으로 관리하기 어려움

스프레드 시트를 이용한 데이터 관리

장점 : 테이블의 열과 행을 사용해 데이터를 구조적으로 관리 가능

단점 : 크기(일반적으로 100만 행까지만 저장가능)

보안(단순 파일, 링크 소유 여부에 따른 단순한 접근 권한 기능 제공)

정확성(데이터 변경시 테이블 모든 위치에서 해당 값을 업데이트 해야 함)

데이터베이스의 역할

데이터를 (구조적)저장하고 조작(CRUD)

관계형 데이터베이스

데이터 간에 관계가 있는 데이터 항목들의 모음

테이블, 행, 열의 정보를 구조화하는 방식

서로 관련된 데이터 포인터를 저장하고 이에 대한 액세스를 제공

관계

여러 테이블 간의 논리적 연결

관계로 인해 두 테이블을 사용하여 데이터를 다양한 형식으로 조회 가능

관계형 데이터베이스 관련 키워드

Table : 데이터를 기록하는 곳

Field : 각 필드에는 고유한 데이터 형식(타입)이 지정됨

Record : 각 레코드에는 구체적인 데이터 값이 저장됨

Database : 테이블의 집합

Primary Key (기본키, PK) : 각 레코드의 고유한 값, 레코드의 식별자로 활용

Foreign Key (외래키, FK) : 테이블의 필드 중 다른 테이블의 레코드를 식별할 수 있는 키, 다른 테이블의 기본 키를 참조, 각 레코드에서 서로 다른 테이블 간의 관계를 만드는데 사용

DBMS (Database Management System)

데이터베이스를 관리하는 소프트웨어 프로그램

데이터 저장 및 관리를 용이하게 하는 시스템

데이터베이스와 사용자 간 인터페이스 역할

사용자가 데이터 구성, 업데이트, 모니터링, 백업, 복구 등을 할 수 있도록 도움

RDBMS (Relational Database Management System)

관계형 데이터베이스를 관리하는 소프트웨어 프로그램

종류 : SQLite, MySQL, PostgreSQL, Oracle Database

SQLite : 경량의 오픈 소스 데이터베이스 관리 시스템 (PC, 모바일 기기에 내장되어 간단하고 효율적인 데이터 저장 및 관리를 제공)

데이터베이스 정리

Table은 데이터가 기록되는 곳

Table 행에는 고유하게 식별 가능한 기본 키 속성이 있으며, 외래 키를 사용하여 각 행에서 서로 다른 테이블 간의 관계를 만들 수 있음

데이터는 기본 키 & 외래 키를 통해 결합(join)될 수 있는 여러 테이블에 걸쳐 구조화 됨

SQL (Structure Query Language)

데이터베이스에 정보를 저장하고 처리하기 위한 프로그래밍 언어

-> 테이블의 형태로 구조화된 관계형 데이터베이스에게 요청을 질의

SQL Syntax

SQL 키워드는 대문자 작성 권장 (대소문자 구분하지는 않음)

각 SQL Statements 끝에는 세미콜론(;)이 필요 (명령어의 마침표)

SQL Statements

SQL을 구성하는 가장 기본적인 코드 블록 종류 : DDL, DQL, DML, DCL

DDL - 데이터 정의

데이터의 기본 구조 및 형식 변경 (CREATE, DROP, ALTER)

DQL - 데이터 검색

데이터 검색 (SELECT)

DML - 데이터 조작

데이터 추가, 수정, 삭제 (INSERT, UPDATE, DELETE)

DCL - 데이터 제어

데이터 및 작업에 대한 사용자 권한 제어 (COMMIT, ROLLBACK, GRANT, REVOKE)

Query

데이터베이스로부터 정보를 요청하는 것

SQL로 작성하는 코드를 쿼리문(SQL문)이라고 함

SQL 표준

ANSI(미국 국립 표준 협회)와 ISO(국제 표준화 기구)에 의해 표준이 채택 (모든 RDBMS에서 SQL 표준 지원)

데이터 검색의 종류

SELECT

테이블에서 데이터를 조회 및 반환

SELECT 키워드 이후 데이터를 선택하려는 필드를 하나 이상 지정

-> SELECT *을 사용하면 모든 필드 데이터를 조회

-> 필드 지정 후 AS '이름' 사용하면 '이름'으로 출력됨

FROM 키워드 이후 데이터를 선택하려는 테이블의 이름을 지정

ORDER BY

조회 결과의 레코드를 정렬하는 구문

FROM 뒤에 작성

하나 이상의 컬럼을 기준으로 결과를 오름차순(ASC), 내림차순(DESC)으로 정렬 (기본값은 ASC)

NULL 값이 있는 경우 오름차순 정렬 시 가장 먼저 출력, 내림차순은 마지막에 출력

SELECT statement 실행순서

1. 테이블에서 FROM
2. 조회하여 SELECT
3. 정렬 ORDER BY