

# 데이터 모델링

## 1. 데이터모델의 개념

- 현실세계를 추상화하여 사람, 장소, 사물, 범주에 대한 데이터 구조와 이들 간의 관계를 시각적으로 표현한 것을 의미함
- 데이터모델링은 데이터 모델을 개발하기 위한 과정으로 데이터를 정의하고 구조화함으로써 효과적인 정보 시스템을 개발할 수 있음

## 2. 데이터모델의 종류

- 계층적 데이터 모델
- 네트워크 데이터 모델
- 개체-관계형 데이터 모델 - 개체-관계형 모델은 현실 세계의 개체(Entity)와 그들 간의 관계(Relationship)를 표현하는데 중점을 둔 데이터 모델. 이 모델에서 개체는 현실 세계에서 식별할 수 있는 무언가를 나타내며, 속성(Attribute)을 가질 수 있음. 관계는 개체들 간의 연결을 나타내며, 일대일, 일대다, 다대다 등 다양한 유형이 있음. 개체-관계형 모델은 개체와 관계에 대한 엔터티-관계 다이어그램(ER 다이어그램)을 사용하여 시각적으로 표현됨. 이 모델은 주로 데이터베이스 설계와 시스템 분석에서 사용.
- **관계형 데이터 모델** - 관계형 모델은 테이블 형태로 데이터를 저장하고 관리하는데 중점을 둔 데이터 모델. 데이터는 테이블로 구성되며, 각 테이블은 행(Row)과 열(Column)로 구성. 각 테이블은 관계(Relationship)를 통해 다른 테이블과 연결될 수 있으며 테이블 간의 관계는 외래 키(Foreign Key)를 사용하여 정의됨. 관계형 모델은 SQL(Structured Query Language)을 사용하여 데이터를 쿼리하고 조작함. 관계형 데이터베이스 시스템(RDBMS)은 관계형 모델을 기반으로 구축되며, 대부분의 기업 및 조직에서 널리 사용됨.

## 3. 데이터 모델링 프로세스

### 1) **요구사항 분석**

### 2) 개념적 데이터 모델링

- 엔티티 도출 - 학생, 과목, 교수와 같이 데이터베이스내에서 관리되는 정보의 단위로 고유한 식별자로 식별되는 개념

- 속성 도출 - 엔티티 집합에 포함되는 최소의 데이터 단위. 하나의 엔티티는 각각의 속성에 대해 단일 속성값을 가져야 함
- 식별자 도출 - 각 엔티티를 구분할 수 있는 속성 또는 속성의 집합. 엔티티는 반드시 식별자를 가져야 함
- 관계 도출 - 엔티티 간의 관계를 식별. 이 단계에서는 각 엔티티가 다른 엔티티와 어떻게 연결되는지를 결정하며 관계는 일대일, 일대다, 다대다 등의 형태를 가질 수 있음.

### 3) 논리적 데이터 모델링

- 개념적 모델을 관계형 데이터베이스의 구조로 변환. 각 엔티티를 테이블로, 속성을 열로 변환함.
- 일대일, 일대다, 다대다 관계를 표현하기 위한 외래키를 설정함
- 식별관계 - 식별 관계는 자식 엔티티가 부모 엔티티의 주 식별자(Primary Key)를 포함하는 관계. 즉, 자식 엔티티의 주 식별자는 부모 엔티티의 주 식별자로 구성. 이 관계에서 자식 엔티티의 주 식별자는 부모 엔티티의 키 값과 결합하여 고유하게 식별됨. 실선으로 표시
- 비식별관계 - 비식별 관계는 자식 엔티티가 부모 엔티티의 주 식별자를 기본키로 포함하지 않는 관계임. 두 관계가 서로에 대해 필수 정보가 아님을 의미. 점선으로 표시
- 식별 관계와 비식별 관계는 외래키(Foreign Key)의 설정과 관련하여 기본키(Primary Key)와의 관계에 따라 결정됨.

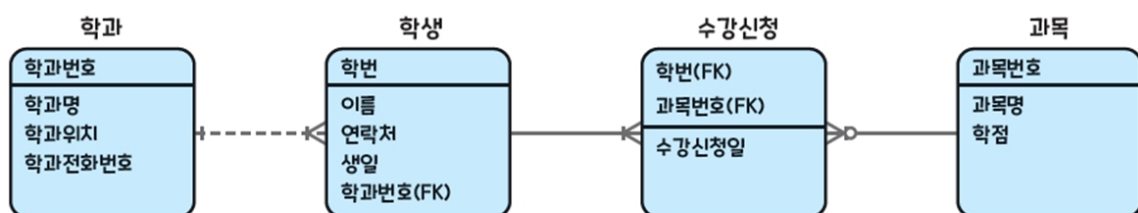


그림 12-13 IE 표기법을 사용하여 ERD로 표현한 논리적 데이터 모델의 예

### 4) 물리적 데이터 모델링

- 물리적 데이터 모델링에서는 논리적 모델을 실제 데이터베이스 관리 시스템 (DBMS)의 구조로 변환함. 데이터베이스 플랫폼에 맞춰 테이블 스페이스, 인덱스, 파티셔닝 등의 물리적 구성을 고려함. 이 단계에서는 성능, 보안, 확장성 등과 관련된 요구사항도 고려됨.
- 각 테이블을 특정 데이터베이스의 테이블로, 열의 데이터 형식과 크기를 지정함.

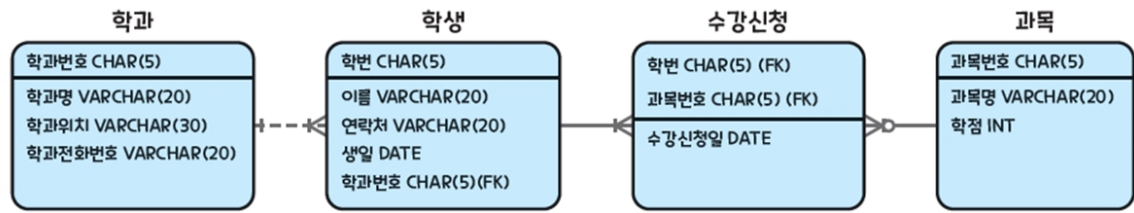


그림 12-14 물리적 데이터 모델의 예

- 5) 구현 및 유지보수
- 6) 검증 및 최적화, 문서화