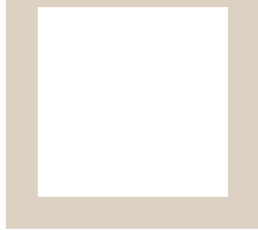


# 데이터베이스시스템



## 03. 데이터 모델링과 설계



나 홍 석    교수



고려사이버대학교  
THE CYBER UNIVERSITY OF KOREA



3  
LESSON

# 데이터 모델링과 설계

# 학습 목표

- 1** 데이터모델링의 개념을 이해하고 필요성을 설명할 수 있다.
- 2** 데이터모델링 과정을 단계별로 설명할 수 있다.

# 학습 내용

## 1 데이터 모델링

---

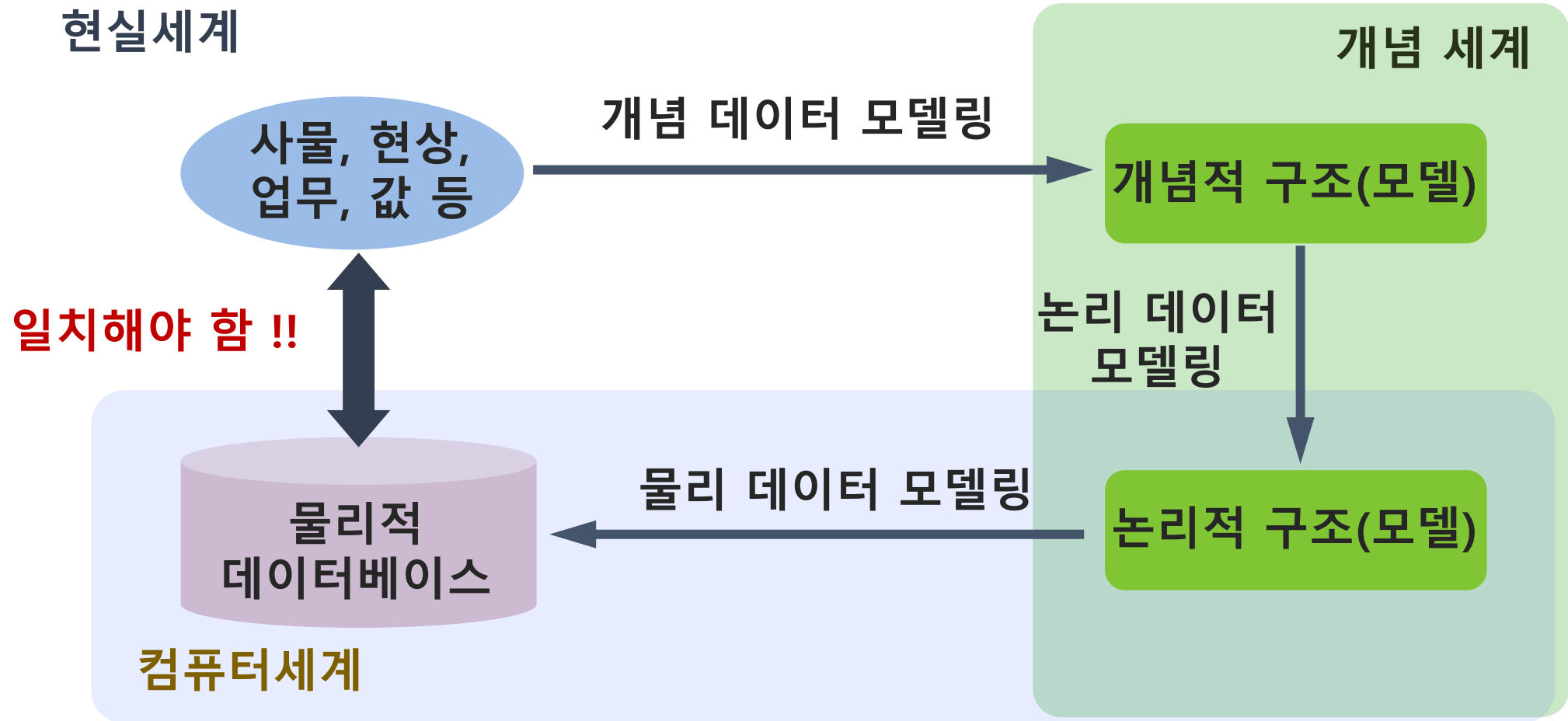
## 2 데이터베이스 설계

---

# Chapter 01 데이터 모델링

# 1 데이터 모델

## 1 데이터의 세계



# 1 데이터 모델

## 2 모델과 데이터모델

### 모델

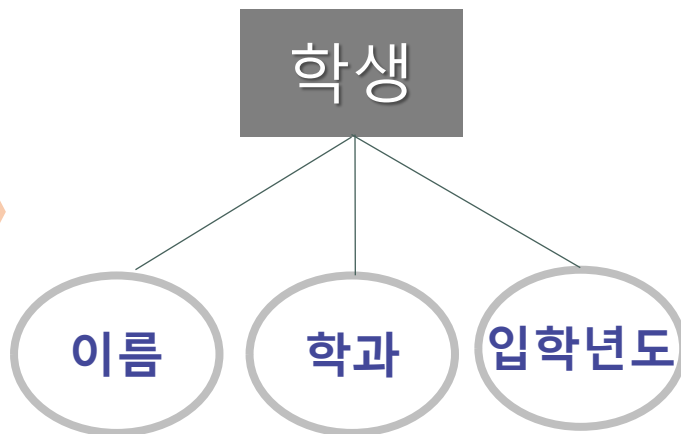
사물 또는 사건에 관한 특징이나 관점을 연관된 사람이나 그룹을 위하여 단순하고 명확하게 표현한 것

### 데이터 모델

현실 세계를 데이터베이스로 표현하는 과정에서 데이터베이스의 구조를 개념적/논리적으로 표현하기 위해 사용되는 도구

# 1 데이터 모델

## 3 데이터 모델의 예



학생	이름	학과	입학년도
	=====	=====	=====
	김하늘	컴퓨터	2019
	장동석	교육	2020
	한마음	복지	2021
	...		

현실세계 : 학생

데이터모델 : 학생개체

데이터베이스 : 학생 테이블



# 2 데이터 모델링

## 1 정의 #1

### 데이터 모델링(Data Modeling)

“정보시스템을 구축하기 위해,  
어떤 데이터가 존재하는지 또는 업무가  
필요로 하는 정보는 무엇인지를  
분석/표현하는 방법”



## 2 데이터 모델링

### 1 정의 #2

- ☑ 복잡한 실세계를 단순화하여 현실세계에 존재하는 개체들을 식별하고 이들 개체와 개체 사이의 관계를 정의하는 과정
- ☑ 고객으로부터 데이터에 대한 요구사항을 파악
- ☑ 프로젝트에 참여하는 분석자, 설계자, 개발자, 사용자 간의 효율적인 의사 소통을 위해 필수적인 과정
- ☑ 신규 또는 개선시스템 개발을 위한 기초가 됨

## 2 데이터 모델링

### 2 데이터 모델의 역할

- 1 시스템을 현재 또는 원하는 모습으로 **가시화**하도록 도와준다.
- 2 시스템의 구조와 행동을 **명세화**할 수 있게 한다.
- 3 시스템을 구축하는 **틀**을 제공한다.
- 4 결정된 것들을 **문서화**한다.
- 5 시스템을 바라보는 **다양한 관점**을 제공한다.
- 6 특정 목표에 따라 다양한 **상세 수준**을 제공한다.

## 2 데이터 모델링

### 3 데이터 모델링 단계 #1

#### 개념 데이터 모델링

핵심 개체를 추출하고 그들간의 관계를 정의(ER모델)

#### 논리 데이터 모델링

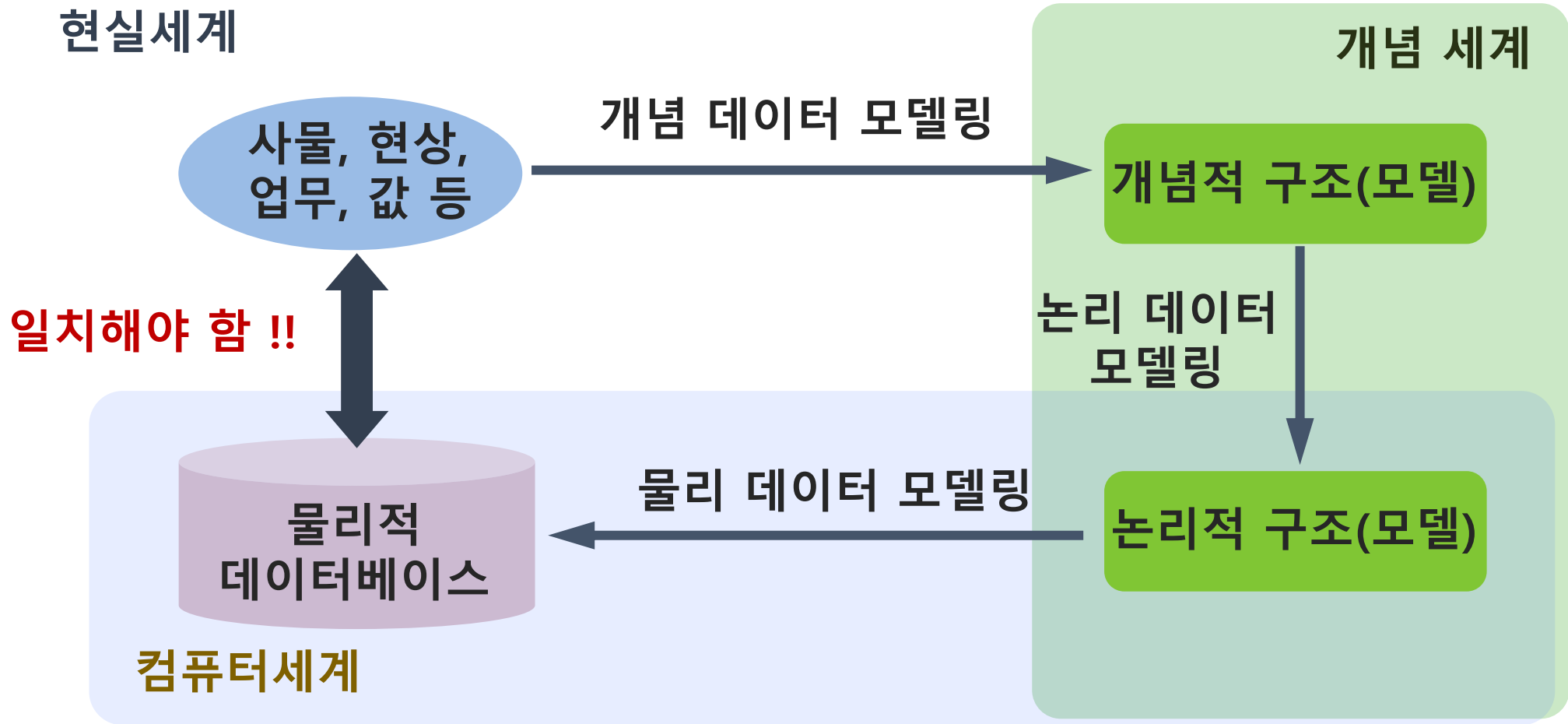
데이터베이스 유형에 맞추어 저장될 데이터의 구조(스키마) 작성

#### 물리 데이터 모델링

목표로 하는 DBMS의 특성 및 구현 환경 등을 고려한 데이터 상세구조 정의

## 2 데이터 모델링

### 3 데이터 모델링 단계 #2



## Chapter 02 데이터베이스 설계

# 1 데이터베이스 설계 개요

## 1 정의



### 데이터베이스 설계

- 사용자의 요구사항으로부터 그들에게 필요한 정보를 제공할 수 있도록 데이터베이스 구조를 개발하는 과정
- 개념적 데이터베이스 설계, 논리적 데이터베이스 설계, 그리고 물리적 데이터베이스 설계 과정을 포괄하는 개념

# 1 데이터베이스 설계 개요

## 2 데이터베이스 설계 시 고려사항

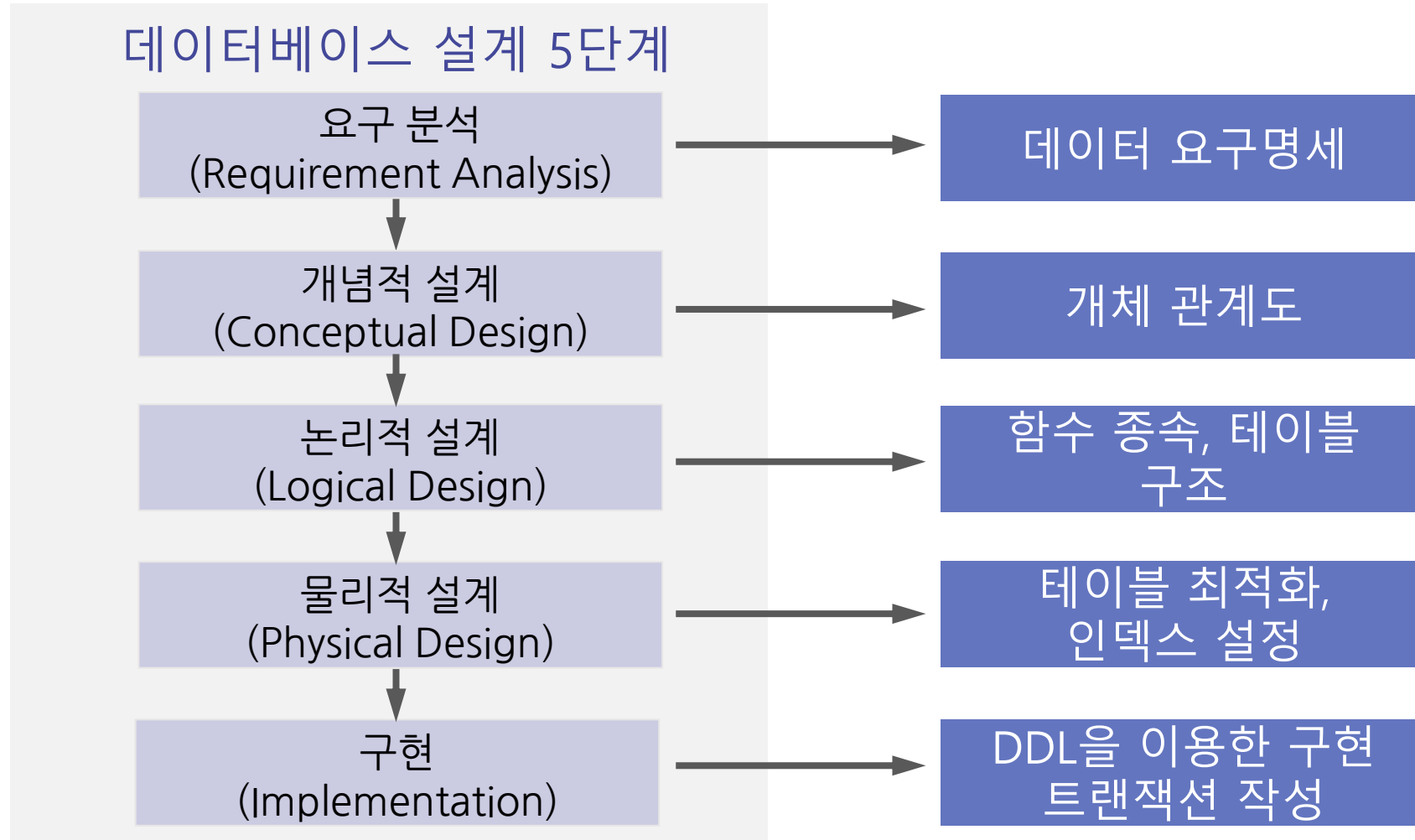
 현재 및 미래의 사용자 요구사항을 충족시킬 수 있도록 데이터 지향적 관점에서 설계해야 함

항목	설명
무결성	데이터가 만족해야 할 제약조건을 만족함
일관성	저장된 데이터들 사이에 모순성이 없어야 함
회복	시스템에 장애 발생 시 장애 발생 전의 일관된 상태로 복구가 가능해야 함
보안	불법 접근(데이터의 변경, 손실, 노출)에 대한 보호
효율성	응답시간, 저장 공간 최적화, 생산성 등을 고려함
확장성	응용 프로그램 추가, 데이터 증가, 변경 등을 수용할 수 있는 구조



# 1 데이터베이스 설계 개요

## 3 데이터베이스 설계 5단계



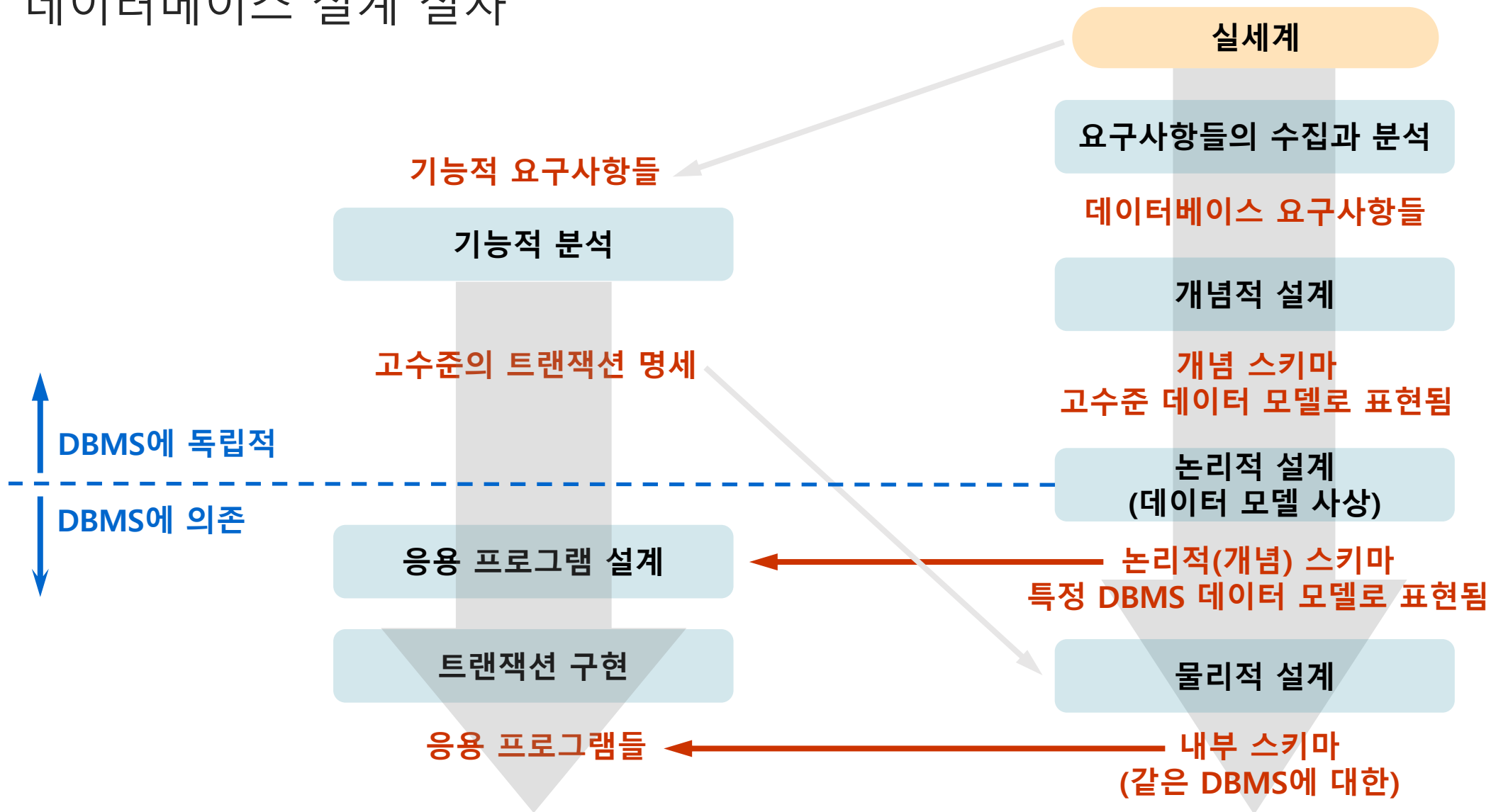
# 1 데이터베이스 설계 개요

## 4 데이터 모델링과 데이터베이스 설계

- ☑ 데이터 모델링이란 개념적, 논리적 수준에서 현실 세계의 상황을 컴퓨터가 이해할 수 있는 논리적으로 명확하게 정형적으로 표현하는 과정
- ☑ 개념/논리/물리 데이터모델링으로 구분
- ☑ 데이터베이스 설계는 이러한 데이터 모델링 과정을 포함
- ☑ 최적화된 데이터베이스를 구현하기 위한 단계적 실행 절차라고 정의
- ☑ 거의 비슷한 의미, 데이터 모델링은 개념적/논리적 수준을 말하는 것이 일반적
- ☑ 데이터베이스 설계는 시스템 구현에 좀더 가까운 개념

# 1 데이터베이스 설계 개요

## 5 데이터베이스 설계 절차



## 2 데이터베이스 설계 5단계

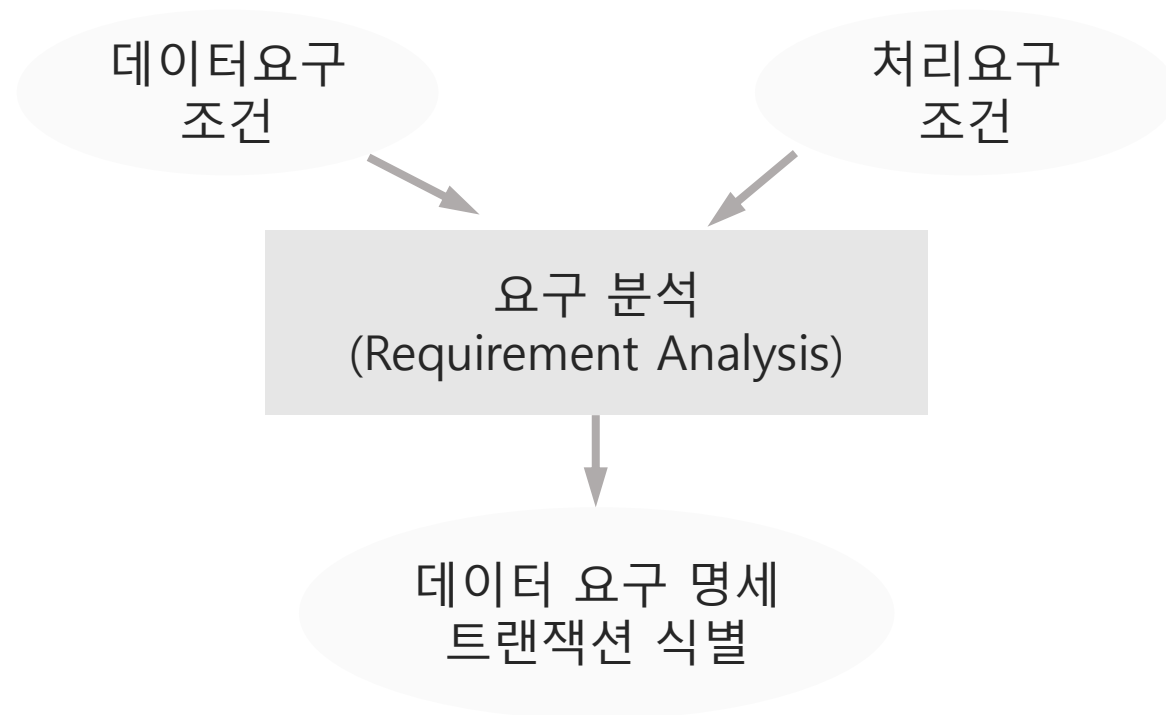
### 1

#### 요구 분석 단계 #1



사용자의 요구조건을 수집하고 분석해서 공식적인 요구조건 명세를 생성

- 개체, 속성, 관계, 제약조건 등과 같은 정적인 구조에 대한 요구조건 명세 생성
- 트랜잭션의 유형, 트랜잭션의 실행 빈도와 같은 동적인 요구조건 명세 생성
- 기관의 경영 목표 및 정책, 규정과 같은 제약조건이 포함됨



\* 반드시 문서화가 되어야 함

## 2 데이터베이스 설계 5단계

### 1

#### 요구 분석 단계 #1



#### 예) 학교 데이터베이스 시스템의 데이터 요구 명세

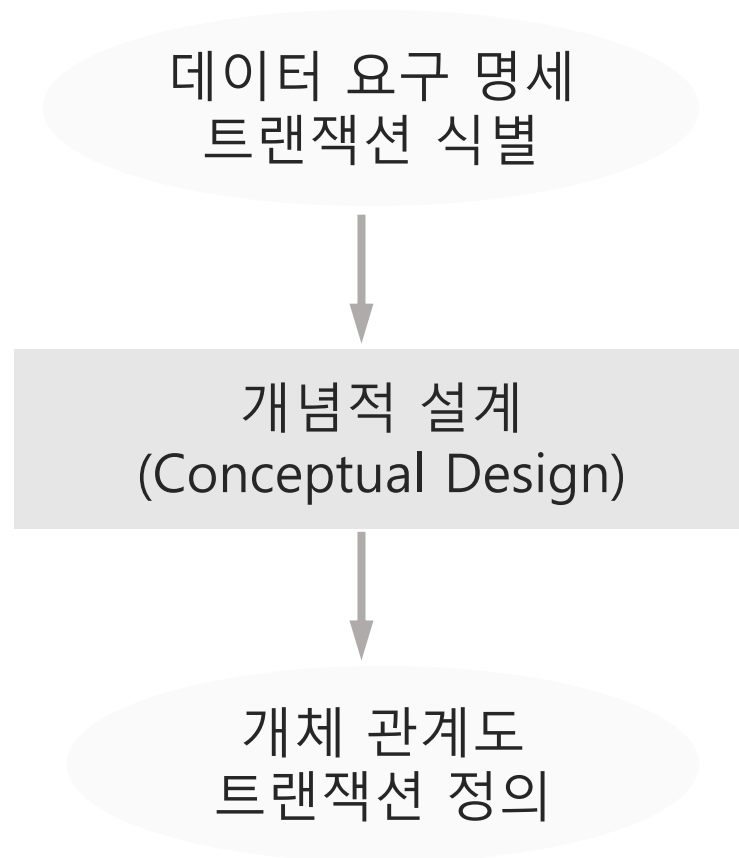
- 1) 대학교에는 각 과목에 대하여 수강한 학생, 강의가 이루어지는 시간 (여러 개의 값) 및 장소 등의 데이터가 유지된다.
- 2) 한 명의 교수는 여러 개의 과목을 강의할 수 있으며, 각 과목과 학생 간에는 학점이 부여 된다.
- 3) 교수에 관해서는 교수번호, 이름, 성별 등의 정보가 유지되어야 한다.
- 4) 학생을 위해서는 학번, 이름, 학과, 연락처 등의 정보가 유지 되어야 한다.
- 5) 과목 정보로는 과목번호, 과목명, 담당교수

## 2 데이터베이스 설계 5단계

### 2 개념적 설계 단계 #1

#### 개체 간의 관계를 정확히 표현하는 단계

- 요구분석 명세로부터 개체, 속성, 관계를 식별해서 결정
- 특히, 개념적 설계에서는 개체와 개체 간의 연관성(관계:relationship)에 중점을 둠
- 개념적 데이터 모델로 기술  
(개체관계도 : Entity-Relationship Diagram)
- 트랜잭션을 정의한다.


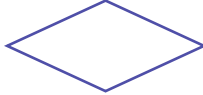



## 2 데이터베이스 설계 5단계

### 2 개념적 설계 단계 #2

#### 개체-관계도(Entity-Relationship Diagram)

- 개체, 관계, 속성으로 구성
- 개념적 데이터 모델을 위한 도구

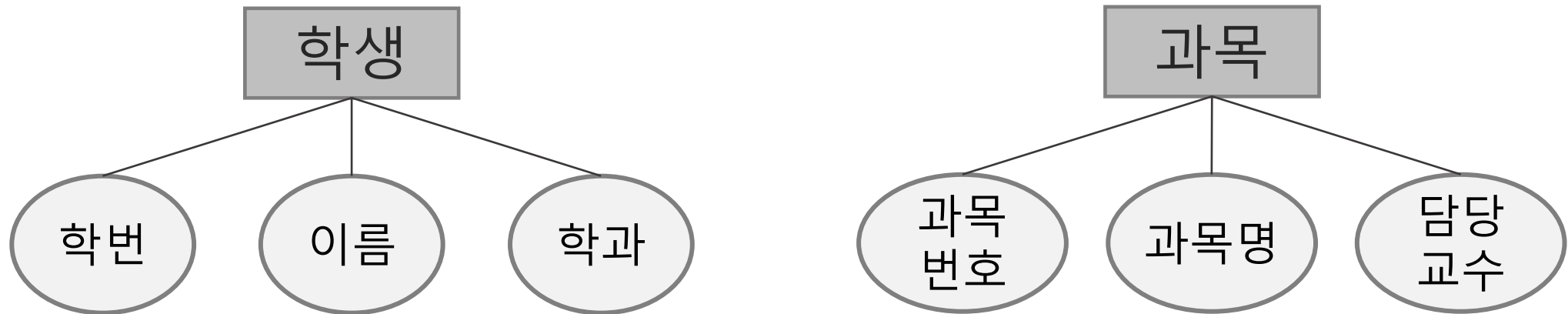
구성 요소	설명	표현 기호(Symbol)
개체	데이터베이스 하고자 하는 사건, 사물	
관계성	개체간 의 연관성	
속성	개체, 관계의 성질을 나타내는 더 이상 쪼갤 수 없는 정보의 단위	

## 2 데이터베이스 설계 5단계

### 2 개념적 설계 단계 #3

 개체(entity)와 속성(애틀리뷰트: attribute)

- 개체란 데이터베이스에 표현하려고 하는 유형, 무형의 객체로써 서로 구별되는 것
- 하나의 개체는 하나 이상의 속성으로 구성되며, 각 속성은 그 개체의 특성이나 상태를 기술



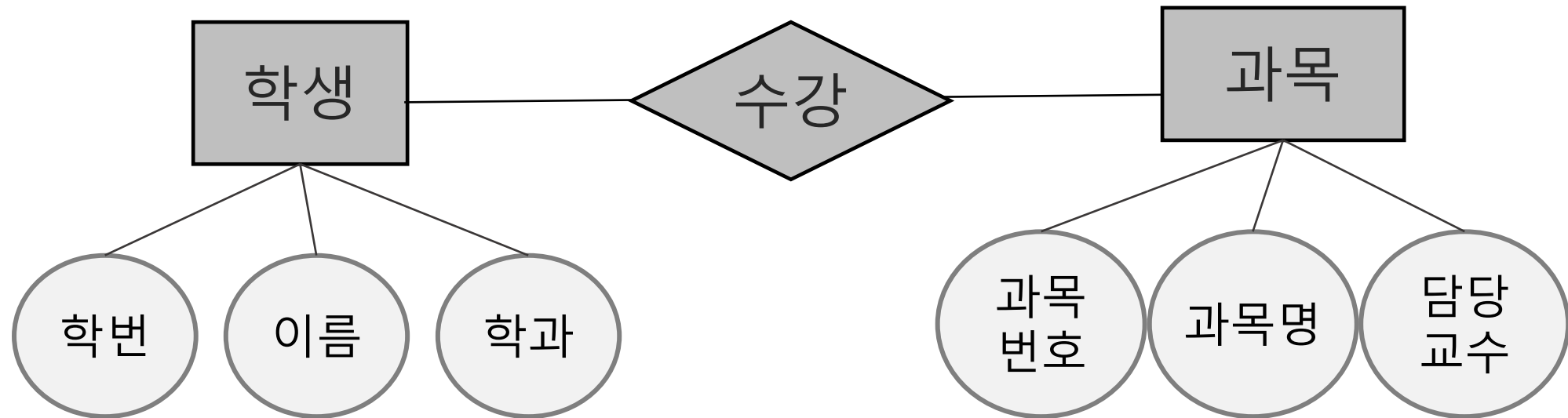


## 2 데이터베이스 설계 5단계

### 2 개념적 설계 단계 #4

#### 관계(relationship)

- 개체와 개체 사이의 관련성을 나타내는 정보
- 개체와 함께 데이터베이스 내에 저장되어야 할 대상

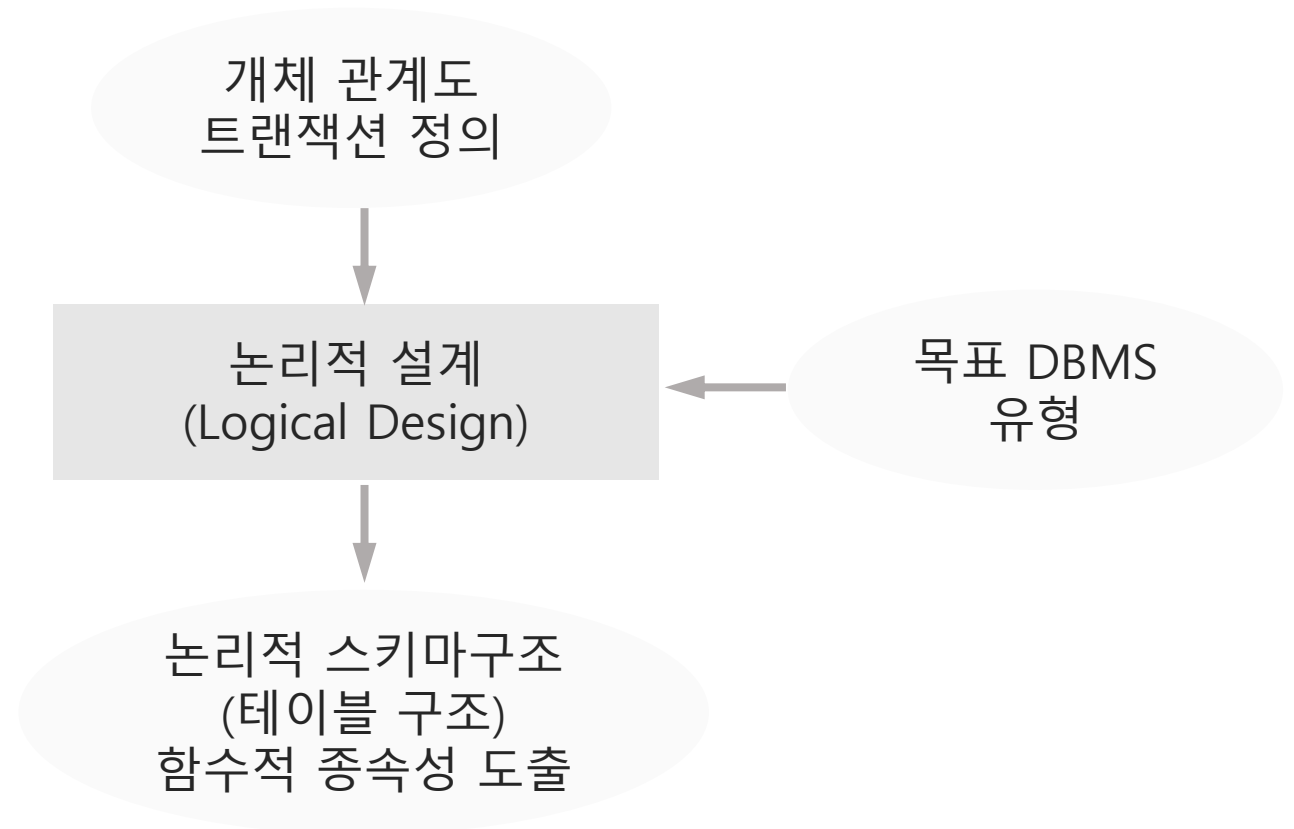


## 2 데이터베이스 설계 5단계

### 3 논리적 설계 단계 #1

 개념적 구조로부터 특정 목표 DBMS가 처리할 수 있는 스키마를 생성

- 관계형, 계층형, 네트워크형, 객체지향형, 객체관계형 등
- 무결성이나 일관성 제약조건도 검토
- 관계형 데이터베이스 모델
  - 함수적 종속성 분석과 테이블 정규화
  - DBMS의 특성을 많이 고려



## 2 데이터베이스 설계 5단계

### 3 논리적 설계 단계 #2

#### 트랜잭션(Transaction)

- 트랜잭션(Transaction)은 하나의 논리적 기능을 수행하기 위한 작업의 단위
- 데이터베이스의 일관성을 유지하기 위한 수단



#### 예) 계좌 이체

- 은행의 A 계좌에서 1000만원을 인출해서, B계좌로 입금

## 2 데이터베이스 설계 5단계

### 3 논리적 설계 단계 #3

 트랜잭션(Transaction)이 가져야할 특성(ACID 성질)

**원자성**  
(atomicity)

자기의 연산을 전부 또는 전무(All or Nothing) 실행

**일관성**  
(consistency)

언제나 일관성 있는 데이터베이스의 상태 유지

**격리성**  
(isolation)

트랜잭션 실행 중에 다른 트랜잭션이 접근할 수 없다.

**영속성**  
(durability)

완료된 결과의 영속성을 보장

## 2 데이터베이스 설계 5단계

4

### 논리적 설계 단계 #4



### 논리적 스키마구조

학생	<u>학번</u>	이름	학과
----	-----------	----	----

과목	<u>과목번호</u>	과목명	담당교수
----	-------------	-----	------

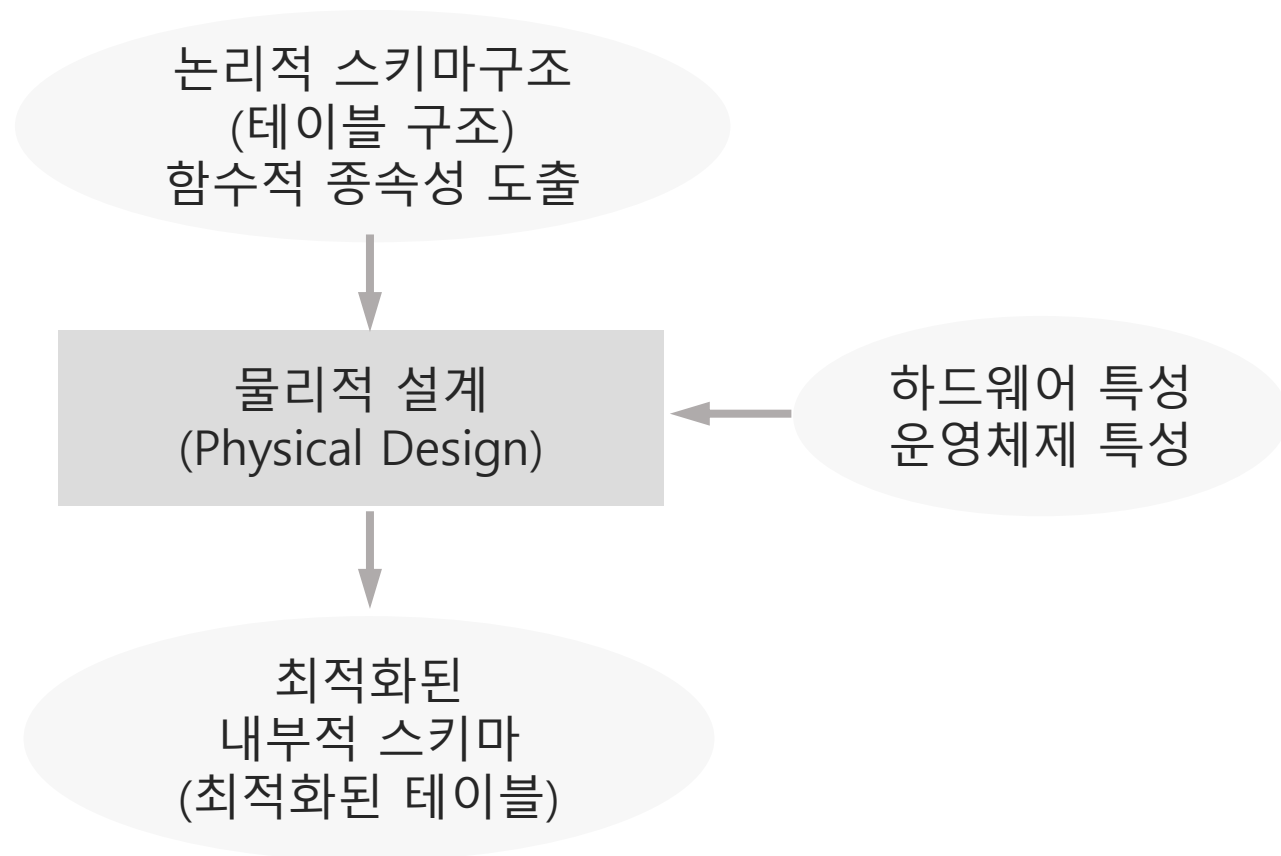
수강	<u>학번</u>	<u>과목번호</u>
----	-----------	-------------

## 2 데이터베이스 설계 5단계

### 5 물리적 설계 단계 #1

 효율적이고 구현 가능한 물리적 데이터베이스 구조를 설계

- 데이터의 저장 구조나 접근 경로에 대해서 DBMS가 지원하는 방법 중에서 선택
- 응답시간, 저장공간, 트랜잭션 처리도 고려
- 레코드의 양식, 순서, 접근경로, 저장공간 할당
- 인덱스, 역정규화에 대한 결정



## 2 데이터베이스 설계 5단계

### 5 물리적 설계 단계 #2

#### 테이블 명세서

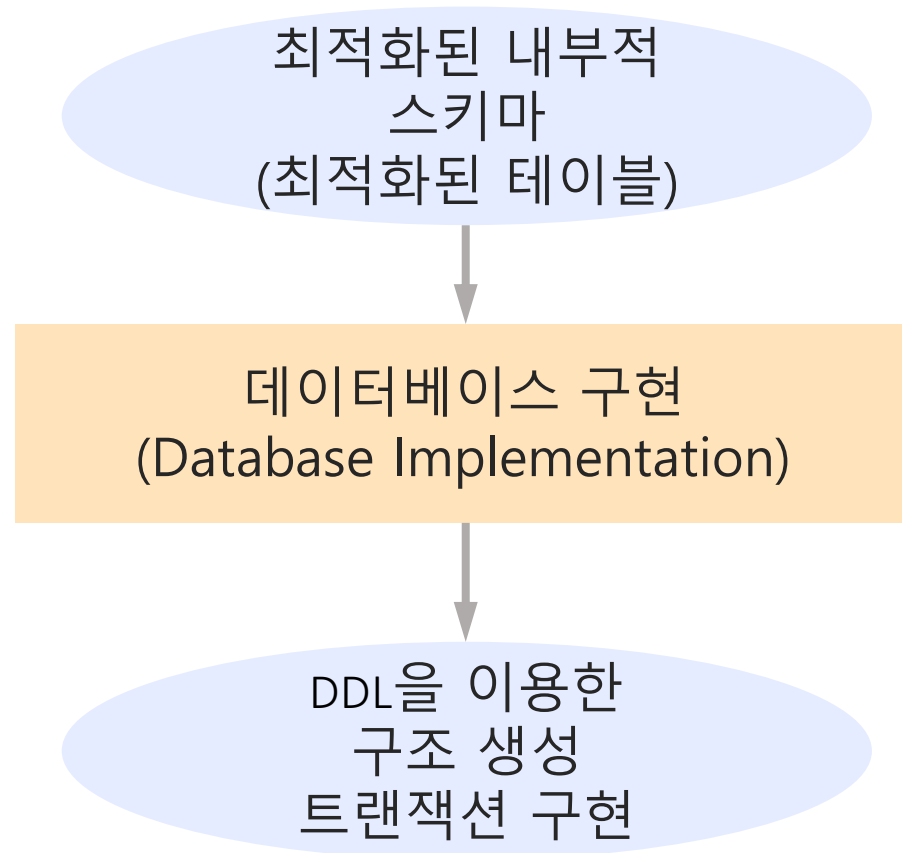
테이블 명	STUDENT		
컬럼명	컬럼타입	크기	비고
st_number	고정문자열	5	학번, 기본키
st_name	가변문자열	20	이름, NOT NULL
st_dept	가변문자열	20	학과

## 2 데이터베이스 설계 5단계

### 6 데이터베이스 구현 단계 #1

#### DDL을 이용하여 데이터베이스 스키마 구조 생성

- 목표 DBMS의 DDL로 기술하고 실행
- 트랜잭션으로 구현(저장프로시저, 프로그램 등)
- 초기데이터 입력
- 운영 데이터베이스로 완성





## 2 데이터베이스 설계 5단계

### 6 데이터베이스 구현 단계 #2

#### SQL 문

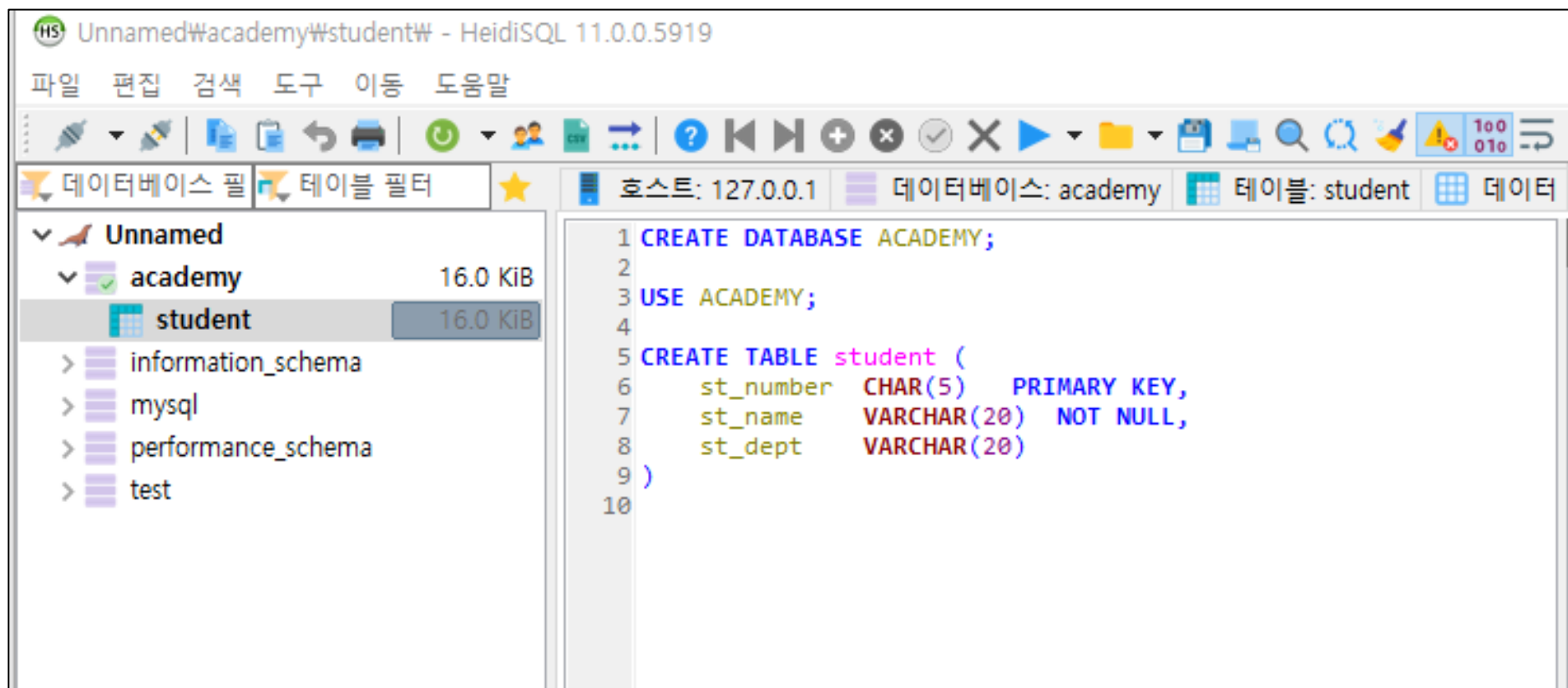
**// 학생 테이블 생성을 위한 SQL 문**

```
CREATE TABLE student (  
    st_number    CHAR(5)    PRIMARY KEY,  
    st_name      VARCHAR(20) NOT NULL,  
    st_dept      VARCHAR(20)  
)
```

## 2 데이터베이스 설계 5단계

### 6 데이터베이스 구현 단계 #3

#### 생성된 테이블



# 학습 정리



## 데이터 모델링

- 데이터 모델

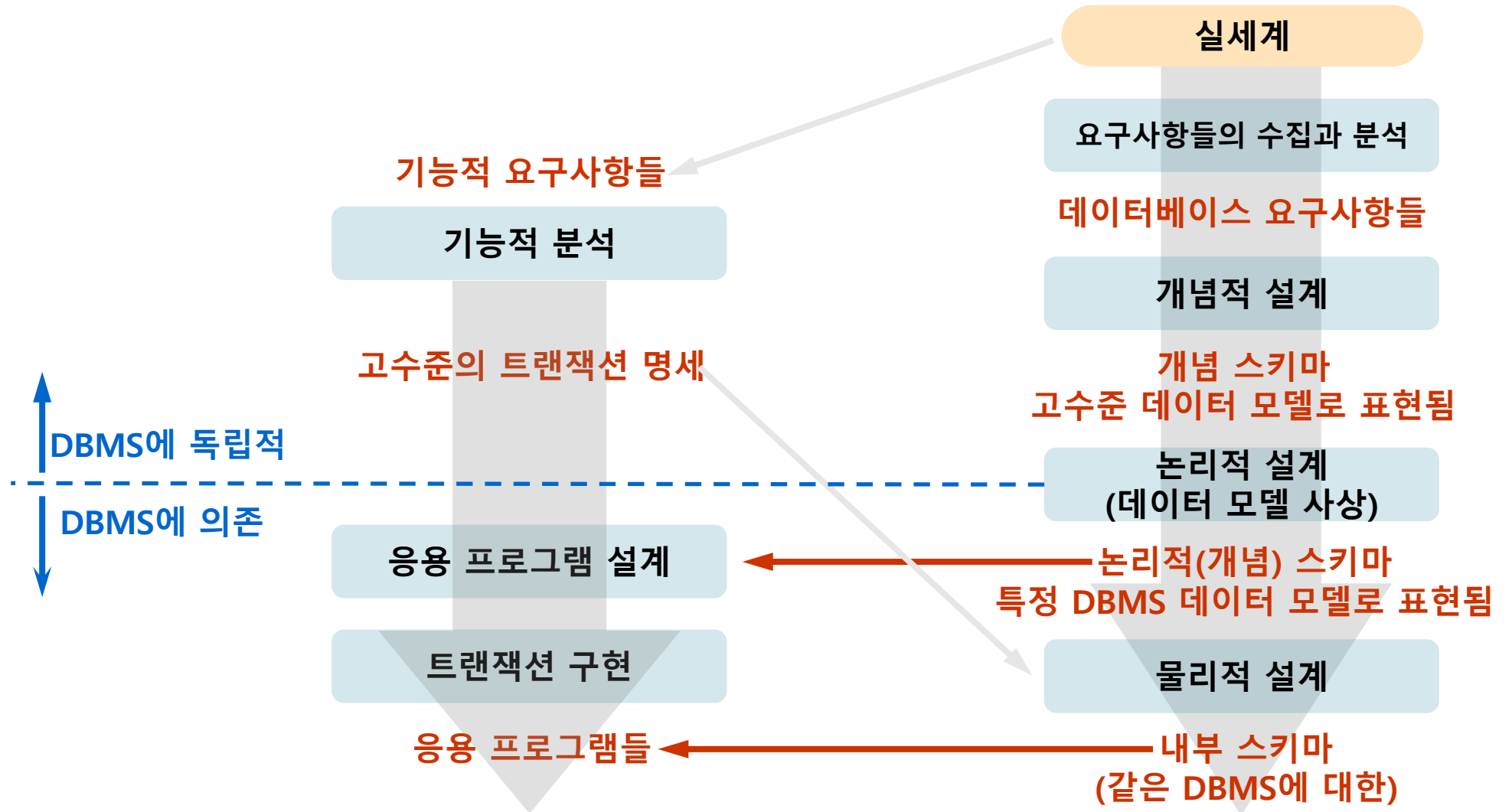
- 현실 세계를 데이터베이스로 표현하는 과정에서 데이터베이스의 구조를 개념적/논리적으로 표현하기 위해 사용되는 도구

- 데이터 모델링

- 정보시스템을 구축하기 위해, 어떤 데이터가 존재하는지 또는 업무가 필요로 하는 정보는 무엇인지를 분석/표현하는 방법

# 학습 정리

## 📄 데이터베이스 설계 절차



# 참고 문헌



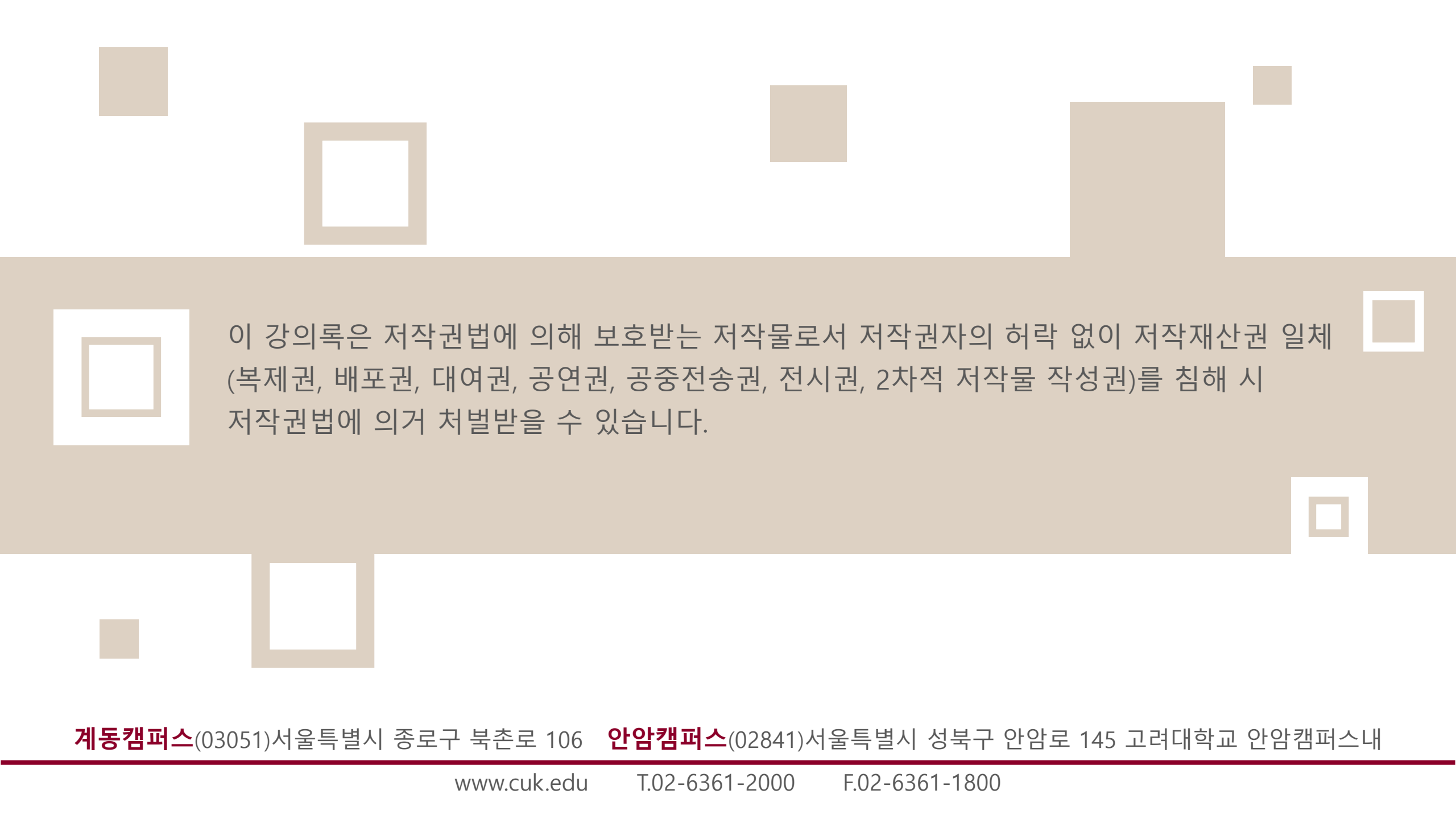
데이터베이스 시스템 7판,  
Ramez Elmasri , Shamkant B. Navathe 지음,  
황규영 등 옮김, 홍릉과학출판사,  
2018년 8월

---



[www.wikipedia.org](http://www.wikipedia.org)

---



이 강의록은 저작권법에 의해 보호받는 저작물로서 저작권자의 허락 없이 저작권재산권 일체 (복제권, 배포권, 대여권, 공연권, 공중전송권, 전시권, 2차적 저작물 작성권)를 침해 시 저작권법에 의거 처벌받을 수 있습니다.