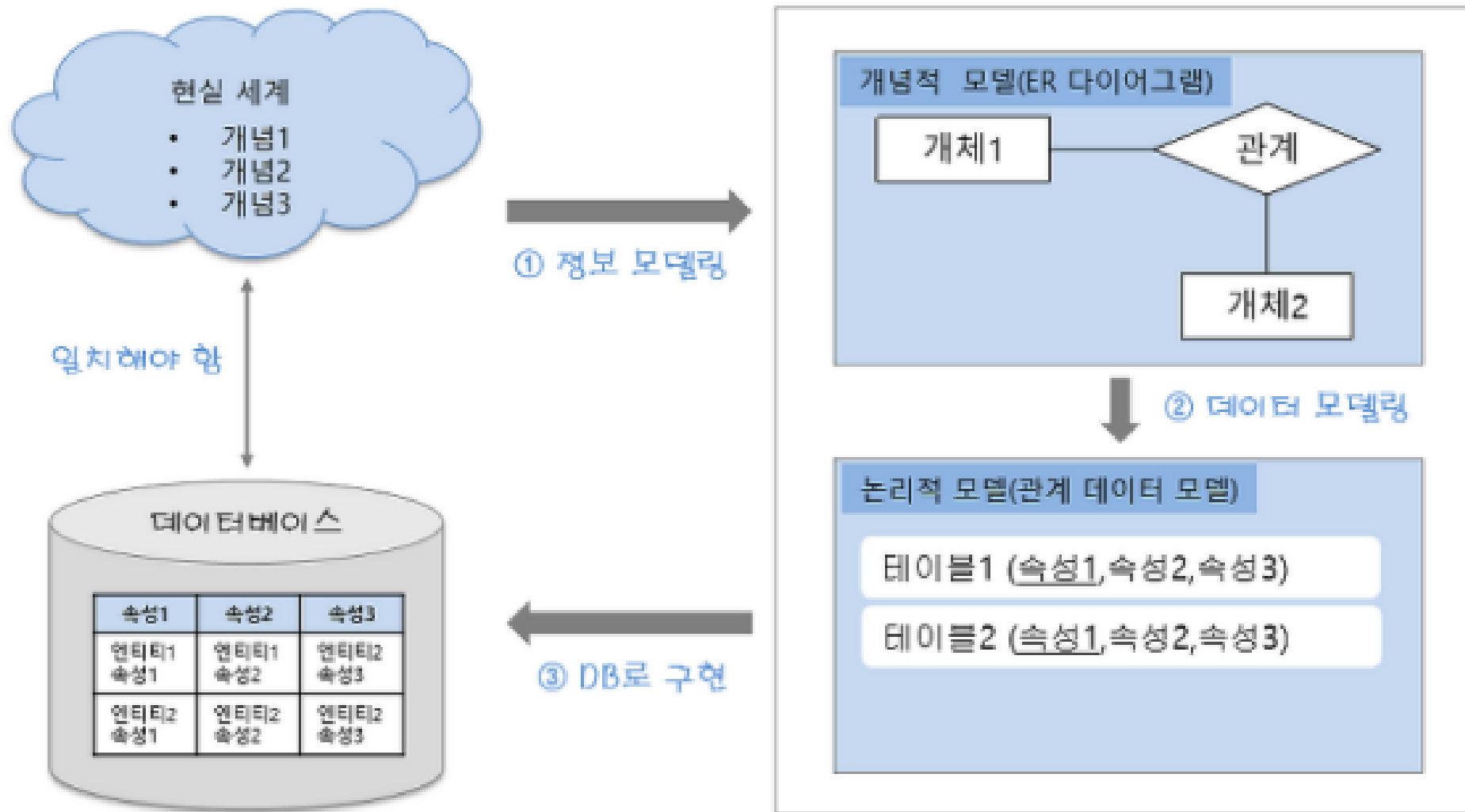
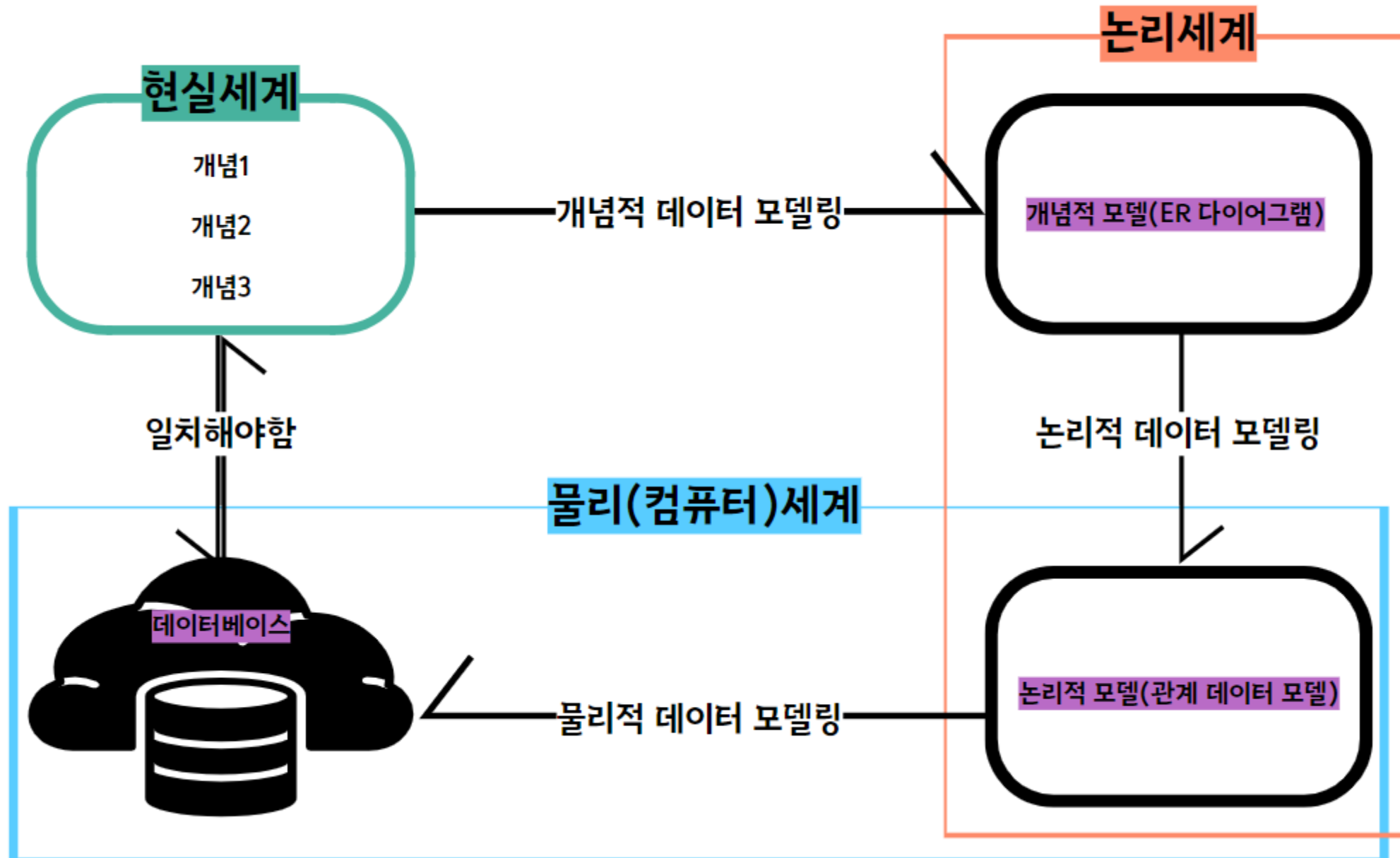


**CODINGO** x **posco**

**K-Digital Training** 스마트 팩토리 3기

# 데이터 모델





# 데이터 모델

- 데이터베이스 시스템에서 **데이터를 저장하는 이론적인 방법**
- 데이터베이스에 데이터가 어떻게 구조화되어 저장되는지 결정
- 데이터 모델 종류
  - 계층형 데이터 모델 ( Hierarchical Data Model )
  - 네트워크형 데이터 모델 ( Network Data Model )
  - **관계형 데이터 모델 (relational data model)**
  - 객체지향형 데이터 모델 ( Object Oriented Data Model )

# 계층형 데이터 모델

- 계층형 데이터 모델은 트리 구조를 활용
- 데이터 관련성을 계층별로 관계를 정의하고 데이터를 관리
  - 부모 자식 간의 관계
- 부모는 여러 자식을 둘 수 있지만 자식은 여러 부모를 둘 수 없음

# 계층형 데이터 모델



계층형 데이터 모델

# 계층형 데이터 모델

## 단점

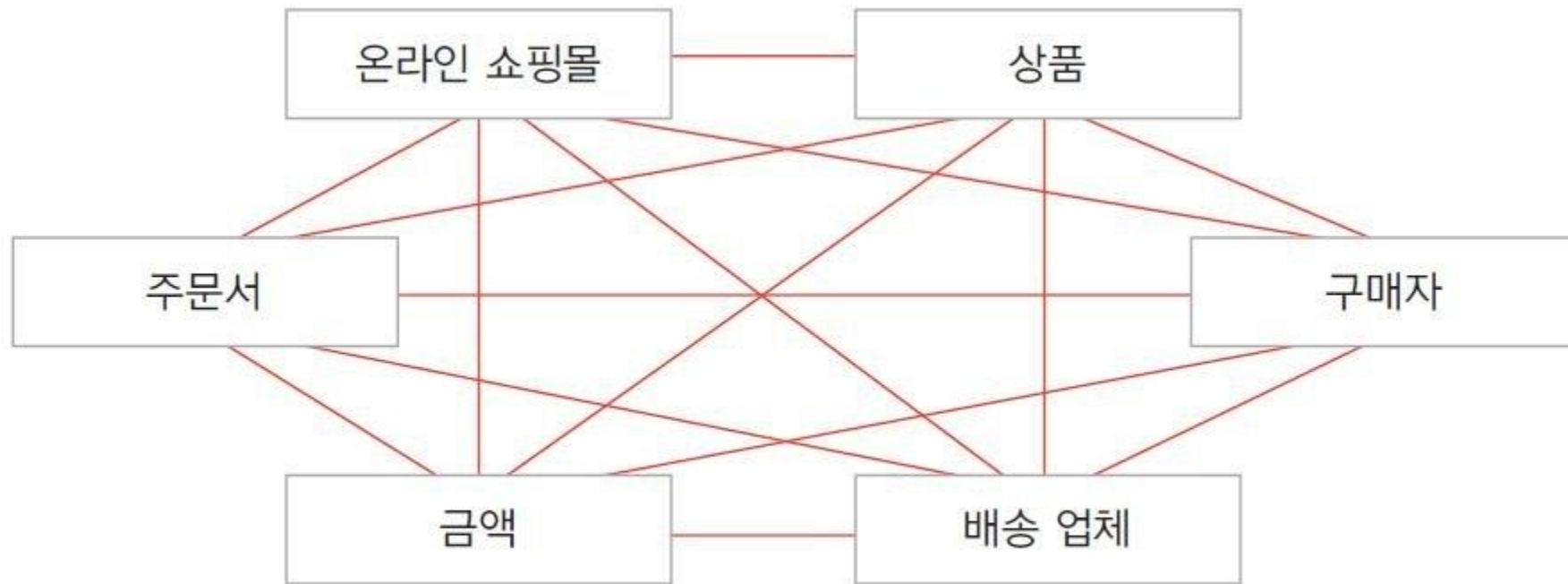
- 계층 구조의 제약으로 인해 복잡한 관계를 표현하기 어려움
- 데이터의 확장이 어려움
- 데이터의 유연성이 제한 됨



# 네트워크형 데이터 모델

- 그래프 구조를 기반으로 함
- 개체 간 관계를 그래프 구조로 연결함
- 자식 개체가 여러 부모 개체를 가질 수 있음

# 네트워크 데이터 모델



네트워크형 데이터 모델

# 네트워크형 데이터 모델

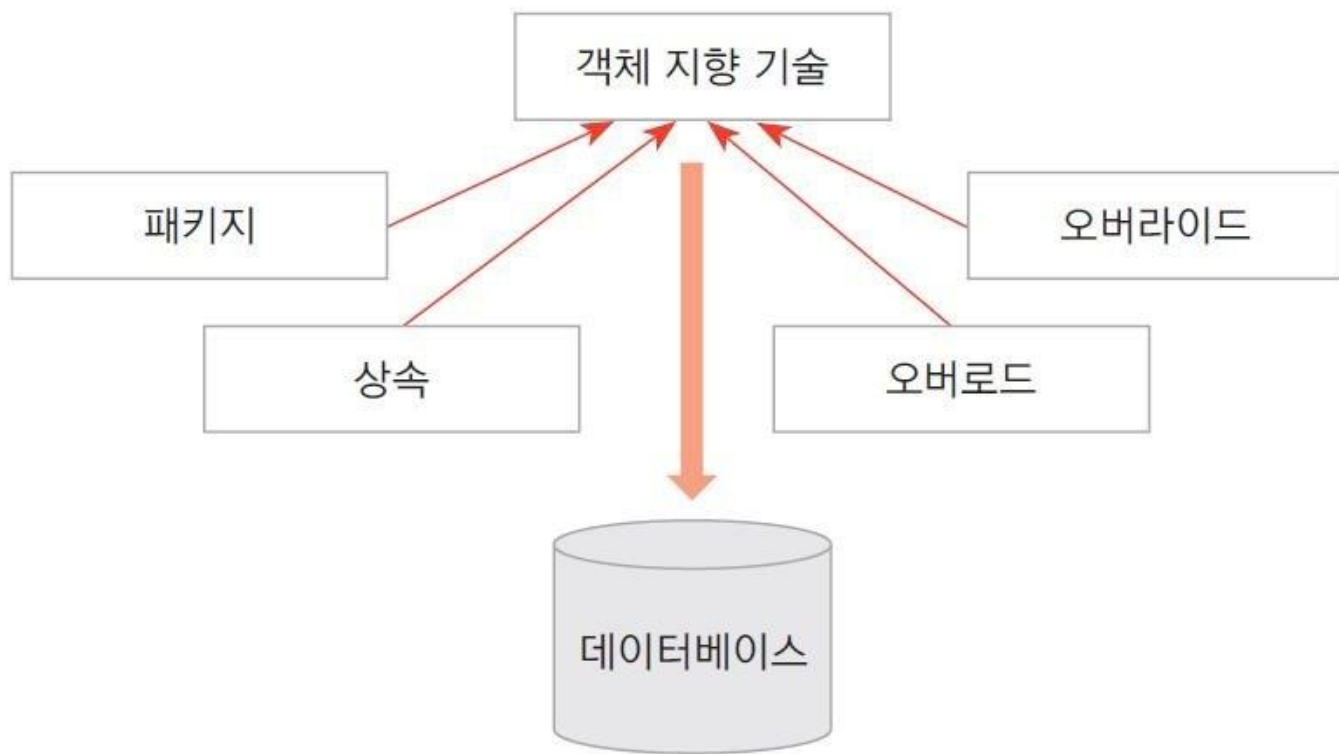
## 단점

- 접근이 복잡함, 성능 저하 우려
- 데이터의 일관성을 유지하기 어려움
- 데이터의 중복이 발생할 수 있음

# 객체 지향형 모델

- 1980년대 후반에 등장
- 객체 지향 프로그래밍에서 사용하는 객체 개념을 기반으로 함
- 객체를 중심으로 데이터를 저장하므로 응용 프로그램과의 호환성이 높음
- 계속해서 발전 중

# 객체 지향형 모델



객체 지향형 데이터 모델

# 객체 지향형 데이터 모델

## 단점

- 객체 간의 관계를 표현하는 것이 복잡함
- 저장공간 낭비
- 수행 속도 및 성능의 저하 유발
- 비용이 증가

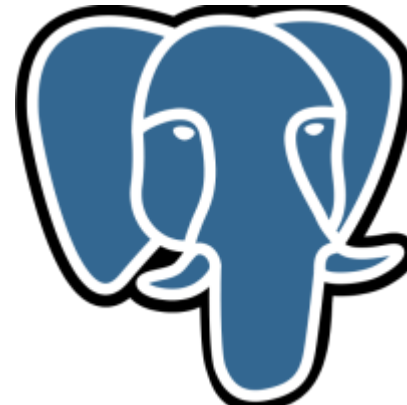
# 관계 데이터 모델

- 1970년대 제안
- 수학의 집합이론에 근거하여 탄탄한 이론적 토대
- 데이터를 관계형 테이블로 표현하는 데이터 모델링 기법
- 관계 데이터 모델이 적용된 SQL 언어는 비절차적인 언어로 원하는 데이터를 쉽게 표현

# 관계 데이터 모델 DMBS

- Oracle
- SQL Server
- DB2
- MySQL
- PostgreSQL
- SQLite

ORACLE®





# 관계 데이터 모델 개념

- 릴레이션
- 릴레이션 스키마와 인스턴스
- 릴레이션 특징
- 관계 데이터 모델

# 릴레이션 (relation)

- 행과 열로 구성된 테이블
- ⚠ 관계라고 하지 않음



Relation

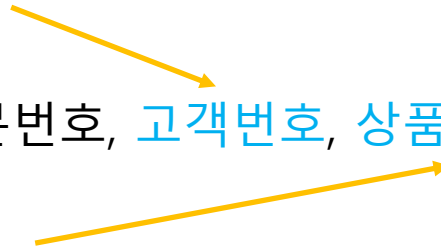
# 관계 (relationship)

- 릴레이션 내에서 생성되는 관계
- 릴레이션 간에 생성되는 관계
- ex. "고객"과 "주문" 릴레이션의 관계, "상품"과 "주문" 릴레이션의 관계

고객 (고객번호, 고객명, 전화번호)

주문 (주문번호, 고객번호, 상품번호, 구매일)

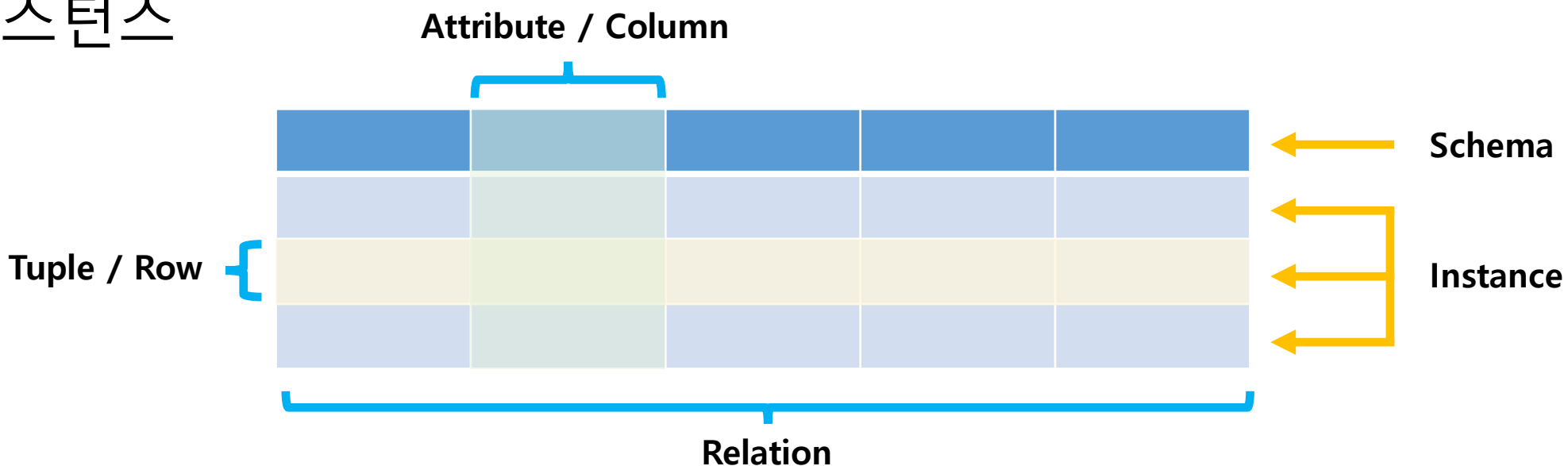
상품 (상품번호, 상품명, 가격)





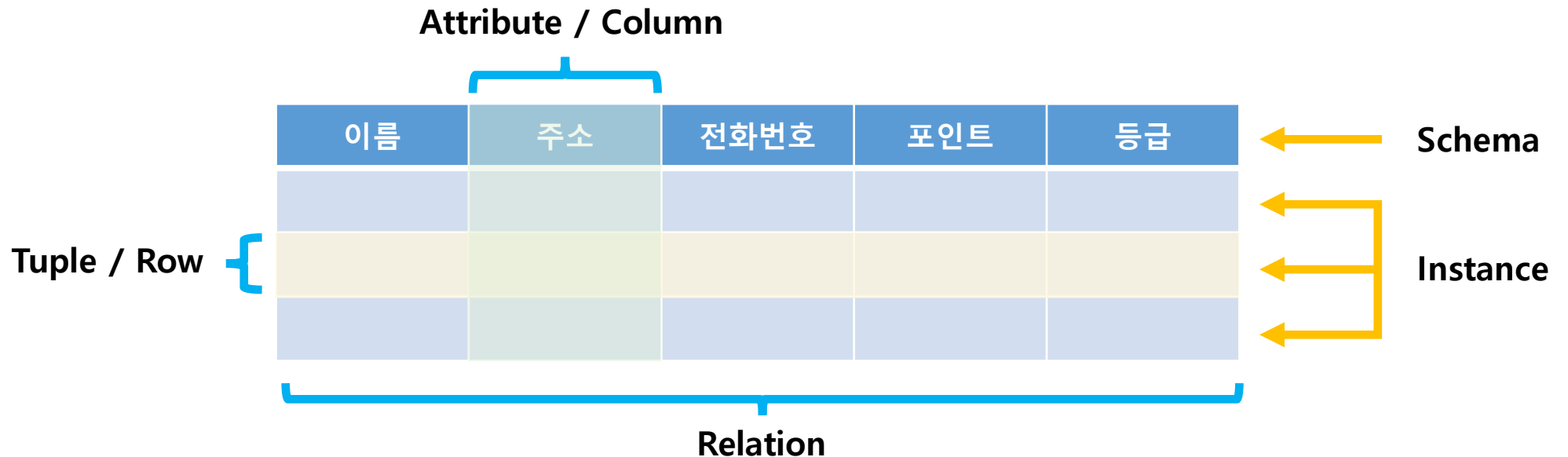
# 릴레이션 스키마와 인스턴스

- 튜플, 행
- 속성, 열
- 스키마
- 인스턴스



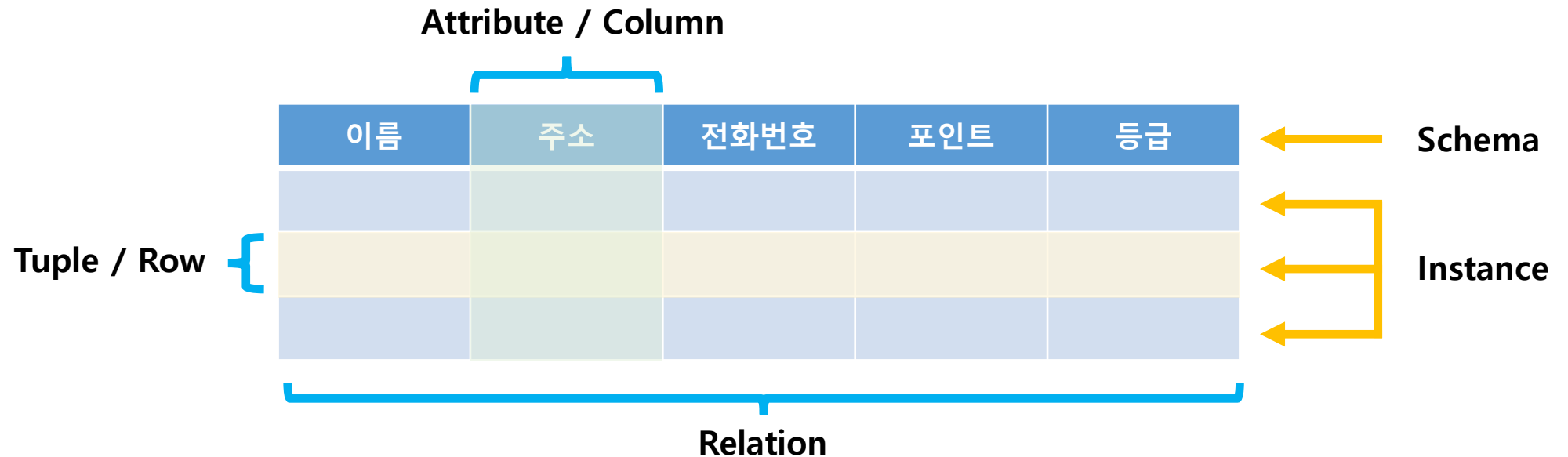
# 릴레이션 스키마

- 릴레이션에 어떤 정보가 담길지 정의
- 표기법: 릴레이션 이름(속성A, 속성B, ...)
- ex. 고객(이름, 주소, 전화번호, 포인트, 등급)



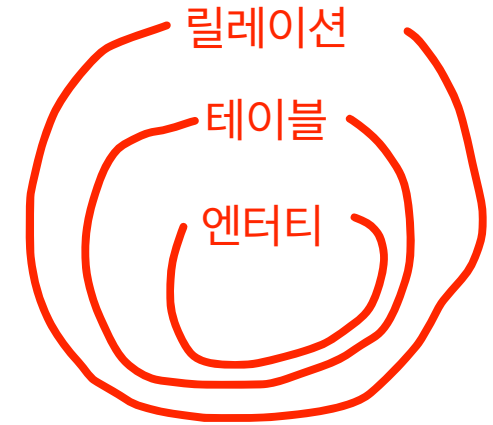
# 릴레이션 인스턴스

- 릴레이션의 스키마에 실제로 저장된 데이터



# 속성 (Attribute)

- Entity의 특성이나 속성을 의미
- 쉽게 말하면 데이터를 가져와서 알고 싶은 것!
- ex. 한 마트의 고객 개체에서 알고 싶은 데이터



Attribute / Column					← Schema  ← Instance ← Instance ← Instance
이름	주소	전화번호	포인트	등급	
Relation					

→ DB의 설계 단계에서는 엔터티(Entity),  
→ DBMS로 구현되는 단계에서는 테이블(Table),  
엔터티 < 테이블 < 릴레이션 → 개념 단계에서 엔터티간 연관관계를 릴레이션(Relation)

# Entity 퀴즈!!

다음 중 아래 시나리오에서 Entity로 적합한 것은?

대한민국 축구 국가대표팀은 여러 명의 선수가 존재하고 각 선수에 대한 이름, 주소 등을 관리해야 한다.

- (1) 대한민국 축구 국가대표팀
- (2) 선수
- (3) 이름
- (4) 주소





# 릴레이션 특징

1. 속성은 **단일 값**을 가짐
2. 속성은 **서로 다른 이름**을 가짐
3. 한 속성의 값은 모두 **같은 도메인** 값을 가짐 문자열이면 문자열, 숫자면 숫자
4. 속성의 **순서**는 상관 없음
5. 릴레이션 내의 **중복된 튜플** 허용 안됨
6. 튜플 **순서** 상관 없음

**도메인:** 릴레이션의 각 속성이 가질 수 있는 값의 집합

# 퀴즈 1

관계 데이터 모델의 릴레이션에 대한 설명 중 옳지 않은 것?

- (1) 릴레이션은 릴레이션 스키마와 인스턴스로 구성 됨
- (2) 릴레이션 스키마를 외연이라고 함
- (3) 릴레이션 스키마는 정적인 성질을 가짐
- (4) 릴레이션 인스턴스는 동적인 성질을 가짐

## 퀴즈 2

릴레이션 특징으로 알맞은 것은?

- (1) 중복된 튜플 존재
- (2) 튜플간 순서 정의됨
- (3) 속성간 순서 정의됨
- (4) 모든 속성 값은 원자값임

## 퀴즈 3

하나의 속성이 가질 수 있는 값을 총칭하여 무엇이라 하는가?

- (1) 튜플
- (2) 릴레이션
- (3) 도메인
- (4) 엔티티

# 관계 데이터 모델

- 관계 데이터 모델은 데이터를 2차원 테이블 형태인 릴레이션으로 표현
- ex. 고객 정보는 고객 릴레이션, 주문 정보는 주문 릴레이션, 상품 정보는 상품 릴레이션

