



[AI초급] SQLD 자격증 코스 - 챕터 2



매 주차 강의자료 시작에 PDF파일을 올려두었어요!

▼ PDF 파일

[수업 목표]

- 모델링의 기본 개념과 특징을 이해하고 데이터베이스 구조에 대해 학습합니다 🧐

[목차]

01. 모델링의 이해

02. 데이터 모델링의 중요성

03. 3층 스키마 (3-Level Schema)

04. 데이터 모델링의 요소와 ERD



모든 토글을 열고 닫는 단축키

Windows : **Ctrl** + **alt** + **t**

Mac : **⌘** + **⌘** + **t**

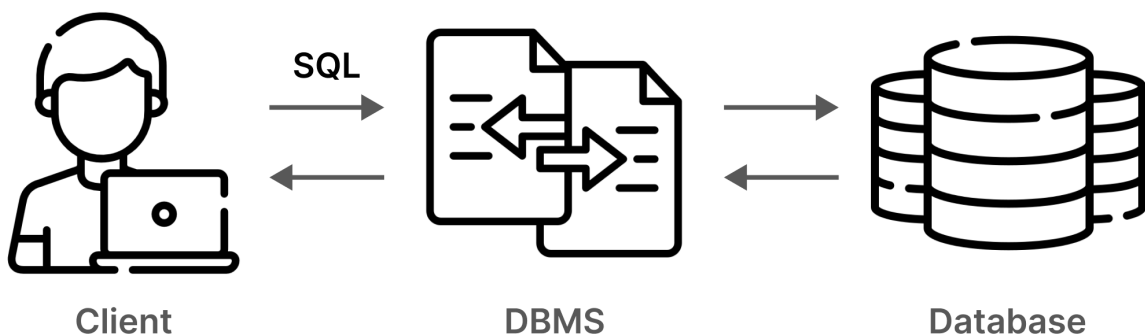
01. 모델링의 이해

✓ 데이터 모델과 모델링에 대해 이해해보자

▼ 1) 모델링의 정의

✓ 모델링의 정의

데이터베이스에서의 모델은 데이터베이스 관리 시스템(RDBMS)이 지원하는 데이터베이스의 구조나 형식을 의미합니다. 즉, 관리하는 데이터에 대한 청사진이며 우리는 모델 정보(구조)만 보고도 어떤 데이터를 다루는지 알 수 있습니다.



데이터 모델은 데이터를 기반으로 한 모델링의 과정이라고 할 수 있습니다. 결국 모델을 만드는 과정인 모델링은 **현실 세계를 얼마만큼 추상화시켜 표현할 것**인지 그림을 그리는 과정이고 그 결과를 토대로 세상을 단순화 시켜 명확하게 바라볼 수 있게 됩니다. 모델링은 모두 데이터를 기반으로 행해집니다. 세상에 존재하는 모든 것은 데이터로 표현 가능하고 그 데이터를 모아 구조화하고 집약하는 작업이 모델링이기 때문입니다.

데이터 모델은 데이터베이스의 큰 그림을 이해하고 그걸 바탕으로 SQL 문장을 작성하는데 필요한 매우 중요한 요소입니다. 데이터 모델에 대한 이해가 잘 되어 있으면 단순한 기능을 사용하는 것뿐만 아니라 성능적인 측면도 고려하여 SQL문을 작성할 수 있기 때문입니다. 이는 설계 도면을 기반으로 집을 짓는 것과 같고 목차를 통해 책의 전체 구조를 먼저 파악하는 것과 같은 이치입니다. 현업에서 SQL을 사용하는 과정도 SQL 문장을 어떻게 구성할지에 대한 지식과 효율적인 구성에 대한 지식이 필수적이기 때문에 모델링을 올바른 관점으로 바라보고 구성하는 것은 매우 중요합니다. 일반적으로 데이터 모델링은 다음과 같이 다양하게 정의합니다.



데이터 모델링의 정의

- 정보 시스템 구축을 위해 데이터 관점의 업무를 분석하는 과정
- 현실 세계의 데이터를 약속된 표기법에 의해 표현하는 과정
- 데이터베이스를 구축하기 위한 분석 및 설계의 과정

결국, 데이터 모델링의 목적을 살펴보면 크게 2가지로 정리할 수 있습니다.

▼ 1. 업무에 필요한 정보를 정확하게 '정의'하고 '표현'하여 **업무를 분석**

→ 개발하기 위해서만 데이터 모델링을 하는 것이 아님 !

- 2. 분석 모델을 통해 실제 데이터베이스를 생성하여 데이터를 관리

▼ 연습문제

Q. 문제	다음 중 모델링의 특징으로 가장 옳지 않은 것은?
A. 답지1	복잡한 현실을 제한된 표기로 이해하기 쉽게 단순화 하는 의미를 가짐
A. 답지2	모호함을 배제하고 모두가 이해할 수 있도록 정확하게 현실을 기술하는 의미를 가짐
A. 답지3	데이터베이스 구축 전 시스템 구현만을 위한 사전 단계의 작업의 의미를 가짐
A. 답지4	현실 세계를 약속된 표기를 사용해 추상화하는 의미를 가짐

▼ 정답

(3) - 데이터 모델링은 구현만을 위한 단계가 아님

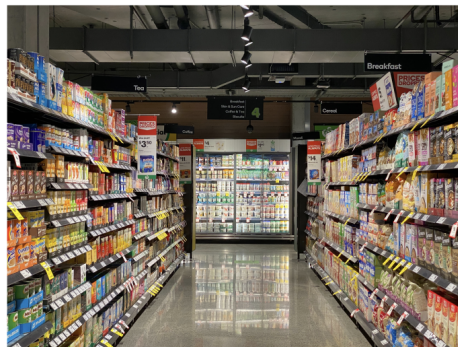
데이터 모델이 제공하는 기능

업무를 분석하는 관점에서 데이터 모델이 제공하는 기능은 여러 가지가 있습니다.

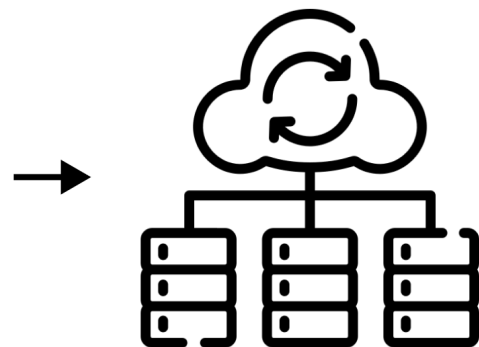
기능	설명
시각화	시스템을 원하는 모습으로 시각화해서 보여줄 수 있도록 한다
문서화	시스템의 구조와 행동을 문서화한다

기능	설명
구체화	특정한 목표에 따라 구체화된 상세 수준의 표현 방법을 제공한다
구조화된 틀 제공	시스템을 구축하는 구조화된 틀을 제공한다
다양한 관점 제공	다양한 영역에 집중하기 위해 다른 영역의 세부 사항은 숨기는 다양한 관점을 제공한다

▼ 2) 모델링의 특징



현실세계



데이터 모델

특징	설명
추상화 (Abstraction)	추상화는 현실 세계를 <u>일정한 형식에 맞추어 표현하는 것</u> 을 의미합니다. 모델링은 세상에 존재하는 수많은 대상을 특정한 기준에 따라 나누고 나뉜 기준의 상위 기준을 모아서 또 다른 기준으로 구분하여 나누는 과정을 반복합니다. 위와 같은 과정을 계속 따라갔을 수많은 요소를 하나의 문장, 단어, 개념으로 표현할 수 있는 순간에 도달하게 되는데, 이는 결국 모델링을 통한 추상화 과정이라고 볼 수 있습니다.
단순화 (Simplification)	단순화는 <u>복잡한 현실 세계를 약속된 규칙에 기반한 제한된 표기법이나 언어로 표현하여 쉽게 이해할 수 있도록 하는 것</u> 입니다. 현실 세계에 존재하는 수많은 요소를 정확하게 분류하여 나누고 구분하는 것은 현실적으로 불가능합니다. 모델링은 복잡한 세상을 단순화 시킬 수 있도록 구조화합니다.
명확화 (Clarity)	명확화는 누구나 이해하기 쉽게 <u>대상에 대한 애매모호함을 제거하고 현상을 정확하게 기술하는 것</u> 을 의미합니다. 현실 세계에 존재하는 데이터를 정리하고 단순화 시키는 과정에서 한 대상이 갖는 의미를 명확하게 기술합니다. 모델링의 과정 속에는 대상을 명확화하고 구체화 시

특징	설명
	키는 과정 뿐만 아니라 그 대상을 어떻게 표기하고 기술할 것인지를 고민하는 과정이 포함되며 이는 명확성을 증가시키는 과정이라고 볼 수 있습니다.

▼ 3) 데이터모델링 3단계 진행

✓ 데이터모델링의 단계

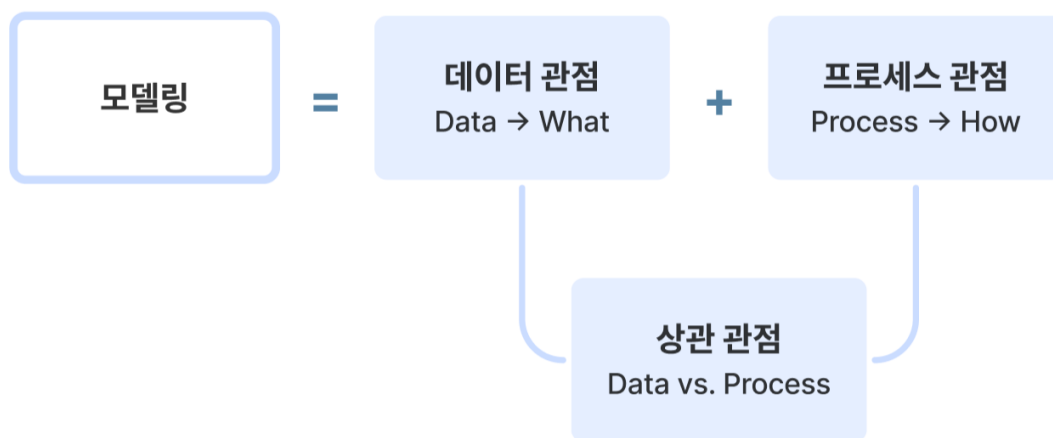


데이터 모델링 단계	설명
개념적 데이터 모델링 Conceptual Data Modeling	- 조직이나 사용자가 필요로 하는 <u>데이터의 요구사항을 찾고 분석하는 과정</u> - 복잡하지 않고 중요한 부분을 위주로 모델링하는 단계 - <u>추상화 수준이 가장 높고 업무 중심적인 모델링</u> - 전사적 관점에서 기업의 데이터 모델링
논리적 데이터 모델링 Logical Data Modeling	- 비즈니스 과정에서 나타나는 <u>정보의 논리적인 구조와 규칙을 명확하게 표현하는 기법/과정</u> - 누가(Who), 어떻게 (How: Process) 그리고 전산화와는 별개로 비즈니스 데이터에 존재하는 사실을 인식하여 기록하는 것 - 정규화*를 수행하여 데이터 모델의 독립성 확보 *정규화: 논리 데이터 모델의 일관성을 확보하고 중복을 제거하여 보다 신뢰성 있는 데이터 구조를 얻는 방법
물리적 데이터 모델링 Physical Data Modeling	- 논리적 데이터 모델이 데이터 저장소로서 어떻게 컴퓨터 하드웨어에 표현될 것인지를 다루는 과정 - 구축할 데이터베이스 관리 시스템에 테이블, 인덱스 등을 생성하는 단계 - 성능, 보안, 가용성을 고려하여 구축

✓ 데이터모델링의 관점

업무 상황에서 훨씬 더 정제되고 일관적인 데이터 수집이 가능하기 때문에 모델링은 일상보단 업무적인 환경에서 더 많이 사용됩니다. 그렇기 때문에 관련된 용어들이 업무적인 환경과 더 깊이 연결되어 있는 경우가 많습니다. 이러한 관점에서의 모델링은 **시스템의 대상이 되는 업무를 분석하여 정보 시스템으로 구성하는 과정에서 업무의 내용과 그 모습을 적절한 표기법(Notation)으로 표현**하는 것을 의미합니다.

이번에는 모델링된 결과물을 바라보는 3가지 관점인 데이터 관점, 프로세스 관점, 데이터와 프로세스의 상관관점에 대해서 알아보겠습니다.



관점	설명
데이터 관점 (What)	- 업무가 <u>어떤 데이터와 관련 있는지</u> 모델링하는 방법에 대해 고민하는 관점 - 비즈니스 프로세스에서 사용되는 데이터
프로세스 관점 (How)	- 업무가 <u>실제 하는 일이 무엇인지</u> 그리고 <u>무엇을 해야 하는지</u> 에 대해 모델링하는 방법을 고민하는 관점 - 도메인 분석, 시나리오 분석
데이터와 프로세스의 상관 관점 (Intersection)	- 업무가 처리하는 일의 방법에 따라 데이터는 어떤 영향을 받고 있는지를 중심으로 모델링을 하는 방법 - <u>일에 의해 데이터가 어떤 변화가 일어나는지</u> 에 대해 초점을 맞추는 관점 - CRUD(Create, Read, Update, Delete) 분석

▼ 연습문제

✓ 문제 1

Q. 문제	다음 중 데이터 모델링의 주요 특징으로 옳지 않은 것은?
A. (1)	추상화
A. (2)	단순화
A. (3)	구체화
A. (4)	명확화

▼ 정답

(3) 구체화

✓ 문제2

Q. 문제	다음 중 데이터 모델링 절차로 올바른 것은?
A. (1)	논리적 모델링 → 개념적 모델링 → 물리적 모델링
A. (2)	물리적 모델링 → 논리적 모델링 → 개념적 모델링
A. (3)	개념적 모델링 → 물리적 모델링 → 논리적 모델링
A. (4)	개념적 모델링 → 논리적 모델링 → 물리적 모델링

▼ 정답

(4) 개념적 모델링 → 논리적 모델링 → 물리적 모델링

✓ 문제3

Q. 문제	다음 중 데이터 모델링의 3가지 관점이 아닌 것은?
A. (1)	What
A. (2)	Where
A. (3)	Intersection
A. (4)	How

▼ 정답

(2) Where

02. 데이터 모델링의 중요성



모델링의 이해를 바탕으로 데이터 모델링의 중요성과 프로젝트 관리 방법론을 학습합니다

▼ 1) 데이터 모델링의 중요성 및 유의점



파급효과 (Leverage)

시스템 구축이 완성되어 가는 과정에서는 대규모 데이터를 통한 분석을 성공적으로 수행하기 위해 수많은 단위의 테스트가 수행됩니다. 결과가 완료되면 비슷한 규모(단위)의 테스트를 묶어 병합병행 테스트를 진행합니다. 이런 과정이 데이터를 업무 단위에서 분석하여 결과를 내는 과정입니다. 자, 이 상황에서 만약에 데이터 모델의 변경이 발생한다면 어떤 일이 벌어질까요? 정말 불가피한 상황을 가정해 보는 거예요.

예를 들면, 회사에서 진행하는 서비스의 방향이 바뀌어서 데이터를 정의하는 속성이 꽤나 많이 변경되어야 하는 상황이라고 해보는 거예요. 큰 구조가 변하지 않았다면 기존 내용을 바꾸기만 하면 되겠지만 만약 모델링 된 결과물 자체가 이전 요소에 완전하게 종속된 형태라면 어떨까요? 그럼 그 결과로 데이터 구조의 변경에 따른 표준 영향 분석, 응용 변경 등의 영향 분석 등의 과정이 변경되어야 하겠죠? 변경을 해야 하는 데이터 모델의 형태에 따라서 얼마만큼의 영향이 받는지 차이가 있겠지만 이 시기의 데이터 구조의 변경으로 인한 일련의 변경 작업은 전체 시스템 구축 과정에서 큰 위협요소가 될 수 있습니다.

결과적으로 모델링을 잘 설계하는 위해서는 데이터가 흘러가는 것을 시스템 전체의 관점에서 바라볼 수 있어야 합니다. 구체적인 내용은 변해도 큰 구조는 변하지 않도록 만드는 것이죠. 어떤 설계인지에 따라 예상하지 못한 상황에 얼마만큼의 추가 비용을 초래할지 결정되는 만큼 중요하다고 할 수 있습니다.



간결한 표현 (Conciseness)

데이터 모델은 구축할 시스템의 정보에 대한 요구 사항과 한계점을 가장 명확하고 간결하게 표현할 수 있는 도구입니다. 정보에 대한 요구 사항을 파악할 때 수십 페이지 분량의 자료의 기능적인 요구 사항을 하나하나 파악하는 것보다 간결하지만 핵심이 포함된 데이터 모델을 보면서 파악하는 것이 훨씬 좋습니다.

데이터 모델은 시스템을 구축하는 많은 관련자들이 설계자의 생각대로 정보요구 사항을 이해하고 이를 운용할 수 있는 어플리케이션을 개발하고 데이터의 일관성을 유지할 수 있도록 하는 것입니다.

이렇게 이상적인 역할 수행을 할 수 있는 데이터 모델이 갖추어야 할 가장 중요한 점은 정보 요구 사항이 정확하고 간결하게 표현되어야 한다는 것입니다.

✓ 데이터 품질 (Data Quality)

데이터베이스에 저장된 데이터는 기업에게는 매우 소중한 자산입니다. 구글 검색을 통해 수많은 자료를 무료로 이용하는 것도 결국 우리가 검색하는 과정 하나하나의 데이터를 구글이 활용하기 때문입니다. 데이터는 기업 입장에서 매우 소중한 자산입니다.

그렇다면 기업에게 모든 데이터는 소중한가요? 그건 아닙니다. 저장된 데이터가 어떤 정보를 제공함에 있어 정확하지 않은 데이터일 수도 있고 일관성이 떨어지는 데이터일 수도 있습니다. 일단 많이 저장만 하면 된다고 생각하시는 분들이 있을 수 있는데요, 기업에게 데이터는 의사 결정의 기준이 되기 때문에 정확하지 않은 정보로 인한 피해는 상상 이상일 수도 있습니다. 잘못된 데이터로 인해 잘못된 판단을 내리면 소중한 비즈니스 기회를 잃게 될 수도 있습니다.

데이터 품질의 문제가 발생하는 중요한 이유 중 하나는 데이터 구조의 문제 때문입니다. 중복된 데이터를 정의하지 않고 데이터 구조 내에서 비즈니스에 대한 정의가 충분하지 않은 경우에 발생할 수 있습니다. 그렇다면 모델링을 하는 시점에서 어떤 기준을 가지고 데이터 품질에 대한 고민을 해볼 수 있을까요? 데이터 품질에 대한 평가는 크게 3가지 기준을 통해 살펴볼 수 있습니다.

▣ 중복 (Duplication)

- 데이터베이스 여러 곳에 같은 정보를 중복해서 저장하는 행위입니다.

▣ 비유연성 (Inflexibility)

- 환경이 바뀌었을 때 데이터가 사용 가능 여부입니다.
- 데이터 모델을 어떻게 설계했느냐에 따라 사소한 업무의 변화에도 데이터 모델의 유지 보수가 쉬울 수도 있고 어려울 수도 있습니다.

▣ 비일관성 (Inconsistency)

- 데이터의 중복이 없다고 해도 일관적이지 않은 데이터가 나타날 수 있습니다.
- 데이터 모델링을 할 때는 데이터와 데이터 간의 상호 연관 관계에 대한 명확한 정의를 해야 할 필요가 있습니다.

▼ 2) 프로젝트 라이프 사이클에서 데이터모델링

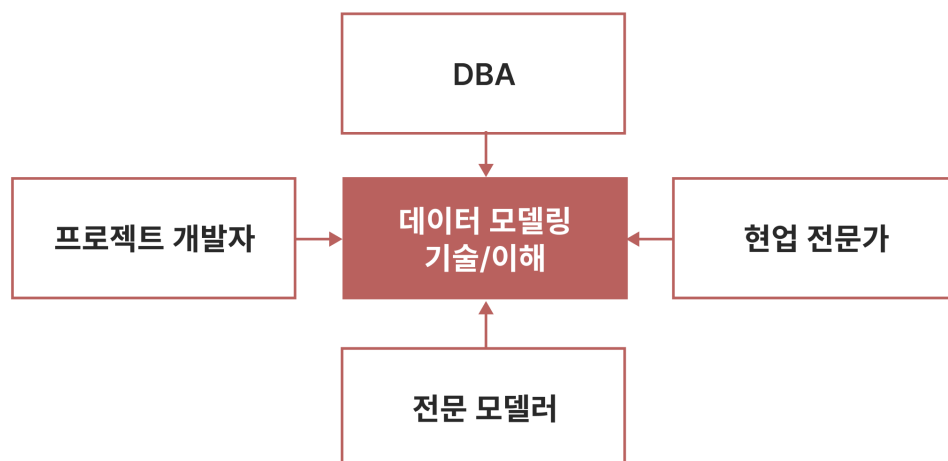
프로젝트 라이프 사이클이란?

프로젝트의 시작부터 완료에 이르기까지 거치는 일련의 단계를 의미합니다. 프로젝트 관리 방법론에는 크게 폭포수 모델(Waterfall)과 애자일 모델(Agile)이 있으며, 프로젝트의 범위가 명확할 때 폭포수 방법론(Waterfall)을 적용합니다.

프로젝트 라이프 사이클 (Waterfall 기반)	정보공학, 구조적 방법론	개발
분석	논리 및 개념 데이터 모델링	프로세스 모델링
설계	물리 데이터 모델링	AP 설계
개발	DB 구축, 변경, 관리	AP 개발
테스트	DB 튜닝	AP 테스트
전환/이행	DB 전환	AP 설치

▼ 3) 데이터 모델링의 이해관계자

정보시스템을 구축하는 모든 사람은 데이터 모델링을 전문적으로 할 수 있거나 적어도 완성된 모델을 정확하게 해석할 수 있어야 합니다. 특히 개발자는 데이터 모델링 기술에 대한 매우 수준 높은 이해도를 가지고 있어야 합니다. 더불어 IT 비종사자, 비전공자도 정보화와 관련된 업무를 하고 있다면 데이터 모델링에 대한 개념과 세부사항에 대해 알아야 합니다. 잘못된 시스템을 구축하는 위험(risk)이 없도록 원활한 업무 소통이 가능해야하기 때문입니다.



▼ 연습문제

✓ 문제 1

Q. 문제	데이터 모델링의 중요성으로 옳지 않은 것은?
A. (1)	파급효과

A. (2)	간결한 표현
A. (3)	중복효과
A. (4)	데이터 품질

▼ 정답

(3) 중복효과

✓ 문제2

Q. 문제	데이터 품질 평가 기준으로 옳지 않은 것은?
A. (1)	중복
A. (2)	향상성
A. (3)	비유연성
A. (4)	비일관성

▼ 정답

(2) 향상성

✓ 문제3

Q. 문제	프로젝트 라이프사이클 '설계' 단계에서 수행되는 모델링은?
A. (1)	개념 데이터 모델링
A. (2)	논리 데이터 모델링
A. (3)	물리 데이터 모델링
A. (4)	논리 및 개념 데이터 모델링

▼ 정답

(3) 물리 데이터 모델링

- 설계 및 구현 단계의 모델링이다.

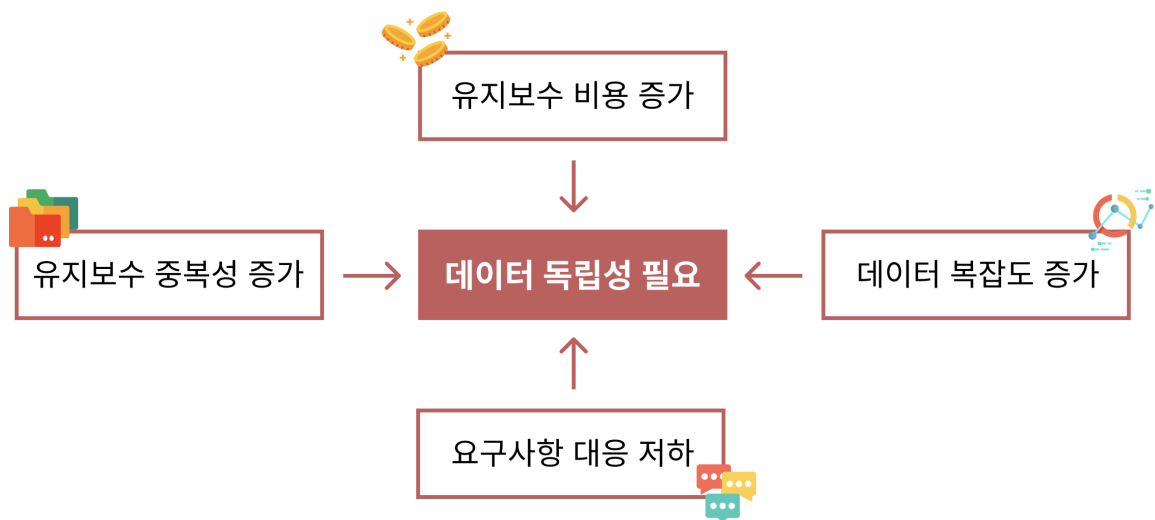
03. 3층 스키마 (3-Level Schema)

✓ 데이터의 독립성 확보에 대한 방법을 학습합니다

▼ 1) 데이터 독립성의 필요성

데이터 모델링의 과정에서 신경 써야 하는 것 중 하나는 데이터의 일체적 구성입니다. 일체적 구성이라고 하는 것은 **일관된 형태로 데이터를 수집하는 것**입니다. 다른 말로 표현하자면 **데이터의 독립적 구성**이라고 할 수 있습니다.

데이터 독립성의 반대말은 데이터 종속성입니다. 여기서 종속의 주체는 보통 응용(Application)을 의미합니다. 응용은 사용자의 요구사항을 처리하는 사용자 접점의 인터페이스를 일컫습니다. 과거에는 파일 방식으로 데이터를 구성하는 것이 일반적이었습니다. 사용자가 접근하는 방법 (트랜잭션의 유형)에 따라 파일의 정렬순서, 인덱스의 정렬순서, 파일 구성 등을 제공하기 쉽게 별도로 구성했습니다. 즉, 사용자가 어떤 방식으로 파일에 접근하냐에 따라서 데이터를 구성하는 방법이 영향을 받았습니다. 하지만 1990년대 이후 Client & Server 모델이 주를 이루며 파일 단위의 데이터베이스 관리가 아닌 원격 서버에 저장해두는 방식이 보편화되면서 파일 처리 방식의 변화가 생기게 됩니다.



데이터 독립성을 확보하게 되면 아래와 같은 기대 효과를 누릴 수 있습니다.

- 각 View의 독립성을 유지하고 계층별 View에 영향을 주지 않고 변경이 가능합니다.

- 단계별 Schema에 따라 데이터 정의어(DDL)와 데이터 조작어(DML)가 다르게 제공됩니다.

→ **응용 프로그램과 물리적 데이터베이스를 분리하자!** 👁👁

▼ 2) 데이터베이스 3단계 구조 (3층 스키마)

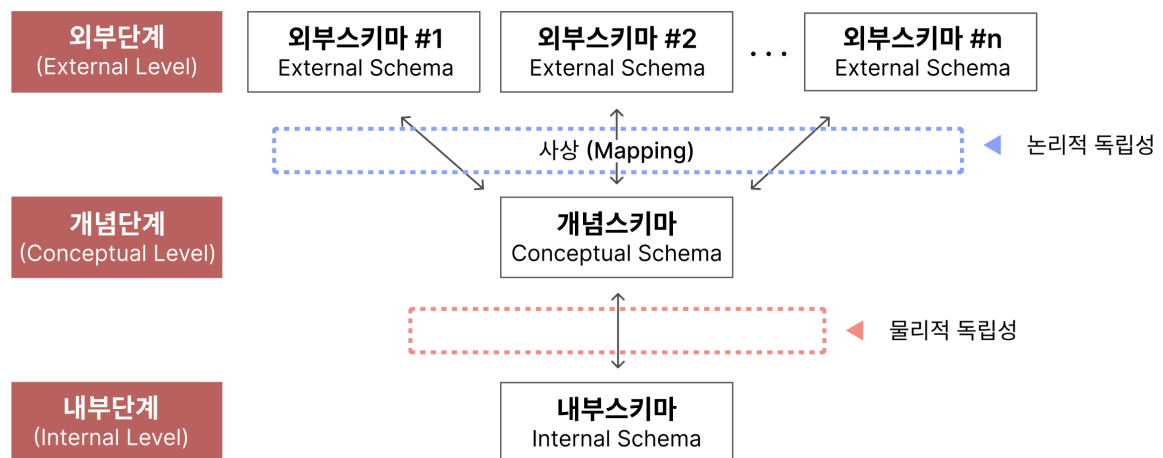
✓ 3층 스키마란?

3층 스키마란, 데이터베이스를 보는 관점에 따라 데이터베이스를 기술하고 이들 간의 관계를 정의한 ANSI 표준입니다. 3단계 계층으로 분리하여 독립성을 확보해 각 계층을 뷰(View)라고도 합니다.

*ANSI : 미국표준협회 (American National Standards Institute)

→ 사용자, 설계자, 개발자가 데이터베이스를 보는 관점에 따라 데이터베이스를 기술하고 이들 간의 관계를 정의한 표준

✓ 3층 스키마의 구조



→ 유저 어플리케이션(프로그램)은 데이터베이스 시스템의 최상위 단계

물리적 데이터베이스는 최하위에 위치

이 사이를 세 개의 단계(스키마)로 나눠보자는 것 👁👁

항목	내용	비고
외부 스키마 (External Schema)	- View 단계 여러 개의 사용자 관점으로 구성합니다. - 개별 사용자가 보는 DB 스키마입니다. - 실	- 사용자 관점 - 접근하는 특성에 따른 스키마를 구성

항목	내용	비고
	제로 관심 있는 데이터베이스 부분을 설명하고 나머지는 감춥니다.	
개념 스키마 (Conceptual Schema)	- 데이터 베이스의 물리적인 저장 구조에 대한 부분은 숨기고, 데이터의 전체적인 구조와 관계에 대해 집중합니다. - 모든 사용자 관점을 통합한 조직 전체의 DB를 기술합니다. - 모든 응용 시스템들이나 사용자들이 필요로 하는 데이터를 통합한 조직 전체의 DB를 기술한 것입니다. - DB에 저장되는 데이터와 그들간의 관계를 표현하는 스키마입니다.	- 설계자 관점, 통합 관점 - 통합 데이터베이스 구조
내부 스키마 (Internal Schema)	- 내부 단계, 내부 스키마로 구성합니다. - DB가 물리적으로 저장된 형식입니다. - 물리적 장치에서 데이터가 실제로 저장되는 완전히 구체적인 방법을 표현하는 스키마입니다.	- 개발자 관점 - 물리적 저장 구조

▼ 조금 더 쉽게 이해해보자

- 데이터베이스를 3가지의 큰 범주로 분리해보자라는 컨셉
- 사상(mapping)은 각 각의 범주간의 요청/응답을 전송하는 것
- 외부 스키마에서 요청이 들어오면 DBMS에 의해 개념 스키마 - 내부 스키마로 전달됨
여기에서 요청과 응답을 변환하는 프로세스를 사상(mapping) 이라고 함
- 내부 스키마
 - 물리적 저장 구조를 갖춘 모델
- 개념 스키마
 - 전체 데이터베이스의 설계를 설명할 수 있는 모델 (구조, 관계등)
 - 데이터 구조의 구현 정보와 같은 내부 상세 정보는 보지 못함

데이터 독립성 (Data Independence)

- 상위 스키마를 변경하지 않고 하나의 계층에서 스키마를 변경할 수 있는 능력
- 데이터베이스의 여러 레벨에서 구조를 수정할 때, 하위 레벨 스키마를 변경하더라도 상위 레벨의 스키마를 건드릴 필요가 없는 것 (영향을 미치지 않음)

✓ 3층 스키마의 독립성

독립성	설명	특징
논리적 독립성	- 개념 스키마가 변경되어도 외부 스키마에는 영향을 미치지 않도록 지원합니다. - 논리적 구조가 변경되어도 응용 프로그램에 영향이 없습니다.	- 사용자 특성에 맞게 변경이 가능합니다. - 통합 구조로 변경 가능합니다.
물리적 독립성	- 내부 스키마가 변경되어도 개념 스키마는 영향을 받지 않도록 지원합니다. - 저장 장치의 구조 변경은 응용프로그램과 개념 스키마에 영향을 주지 않습니다.	- 물리적 구조의 영향 없이 개념 구조로 변경 가능합니다. - 개념 구조의 영향 없이 물리적인 구조로 변경 가능합니다.

- 논리적 독립성
 - 우리가 데이터 저장이 필요한 속성을 추가/변경이 필요할 때 외부 응용 프로그램을 변경하지 않아도 됨
- 물리적 독립성
 - 실제로 데이터를 저장하는 위치나 파일 이름을 변경하더라도 데이터베이스 전체 관점 즉, 개념 스키마 관점에서는 아무런 영향이 없음

예시)

- CS팀 aiden(외부 스키마 #1)이라고 하고, 이벤트 상품 전달을 위해 고객 정보가 필요한 상황이라면, 필요한 정보는 고객의 이름, 전화번호, 주소이다.
 - 하지만 회사가 저장하고 있는 정보는 고객 비밀번호, 계좌번호, 캐시, 이메일, 닉네임 등 굉장히 많은 정보가 있는 상황.
 - 즉 외부 스키마는, 개념 스키마를 이용하여 필요한 정보만 전달받을 수 있다
- CS팀 mae(외부 스키마 #2)가 14세가 안된 고객은 우리 서비스에서 거래를 못하게 해야한다고 한다. 즉, 14세 미만인지 아닌지 여부를 새롭게 저장해야하는 상황 🙄
 - 이때, 14세 이상인지 여부를 판단하는 데이터가 추가되었다고해서 앞서 aiden(외부 스키마 #1)이 하던 고객 정보에 대한 접근이 안된다면 물리적 / 논리적 독립성 모두 없는 것

⇒ 검색 속도를 높이기 위해 데이터가 저장되는 파일의 구조를 바꿨다고 해서 (내부 스키마), 전체적인 데이터 베이스 구조/설계가 달라지거나 (개념 스키마) 응용 프로그램단 (외부 스키마)이 변경되면 안된다는 것 🙄

⇒ 데이터베이스를 바라보는 관점에 따라 세가지로 나눌 수 있구나! 🙄
그 각각은 논리적 / 물리적으로 독립적이면 좋다는거네 !

▼ 연습문제

✓ 문제 1

Q. 문제	데이터 독립성의 필요성으로 옳지 않은 것은?
A. (1)	데이터 복잡도 증가
A. (2)	요구사항 대응 증가
A. (3)	유지보수 중복성 증가
A. (4)	유지보수 비용 증가

▼ 정답

(2) 요구사항 대응 증가

✓ 문제2

Q. 문제	3층 스키마에서 물리적 저장 구조를 의미하는 것은?
A. (1)	외부 스키마
A. (2)	개념 스키마
A. (3)	내부 스키마
A. (4)	논리 스키마

▼ 정답

- (3) 내부 스키마

✓ 문제3

Q. 문제	논리적 독립성에 대한 설명으로 옳지 않은 것은?
A. (1)	물리적 구조의 영향 없이 개념 구조로 변경 가능

A. (2)	사용자 특성에 맞게 변경 가능
A. (3)	통합 구조로 변경 가능
A. (4)	개념 스키마가 변경되어도 외부 스키마에는 영향을 미치지 않도록 지원 가능

▼ 정답

(1) 물리적 구조의 영향 없이 개념 구조로 변경 가능

04. 데이터 모델링의 요소와 ERD

✓ 데이터 모델링의 중요한 3가지 개념에 대해 학습합니다

▼ 1) 데이터 모델링의 중요 요소

✓ 데이터 모델링의 중요한 3가지 개념

데이터 모델링을 구성하는 3가지 중요 개념이 있습니다. 이 요소는 데이터 모델링을 하는 과정에서의 가장 기본이 되기 때문에 반드시 기억해야 합니다.



운전도 명칭부터 익히듯 !

여기서는 용어 위주로 간단히 알아보고 뒤에서 각각에 대해 조금 더 자세히 알아보도록 할게요 🙄

1. 엔티티(Entity) - 업무가 관여하는 어떤 것(thing)

- 사물이나 사건 등을 바라볼 때 전체를 지칭하는 용어입니다. 이 용어를 통해서 업무가 어떤 부분을 중심으로 벌어지고 **초점이 맞춰져있는지**를 확인할 수 있습니다.
- 예를 들어, 우리 모두는 개개인이 하나의 Entity라고 볼 수 있습니다. 사물, 사건이 어떤지에 따라 작은 차이는 있을 수 있습니다.

→ 눈에 보이는 개념이든 아니든 데이터 모델링에서 사용되는 하나의 대상, 객체

2. 속성(Attribute) - 어떤 것이 갖는 성격

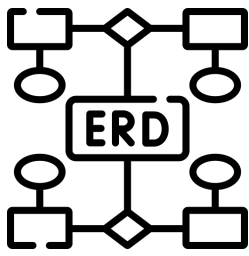
- 어떤 것이 갖는 세부적인 성격을 의미합니다. **Entity가 지닐 수 있는 여러 특징**으로 볼 수 있습니다.
- 예를 들어, 여러분 각자가 같은 어떤 특징입니다. 키, 몸무게, 성격, 취미 등 다양한 것이 속성에 해당합니다.

3. 관계(Relationship) - 업무가 갖는 어떤 것 간의 관계

- **Entity와 Entity가 서로 간의 관계**를 가질 수 있는데, 둘 사이에는 어떠한 연결 고리(연관성)가 존재합니다.
- 예를 들어, 여러 분들 간의 관계가 친구 사이일 수도 있고 연인 사이일 수도 있습니다. 가족 관계, 직장 선후배 관계가 될 수도 있습니다.
- 헬스장이라면 트레이너 엔티티, 고객 엔티티는 어떠한 관계를 맺고 있을까요? 🙄
한 명의 트레이너가 여러명의 고객을 응대하는 관계, 즉 1:N 관계를 맺고 있을거예요.
(뒤에서 다룹니다)

▼ 2) 데이터모델링을 위한 ERD

✓ ERD 정의



ERD란?

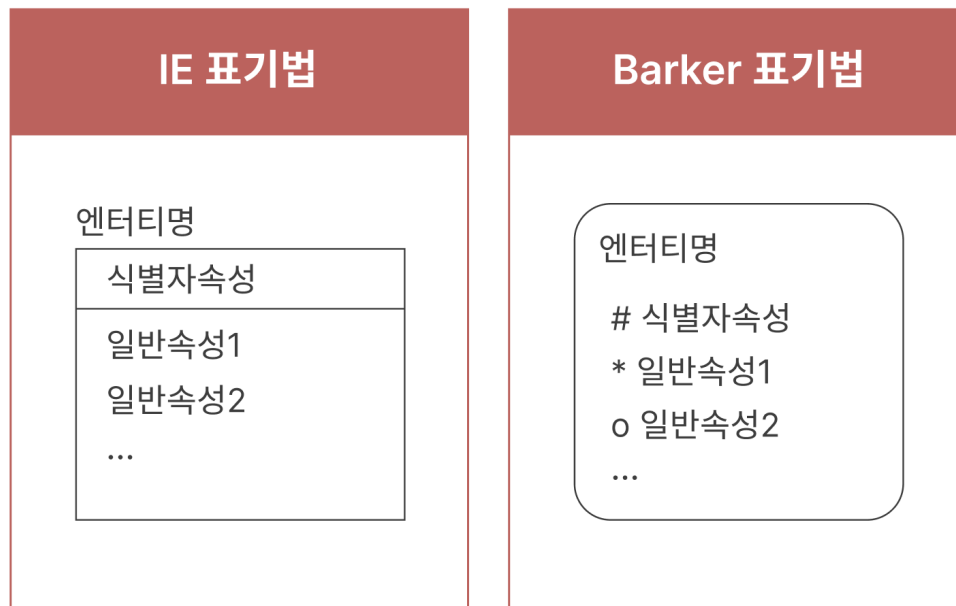
Entity Relationship Diagram의 약자로 데이터들의 관계를 나타낸 도표입니다. 1976년 피터첸(Peter Chen)이 Entity Relationship Model 표기법을 만들어 ERD데이터 모델링의 표준으로 사용되고 있습니다.

✓ ERD 작성법

1. 엔터티를 정의하고 그린다.
2. 엔터티를 적절하게 배치한다.
 - 가장 중요한 엔터티를 좌측 상단에 배치하고 이것을 중심으로 다른 엔터티들을 나열한다.
(왼쪽에서 오른쪽, 위쪽에서 아래쪽으로)
3. 엔터티간의 관계를 설정한다.
4. 관계명을 서술한다.
5. 관계의 참여도를 기술한다.
 - 특정 엔터티와 다른 엔터티 간의 관계수를 의미해요.
6. 관계의 필수 여부를 기술한다.

✓ 데이터 모델 표기법

대표적인 데이터 모델 표기법은 IE/Crow's Foot 표기법과 Barker/Case*Method 표기법이 있습니다.

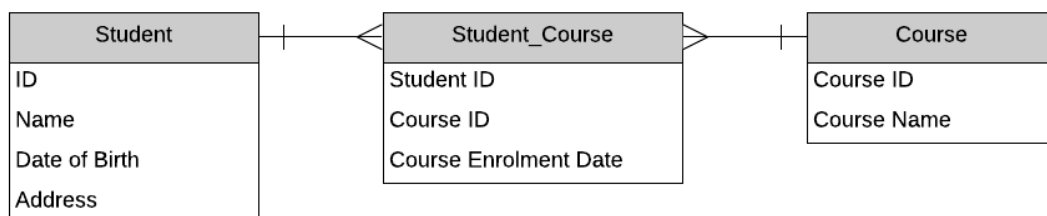


엔터티(Entity)와 속성(Attribute) 표기

• Barker 표기법 참고사항

- #: 식별자 속성 앞에 표기
- *: 필수 속성 앞에 표기
- o: 선택 속성 앞에 표기

식별자: 각각의 데이터를 구분해주는 속성 (뒤에서 자세히 배워요 🙄)



어떤 표기법일까요?

▼ 3) 좋은 데이터 모델의 요소

좋은 데이터 모델을 평가하는 절대 기준은 존재하지 않습니다. 기본적으로 데이터 모델은 상황 맥락에 영향을 받기 때문에 현시점의 최선이 시간이 흐르면 최악 모델이 될 수도 있습니다. 그럼에도 전반적인 상황에서 좋은 데이터 모델로 평가할 수 있는 몇 가지 요소를 살펴보면 다음과 같습니다.

✓ 완전성(Completeness)

- 업무에서 필요로 하는 모든 데이터가 모델에 정의되어 있어야 합니다.
- 예를 들면, 학교 데이터 모델은 학생과 선생님에 대한 정보가 포함되어 있어야 하며 보험사의 데이터 모델은 고객에 대한 정보가 포함되어 있어야 합니다.

✓ 중복 제재(Non-Redundancy)

- 하나의 데이터베이스 내에 동일한 사실은 한 번만 기록해야 합니다.
- 예를 들어, 하나의 테이블에서 '나이'라는 칼럼과 '생년월일'이라는 칼럼이 동시에 존재하면 이는 중복 제재 원칙을 위반했다고 볼 수 있습니다.

✓ 업무 규칙(Business Rules)

- 데이터 모델에서 매우 중요한 요소 중 하나는 데이터 모델링 과정에서 도출되고 규명되는 수많은 업무규칙(Business Rules)을 데이터 모델에 표현하고 이를 해당 데이터 모델을 활용하는 모든 사용자가 공유할 수 있도록 제공하는 것입니다. 특히, 데이터 아키텍처에서 언급되는 논리 데이터 모델(Logical Data Model)에서 이러한 요소들이 포함되어야 합니다.
- 예를 들면, 보험사의 직원들은 매월 여러 가지 항목에 대해서 급여를 지급받고 있고 이를 데이터로 관리하고 있습니다. 각 직원들은 월별로 하나 이상의 급여 항목(기본급, 상여금, 수당, 수수료, 등등)에 대해서 급여를 지급받고 있습니다. 나아가 각 직원은 직원 구분별(내근, 설계사, 계약직, 대리점 등)로 위의 급여 항목을 차등적으로 지급받는다는 업무규칙에 따라 돈을 받습니다. 이러한 내용은 데이터 모델에 나타나야 합니다. 이렇게 함으로써 해당 데이터 모델을 사용하는 모든 사용자(개발자, 관리자 등)가 해당 규칙에 대해서 동일한 판단을 하고 데이터를 조작할 수 있게 됩니다.

✓ 데이터 재사용(Data Reusability)

- 데이터는 언제든지 다시 사용할 수 있는 형태로 가공되고 보관되어야 합니다.

✓ 의사소통(Communication)

- 데이터 모델은 의사소통 도구로서의 역할을 해야 합니다.

✓ 통합성(Integration)

- 동일한 데이터 구조는 데이터를 구성하는 조직 전체에서 한 번만 정의되어야 합니다.

- 바람직한 데이터 구조의 형태는 동일한 데이터일 경우 조직의 전체에서 한 번 만 정의되고 이를 여러 다른 영역에서 참조, 활용하는 것입니다.

▼ 연습문제

✓ 문제 1

Q. 문제	데이터 모델링의 중요 요소가 아닌 것은?
A. (1)	Attribute
A. (2)	Entity
A. (3)	Relationship
A. (4)	External

▼ 정답

(4) External

✓ 문제2

Q. 문제	아래 보기는 ERD를 그리는 각각의 절차이다. ERD 작성 순서로 옳은 것은? <보기> 가. 관계명을 서술한다 나. 관계의 필수 여부를 기술한다 다. 엔터티를 정의하고 그린다 라. 엔터티간의 관계를 설정한다 마. 관계의 참여도를 기술한다 바. 엔터티를 적절하게 배치한다
A. (1)	라 → 다 → 바 → 마 → 가 → 나
A. (2)	가 → 나 → 다 → 라 → 마 → 바
A. (3)	다 → 라 → 바 → 가 → 나 → 마
A. (4)	다 → 바 → 라 → 가 → 마 → 나

▼ 정답

(4) 다 → 바 → 라 → 가 → 마 → 나