

### 3. 데이터베이스 시스템

---

#### 데이터베이스 시스템의 정의

- 데이터베이스 시스템은 데이터베이스에 데이터를 저장하고, 저장된 데이터를 관리하여 조직에 필요한 정보를 생성해주는 시스템이다
  - 따라서 다양한 목적의 정보 처리 시스템을 구축하는데 필요한 핵심 요소가 된다
  - 데이터베이스 시스템은 데이터베이스와 데이터베이스 관리 시스템을 이용해 조직에 피룡한 정보를 제공해주는 전체 시스템이다
    - 데이터베이스 시스템 = 데이터베이스 + 데이터베이스 관리 시스템 + 사용자 + 데이터베이스 접근에 사용하는 데이터 언어 + 데이터 처리 연산을 담당하는 컴퓨터
- 

#### 데이터베이스의 구조

##### 스키마

- 스키마는 데이터베이스에 저장되는 데이터 구조와 제약조건을 정의한 것

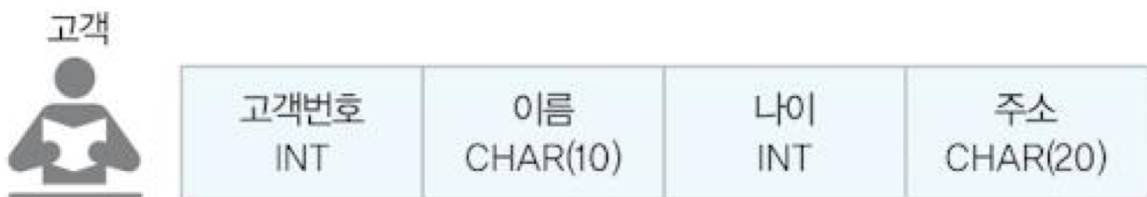


그림 3-2 스키마의 예

- 고객과 관련된 데이터인 고객번호, 이름, 나이, 주소를 저장한다고 가정하자
  - 고객번호는 정수로, 이름은 최대 10자의 문자열로, 나이는 정수로, 주소는 최대 20자의 문자열만 허용하기로 했다면 이 모든 정해진 내용이 스키마다
- 정의된 스키마에 따라 데이터베이스에 실제로 저장된 값이 인스턴스다
- 보통 스키마는 한번 정의되면 자주 변경되지 않지만, 인스턴스는 계속 변하는 특성이 있다
  - 객체지향의 클래스, 인스턴스 관계

#### 3단계 데이터베이스 구조

- 데이터베이스는 복잡한 내부 구조와 동작 방식을 가지고 있어 이를 사용자가 모두 이해하고 사용하기엔 무리가 있다
- 따라서 데이터베이스의 복잡한 내부 구조를 감추고 일반 사용자가 데이터베이스를 쉽게 이해하고 이용할 수 있도록 3단계 데이터베이스 구조를 제안하였다
- 3단계 데이터베이스 구조의 개념
  - 3단계 데이터베이스 구조는 하나의 데이터베이스를 세 단계로 나누어 이해한다
  - 개별 사용자 관점에서 바라보는 외부 단계
  - 조직 전체의 관점에서 바라보는 개념 단계
  - 물리적인 저장 장치의 관점에서 바라보는 내부 단계
  - 데이터베이스 하나를 3단계로 나누고 각 단계별로 다른 추상화를 제공하면 데이터베이스를 효과적으로 관리할 수 있다
  - 일반적으로 내부 단계에서 외부 단계로 갈수록 추상화 레벨이 높아진다
  - 외부 단계
    - 외부 단계에서는 개별 사용자 관점에서 데이터베이스를 이해하고 표현한다
    - 하나의 데이터베이스를 조직 내의 사용자들이 함께 사용하지만 각 사용자가 데이터베이스 전체에 관심이 있는 것은 아니다
    - 사용자마다 업무 내용과 사용 목적이 달라 필요한 데이터 내용이 다를 수 있다
    - 외부 단계에서는 개별 사용자가 데이터베이스를 어떻게 보는가를 표현하므로 사용자마다 생각하는 데이터베이스의 구조가 다르다
    - 이처럼 외부 단계에서 사용자에게 필요한 데이터베이스를 정의한 것을 외부 스키마라 한다
    - 외부 스키마는 각 사용자가 생각하는 데이터베이스의 모습을 표현한 논리적인 구조로, 사용자마다 다르다
    - 하나의 데이터베이스에는 외부 스키마가 여러개 존재할 수 있고, 외부 스키마 하나를 사용 목적이 같은 사용자끼리 공유할 수 있다
    - 외부 스키마는 전체 데이터베이스 중 사용자가 관심을 가지는 일부분으로 볼 수 있어 서브 스키마라고도 한다
  - 개념 단계
    - 개념 단계에서는 데이터베이스를 이용하는 사용자들의 관점을 통합하여, 데이터베이스를 조직 전체의 관점에서 이해하고 표현한다
    - DBMS이나 데이터베이스 관리자는 데이터베이스의 일부분이 아닌 전체 데이터베이스에 관심을 둔다
    - 개념 단계에서는 데이터베이스 관리 시스템이나 관리자의 관점에서 모든 사용자에게 필요한 데이터를 통합하여 전체 데이터베이스의 논리적 구조를 정의한다
    - 이를 개념 스키마라 한다
    - 개념 스키마는 조직 전체의 관점에서 생각하는 데이터베이스의 모습이며, 모든 개별 사용자가 생각하는 데이터베이스의 모습을 하나로 합친 형태다
    - 개념 스키마는 전체 데이터베이스에 어떤 데이터가 저장되는지, 데이터들 간에는 어떤 관계가 존재하고 어떤 제약 조건이 있는지에 대한 정의 뿐만 아니라, 데이터에 대한 보안 정책이나 접근 권한에 대한 정의도 포함한다
    - 하나의 데이터베이스에는 개념 스키마가 하나만 존재하고, 각 사용자는 개념 스키마의 일부분을 사용한다

- 즉, 외부 스키마는 개념 스키마를 기초로하여 사용자의 이용 목적에 맞게 만들어진다
- 일반적으로 스키마라고 하면 개념 스키마를 의미한다
- 내부 단계
- 내부 단계에서는 데이터베이스를 디스크나 테이프 같은 저장 장치의 관점에서 이해하고 표현한다
- 즉, 내부 단계에서는 전체 데이터베이스가 저장 장치에 실제로 저장되는 방법을 정의하며 이를 내부 스키마라 한다
- 데이터베이스는 저장 장치에 파일 형태로 저장되는데 내부 스키마는 파일에 데이터를 저장하는 레코드의 구조, 레코드를 구성하는 필드 크기, 인덱스를 이용한 레코드 접근 경로등을 정의한다
- 내부 스키마는 데이터베이스의 개념 스키마에 대한 물리적인 저장 구조를 표현하므로 하나의 데이터베이스에 하나만 존재한다

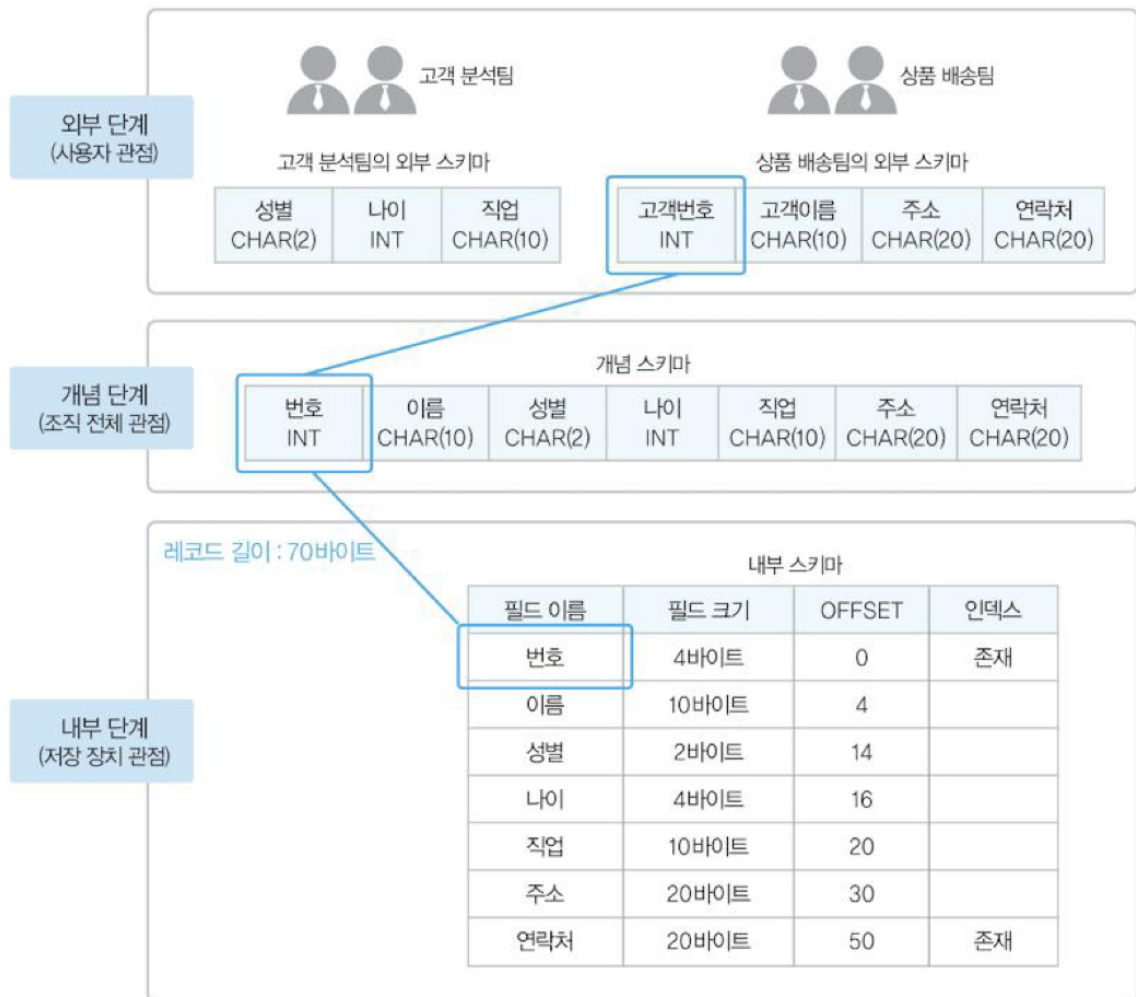


그림 3-5 3단계 데이터베이스 구조의 예

#### • 데이터 독립성

- 하나의 데이터베이스에는 세 가지 유형의 스키마가 존재하지만, 각각의 스키마는 데이터베이스를 바라보는 관점이 다를 뿐 모두 같은 데이터베이스를 표현한다
- 실제 데이터는 물리적 저장 장치에 저장된 데이터베이스에만 존재하므로 사용자가 자신의 외부 스키마를 통해 원하는 데이터를 얻으려면 내부 스키마에 따라 저장된 데이터베이스에 접근해야 한다
- 그러므로 세가지 스키마 사이에는 유기적인 대응 관계가 성립해야 한다
- 위의 예시에서 상품 배송팀의 외부 스키마에 있는 고객 번호 데이터는 개념 스키마에 있는 번호 데이터

에 대응하고, 개념 스키마에 있는 번호 데이터는 내부 스키마에 있는 번호 필드에 대응한다는 연결 관계가 미리 정의되어 있어야 한다

- 스키마 사아의 대응 관계를 사상 또는 매핑이라 한다
- 외부 스키마와 개념 스키마는 외부/개념 사상(논리적 데이터 독립성)에 의해 대응되고
- 개념 스키마와 내부 스키마는 개념/내부 사상(물리적 데이터 독립성)에 의해 대응된다
- 데이터베이스를 3단계 구조로 나누고, 단계별로 스키마를 유지하며 스키마 사이의 대응 관계를 정의하는 궁극적인 목적은 데이터 독립성을 실현하기 위해서다
- 데이터 독립성은 데이터베이스 관리 시스템의 중요한 장점이자 데이터베이스 관리 시스템이 필요한 이유이기도 하다
- 데이터 독립성은 하위 스키마를 변경하더라도 상위 스키마가 영향을 받지 않는 특성이다
- 3단계 데이터베이스 구조에는 논리적 데이터 독립성과 물리적 데이터 독립성이 존재한다

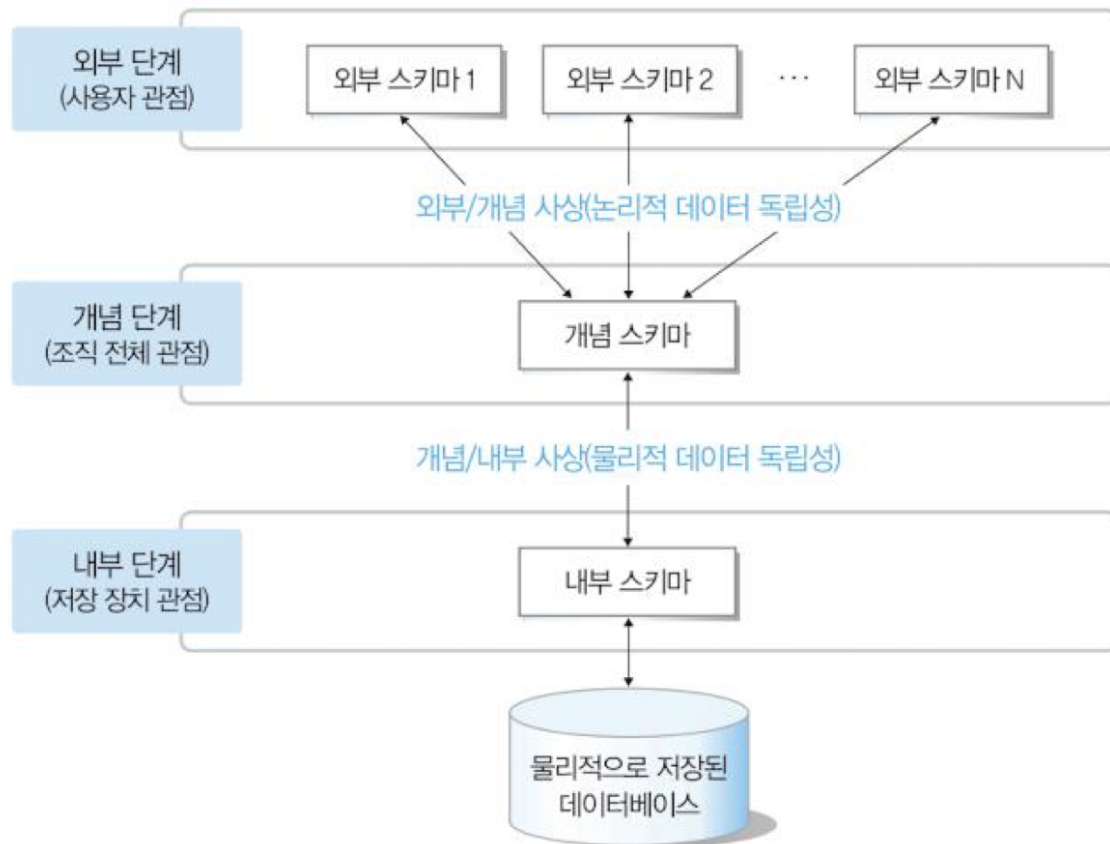


그림 3-6 3단계 데이터베이스 구조에서 스키마 간의 사상

- 논리적 데이터 독립성
  - 논리적 데이터 독립성은 개념 스키마가 변경되더라도 외부 스키마가 영향을 받지 않는 것이다
  - 그래서 전체 데이터베이스의 논리적인 구조가 변경되어도 관련된 외부/개념 사상 정보만 적절히 수정해주면 직접 관련이 없는 사용자를 위한 외부 스키마는 변경할 필요가 없다
  - 외부/개념 사상은 외부 스키마와 개념 스키마의 대응 관계를 정의한 것으로, 응용 인터페이스라고도 한다
  - 즉, 외부 스키마의 사용자는 전체 데이터베이스의 논리적 구조가 변경되었다는 사실을 알 필요가 없다

- 물리적 데이터 독립성
  - 물리적 데이터 독립성은 내부 스키마가 변경되더라도 개념 스키마가 영향을 받지 않는 것이다
    - 따라서 결과적으로 외부 스키마도 영향을 받지 않는다
  - 물리적 데이터 독립성이 실현되면 데이터베이스의 저장 구조가 변경되어도 관련된 개념/내부 사상 정보만 적절히 수정해주면 직접적으로 관련이 없는 데이터베이스의 논리적 구조는 영향을 받지 않는다
  - 개념/내부 사상은 개념 스키마와 내부 스키마의 대응 관계를 정의한 것으로, 저장 인터페이스라고도 한다
- 데이터 사전
  - 데이터베이스는 조직 운영에 필요한 실제 데이터를 저장하는데, 저장된 데이터를 올바르게 관리하고 이용하려면 필요한 부가 정보도 저장해야 한다
    - 대표적인 부가 정보가 스키마와 사상 정보다
  - 데이터 독립성을 실현하면서 데이터베이스를 다양한 관점에서 이해하기 위해 정의되는 세 가지 스키마에 대한 정보와 스키마간의 사상 정보도 어딘가에 저장되어 있어야 필요할 때 사용할 수 있다
    - 데이터베이스에 저장되는 데이터에 관한 정보를 저장하는 곳을 데이터 사전 또는 시스템 카탈로그라고 한다
  - 데이터 사전은 일반 사전처럼 데이터베이스에 저장되어 있는 데이터를 정확하고 효율적으로 이용하기 위해 참고해야 되는 스키마, 사상 정보, 다양한 제약 조건등을 저장하고 있다
  - 데이터베이스에 저장되는 데이터에 관한 정보이므로 데이터 II 에 대한 데이터를 의미해 메타 데이터라고도 한다
  - 데이터 사전도 데이터를 저장하는 데이터베이스의 일종이기 때문에 시스템 데이터베이스라고도 한다
    - 이와 구별하기 위해 사용자가 실제로 이용하는 데이터가 저장되는 일반 데이터베이스를 사용자 데이터베이스라 부르기도 한다
  - 데이터 사전은 데이터베이스 관리 시스템이 스스로 생성하고 유지하는 것으로, 데이터베이스 관리 시스템이 주로 접근하지만 일반 사용자도 접근할 수 있다
    - 단, 데이터베이스 관리 시스템이 데이터 사전에 내용을 새로 추가하거나 수정할 수 있는 반면, 사용자는 저장 내용을 검색만 할 수 있다
  - 데이터 사전에 있는 데이터에 실제로 접근하는데 필요한 위치 정보는 데이터 디렉터리라는 곳에서 관리한다
  - 데이터 사전과 데이터 디렉터리는 둘 다 시스템을 위한 데이터베이스라는 공통점이 있지만, 데이터 사전은 사용자가 접근할 수 있고 데이터 디렉터리는 시스템만 접근할 수 있다는 차이가 있다

---

## 데이터베이스 사용자

- 데이터베이스 사용자는 데이터베이스를 이용하기 위해 접근하는 모든 사람을 의미한다
  - 이를 이용 목적에 따라 데이터베이스 관리자, 최종 사용자, 응용 프로그래머로 나눌 수 있다

- 데이터베이스 관리자
  - 데이터베이스 관리자(DBA : DataBase Administrator)는 데이터베이스 시스템을 운영, 관리한다
  - 데이터베이스를 직접 활용하기 보다는 조직 내의 사용자를 위해 데이터베이스를 설계 및 구축하고, 제대로 서비스할 수 있도록 데이터베이스를 제어한다
  - 그래서 데이터베이스 관리자는 데이터 언어 중 주로 데이터 정의어와 데이터 제어어를 이용해 데이터베이스에 접근한다
  - 데이터베이스 관리자의 주요 업무
    - 데이터베이스 구성 요소 선정
      - 사용자의 요구 사항을 분석하여 데이터베이스를 구성할 데이터를 결정한다
    - 데이터베이스 스키마 정의
      - 선정된 데이터베이스의 구성 요소를 토대로 데이터베이스 스키마를 설계하고, 데이터 정의를 이용해 설계한 스키마를 데이터베이스 관리 시스템에 설명한다
    - 물리적 저장 구조와 접근 방법 결정
      - 데이터베이스를 물리적으로 저장하기 위한 레코드 구조를 설계한다
      - 레코드들 간의 저장 순서와 레코드에 빠르게 접근하기 위해 인덱스를 만들 기준 필드 등도 결정한다
    - 무결성 유지를 위한 제약조건 정의
      - 현실 세계의 실제 데이터와 일치하는, 즉 결함이 없는 데이터만 데이터베이스에 저장할 수 있도록 필요한 규칙을 정의한다
      - 그리고 데이터베이스를 이 규칙에 따라 제어하여 데이터의 정확성과 유효성을 유지한다
    - 보안 및 접근 관한 정책 설정
      - 허가되지 않는 사용자가 데이터베이스에 불법적으로 접근하는 것을 방지하고, 허가된 사용자에게 적절한 권한을 부여하는 보안 관련 정책을 결정한다
    - 백업 및 회복 기법 정의
      - 시스템 장애에 대비하여 데이터베이스를 백업하거나, 손상된 데이터베이스를 일관된 상태로 복구하는 방법을 정의한다
    - 시스템 데이터베이스 관리
      - 데이터 사전 같은 시스템 데이터베이스를 관리한다
    - 시스템 성능 감시 및 성능 분석
      - 시스템 성능을 저해하는 병목 현상 등이 발생하지 않는지 확인하고, 시스템 자원의 활용도 분석등을 통해 시스템의 성능을 감시한다
    - 데이터베이스 재구성
      - 사용자의 요구사항이나 사용 형태가 달라지면, 변화된 내용에 맞게 데이터베이스를 재구성하여 사용자의 만족도를 높여야 한다
      - 시스템의 전체 성능을 향상시켜야 하거나 시스템 장비가 교체된 경우에도 데이터베이스를 재구성 해야한다
- 최종 사용자

- 데이터를 조작(삽입, 삭제, 수정, 검색)하기 위해 데이터베이스에 접근하는 사람들을 일반 사용자 또는 최종 사용자라 한다
- 최종 사용자는 컴퓨터 시스템이나 데이터베이스에 관한 전문 지식을 꼭 갖출 필요는 없다
- 그리고 데이터 정의를 사용할 수도 있지만 주로 데이터 조작어를 사용한다
- 최종 사용자는 캐주얼 사용자와 초보 사용자로 구분할 수 있다
- 캐주얼 사용자는 데이터베이스에 대한 이론적 지식이 있으며, 주로 데이터 조작어를 이용해 원하는 데이터와 데이터에 대한 처리를 데이터베이스 관리 시스템에 직접 설명한다
- 초보 사용자는 데이터베이스를 초보 수준으로 이용할 수 있어, 데이터 조작어로 자신의 요구를 직접 표현하기보다는 메뉴난 GUI 형태의 응용 프로그램을 통해 데이터베이스를 사용한다
- 응용 프로그래머
  - 응용 프로그래머는 자바등과 같은 프로그래밍 언어로 응용 프로그램을 작성할 때 데이터베이스에 접근하여 데이터 조작어를 삽입하는 사용자다
  - 데이터 정의를 삽입할 수도 있지만 주로 데이터 조작어를 삽입한다
  - 최종 사용자는 응용 프로그래머가 작성한 응용 프로그램을 이용해 데이터베이스에 접근할 수 있다

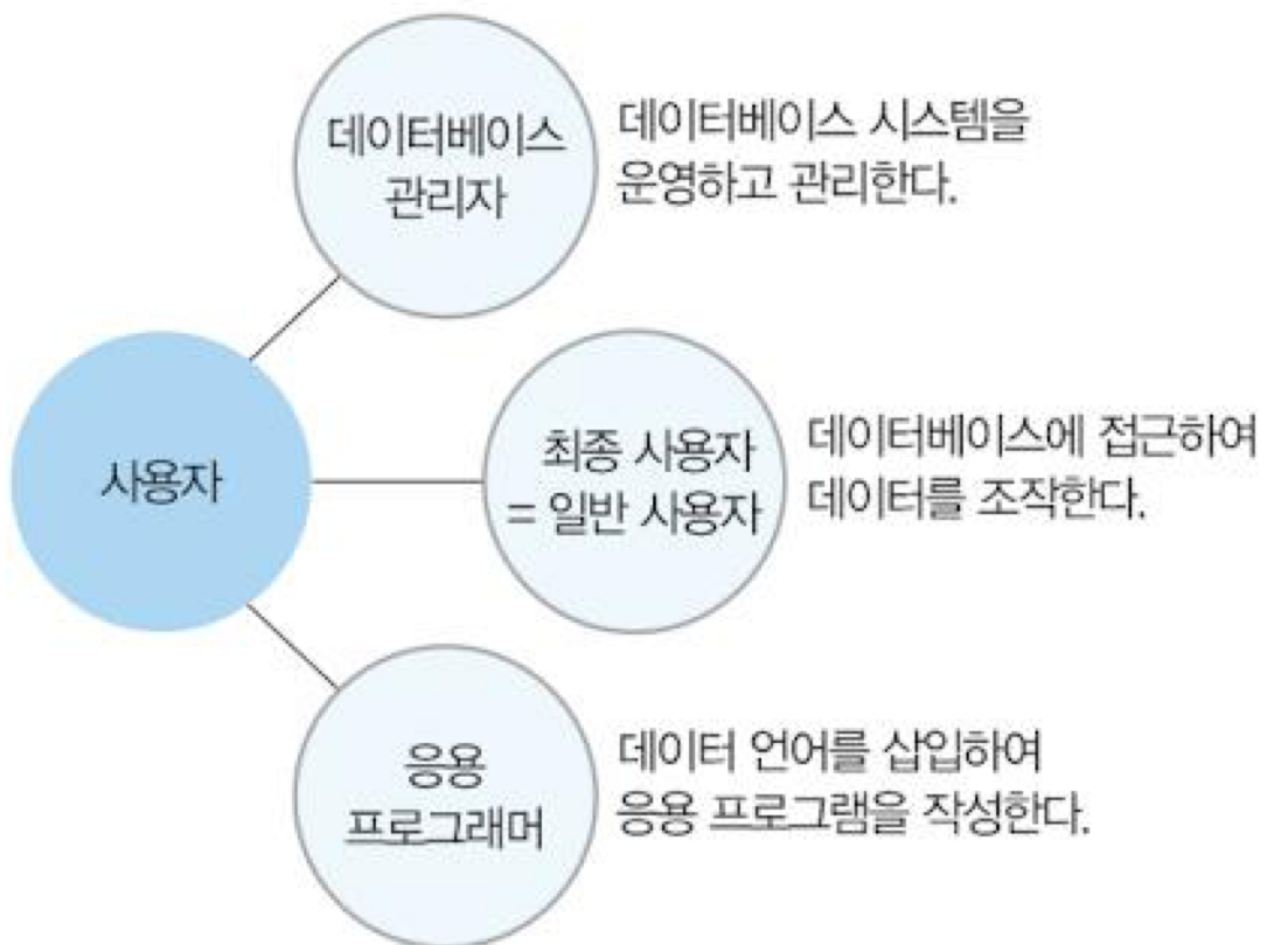


그림 3-7 데이터베이스 사용자

# 데이터 언어

- 데이터 언어는 사용자를 대신해 데이터베이스를 구축하고 활용 및 관리하는 데이터베이스 관리 시스템에 부탁할 때 사용하는 언어다
  - 즉, 데이터 언어는 사용자가 데이터베이스를 구축하고 이에 접근하기 위해 데이터베이스 관리 시스템과 통신하는 수단이다
- 데이터 언어는 데이터베이스 관리 시스템의 정의, 조작, 제어 기능을 이용하기 위한 수단이기 때문에 사용 목적에 따라 데이터 정의어, 데이터 조작어, 데이터 제어어로 나뉜다
- 데이터 정의어
  - 데이터 정의어(DDL : Data Definition Language)는 새로운 데이터베이스를 구축하기 위해 스키마를 정의하거나 기존 스키마의 정의를 삭제 또는 수정하기 위해 사용하는 데이터 언어다
  - 즉, 새로 만들려는 데이터베이스의 스키마를 설명하거나 이미 정의된 스키마의 구조나 제약조건 등을 변경 또는 삭제하고 싶어 이를 데이터베이스 관리 시스템에 알릴 때 사용한다
  - 데이터 정의어로 정의된 스키마는 데이터 사전에 저장되고, 삭제나 수정이 발생하면 이 내용도 데이터 사전에 반영된다
- 데이터 조작어
  - 데이터 조작어(DML : Data Manipulation Language)는 사용자가 데이터의 삽입, 삭제, 수정, 검색 등의 처리를 데이터베이스 관리 시스템에 요구하기 위해 사용하는 데이터 언어다
  - 데이터 정의를 이용해 스키마를 정의하면 스키마에 따라 조직에 필요한 실제 데이터 값(인스턴스)이 저장되는데, 사용자가 실제 데이터 값을 활용하기 위해 사용하는 것이 데이터 조작어다
  - 데이터 조작어는 설명 방식에 따라 절차적 데이터 조작어와 비절차적 데이터 조작어로 나뉜다
  - 절차적 데이터 조작어
    - 절차적 데이터 조작어(procedural DML)는 사용자가 어떤 데이터를 원하고 해당 데이터를 얻으려면 어떻게 처리해야 하는지를 구체적으로 설명한다
  - 비절차적 데이터 조작어
    - 비절차적 데이터 조작어(nonprocedural DML)는 사용자가 어떤 데이터룰 원하는지만 설명한다
      - 즉, 해당 데이터를 얻으려면 어떻게 처리해야 하는지는 데이터베이스 관리 시스템에 맡긴다
    - 비절차적 데이터 조작어는 사용자가 어떤 데이터를 원하는지만 데이터베이스 고나리 시스템에 선언하는 방식이기 때문에 선언적 언어(declarative language)라고도 한다
- 데이터 제어어
  - 데이터 제어어(DCL : Data Control Language)는 데이터베이스에 저장된 데이터를 여러 사용자가 무결성과 일관성을 유지하며 문제없이 공유할 수 있도록, 내부적으로 필요한 규칙이나 기법을 정의하는데 사용하는 데이터 언어다
  - 사용자는 데이터베이스를 올바르게 관리하기 위해 필요한 규칙과 기법을 데이터 제어어를 이용해 데이터베이스 관리 시스템에 설명한다



- 그러면 데이터베이스 관리 시스템이 이 규칙과 기법에 따라 데이터베이스를 제어하고 보호한다
- 데이터 제어어를 이용해 규칙이나 기법을 정의하는 이유는 다음과 같은 특성을 보장하기 위해서다
- 무결성(integrity) : 데이터베이스에 정확하고 유용한 데이터만 유지
- 보안(security) : 허가받지 않는 사용자가 데이터에 접근하는 것을 차단하거나, 허가된 사용자가 접근 권한이 있는 데이터에만 접근할 수 있게한다
- 회복(recovery) : 장애가 발생해도 데이터의 일관성을 유지
- 동시성(concurrency) : 여러 사용자가 같은 데이터에 동시에 접근하여 처리할 수 있게한다

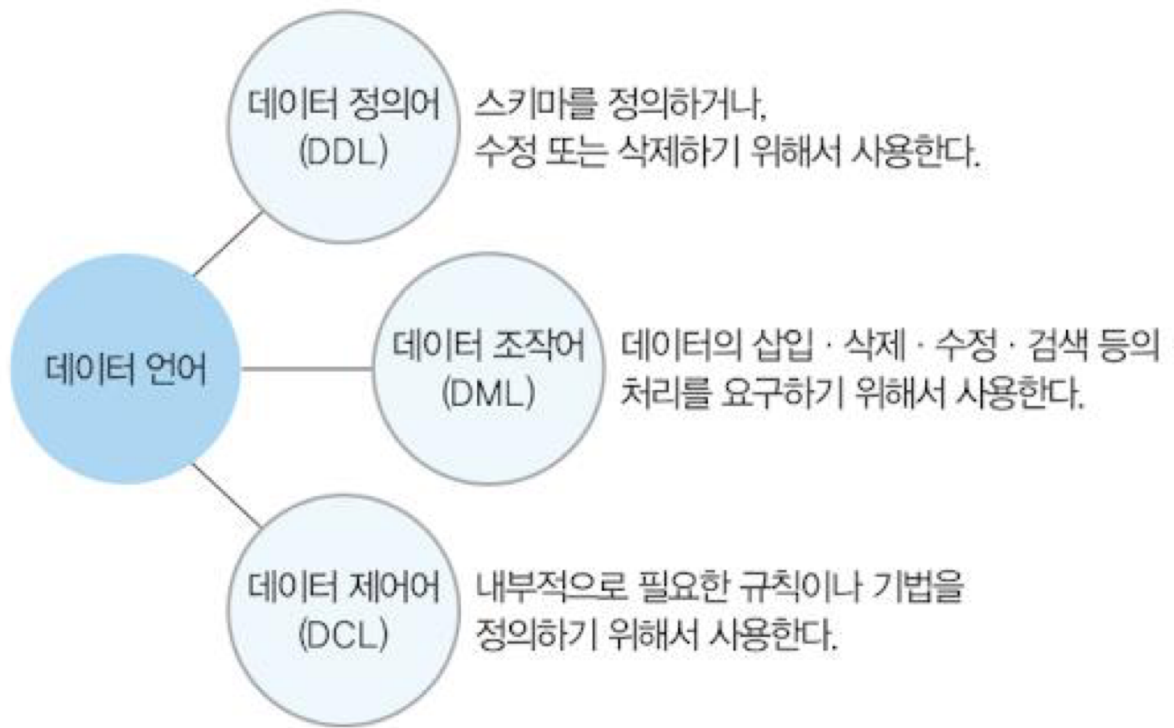


그림 3-9 데이터 언어의 종류와 용도

## 데이터베이스 관리 시스템의 구성

- 데이터베이스를 관리하고 사용자의 데이터 처리 요구를 수행하는 데이터베이스 관리 시스템은 데이터베이스 시스템의 주요 구성 요소다
  - 사용자와 데이터베이스 사이에 위치하며, 기능에 따라 크게 질의 처리기와 저장 데이터 관리자로 구분

할 수 있다

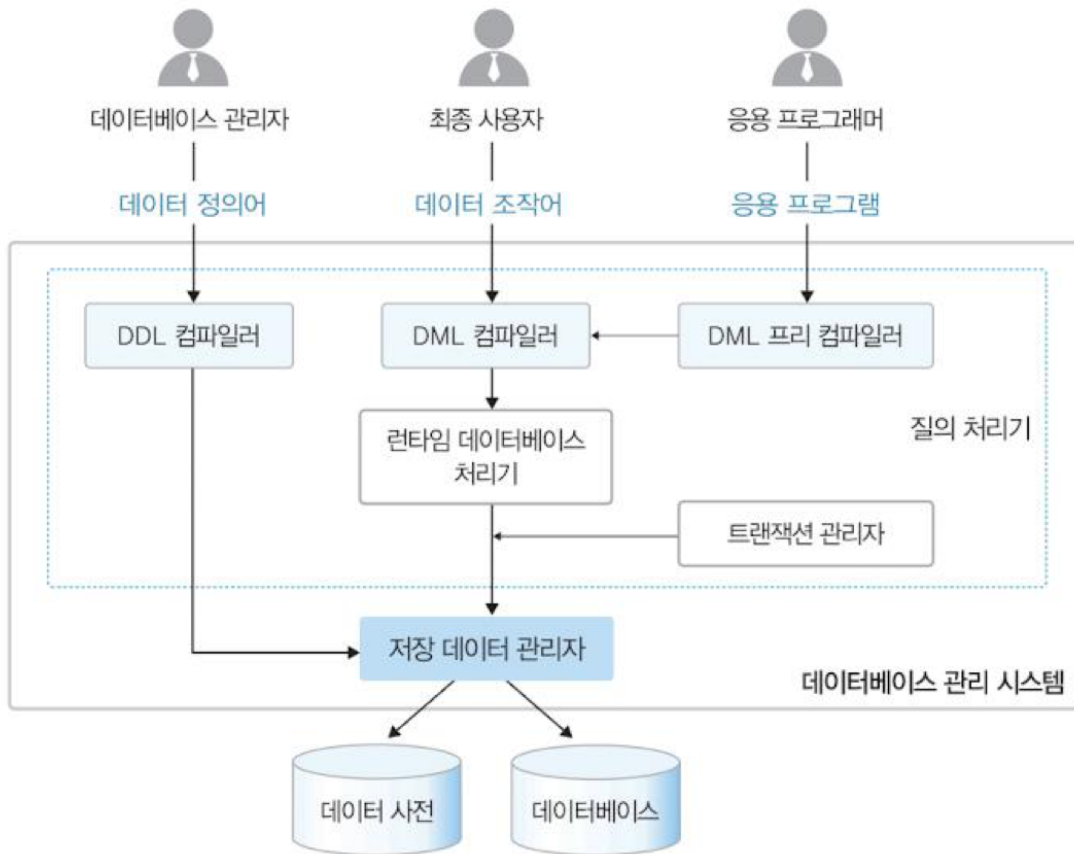


그림 3-11 데이터베이스 관리 시스템의 구성

- 질의 처리기
  - 질의 처리기(query processor)는 사용자의 데이터 처리 요구를 해석하여 처리하는 역할을 담당하고, 다음의 주요 구성 요소들을 포함한다
    - DDL 컴파일러
      - 데이터 정의어로 작성된 스키마의 정의를 해석한다
      - 저장 데이터 관리자의 도움을 받아 새로운 데이터베이스를 구축하고, 스키마의 정의를 데이터 사전에 저장한다
      - 데이터 정의어로 작성된 기존 스키마의 삭제나 수정 요청도 처리하여, 변경된 내용을 데이터 사전에 적용한다
    - DML 프리 컴파일러
      - 응용 프로그램에 삽입된 데이터 조작어를 추출하여 DML 컴파일러에 전달한다
      - 단, 데이터 조작어와 관련 없는 나머지 코드들은 해당 언어의 컴파일러에 보내진다
    - DML 컴파일러
      - 데이터 조작어로 작성된 데이터의 처리(삽입, 삭제, 수정, 검색)요구를 분석하여 런타임 데이터베이스 처리기가 이해할 수 있도록 해석한다
    - 런타임 데이터베이스 처리기

- 저장 데이터 관리자를 통해 데이터베이스에 접근하여, DML컴파일러로부터 전달받은 데이터 처리 요구를 데이터베이스에서 실제로 실행한다
- 트랜잭션 관리자
  - 데이터베이스에 접근하는 과정에서 사용자의 접근 권한이 유효한지를 검사하고, 데이터베이스 무결성일 유지하기 위한 제약조건 위반 여부를 확인한다
  - 회복이나 병행 수행과 관련되 작업도 담당한다
- 저장 데이터 관리자
  - 저장 데이터 관리자(stored data manager)는 디스크에 저장된 데이터베이스와 데이터 사전을 관리하고, 여기에 실제로 접근하는 역할을 담당한다
  - 그런데 디스크에 저장된 데이터에 접근하는 것은 운영체제의 기본 기능이므로 저장 데이터 관리자는 운영체제의 도움을 받아 데이터베이스에 대한 접근을 수행한다