

데이터 모델링 이해

1. 데이터 모델링 개요

2. 엔티티

3. 속성

4. 관계

5. 식별자

데이터 모델링의 정의

- Webster 사전
 - 가설적 또는 일정 양식에 맞춘 표현(a hypothetical or stylized representation)
 - 어떤 것에 대한 예비표현으로 그로부터 최종대상이 구축되도록 하는 계획으로서 기여하는 것
- 복잡한 '현실세계'를 단순화시켜 표현하는 것
- 모델이란 ? 사물 또는 사건에 관한 양상(Aspect)이나 관점(Perspective)을 연관된 사람이나 그룹을 위하여 명확하게 하는 것
- 모델이란 현실 세계의 추상화된 반영

데이터 모델링의 3가지 핵심 개념

1. 추상화 (Abstraction)

- 복잡한 현실세계를 일정한 형식에 맞추어 표현
- 중요한 요소만 선별하여 **본질적인 구조**로 표현

2. 단순화 (Simplification)

- 복잡한 현실을 **약속된 규약과 표기법**으로 간단히 표현
- **불필요한 요소** 제거, 이해와 설계 용이성 향상

3. 명확화 (Clarification)

- **애매모호함 제거**, 현실을 정확하게 기술
- **누구나 이해할 수 있는 구조**로 표현

모델링의 세가지 관점

데이터관점

업무가 어떤 데이터와 관련이 있는지 또는 데이터간의 관계는 무엇인지에 대해서 모델링하는 방법(What, Data)

프로세스관점

업무가 실제하고 있는 일은 무엇인지 또는 무엇을 해야 하는지를 모델링하는 방법(How, Process)

데이터와 프로세스의 상관관점

업무가 처리하는 일의 방법에 따라 데이터는 어떻게 영향을 받고 있는지 모델링하는 방법(Interaction)으로 설명됨.

데이터 모델링의 중요성

파급효과(Leverage)

- 시스템 구축의 완성시점에서의 데이터모델의 변경은 프로젝트의 위험요소

복잡한 정보 요구사항의 간결한 표현(Conciseness)

- 구축할 시스템의 정보 요구사항과 한계를 가장 명확하고 간결하게 표현하는 도구

데이터 품질(Data Quality)

- 부정확한 데이터는 소중한 비즈니스의 기회를 상실

데이터 모델링의 유의점

중복(Duplication)

- 여러 장소에 같은 정보를 저장하지 않도록 할 것

비유연성(Inflexibility)

- 사소한 업무변화에 데이터 모델이 수시로 변경되지 않도록 할 것
- 데이터와 프로세스의 분리
- 작은 변화가 애플리케이션과 데이터베이스에 중대한 변화를 일으키지 않도록 할 것

비 일관성(Inconsistency)

- 데이터와 데이터간 상호 연관 관계에 대한 명확한 정의
- 예) 신용 상태에 대한 갱신 없이 고객의 납부 이력정보를 갱신하는 것

데이터 모델링의 3단계

1. 개념적 데이터 모델링 (Conceptual Modeling)

- 예시: “고객이 상품을 주문한다”
- 업무 중심적, 전사적 관점의 추상적 모델
- 포괄적이고 추상화 수준이 높은 모델링

2. 논리적 데이터 모델링 (Logical Modeling)

- 예시: 고객(Customer), 상품(Product), 주문(Order)
- 업무 흐름과 데이터 구조 중심
- **Key, 속성, 관계** 등을 명확하게 표현
- DBMS에 독립적, **재사용성**이 높음

3. 물리적 데이터 모델링 (Physical Modeling)

- 실제 DB 구현 목적의 모델링
- 성능, 저장 공간 등 물리적 특성 고려
- 특정 DBMS에 맞춰 테이블, 인덱스, 파티션 등 설계

프로젝트 생명주기(Life Cycle)에서 데이터 모델링

- 프로젝트 생명주기



- 정보전략계획 단계 : 개념적 데이터 모델링
- 분석 단계 : 논리적 데이터 모델링
- 설계 단계 : 물리적 데이터 모델링

데이터베이스 3단계 구조

외부스키마(External Schema)-사용자관점

- 사용자가 처리하고자 하는 데이터 유형, 관점, 방법에 따라 다른 스키마구조
- DB의 개개 사용자나 응용프로그래머가 접근하는 DB정의

개념스키마(Conceptual Schema)-통합관점

- 모든 사용자 관점을 통합한 조직 전체의 DB를 기술하는 것
- DB에 저장되는 데이터와 그들간의 관계를 표현하는 스키마

내부스키마(Internal Schema)-물리적 저장구조

- DB가 물리적으로 저장된 형식
- 물리적 장치에서 데이터가 실제로 저장되는 방법을 표현하는 스키마

데이터베이스 3단계 구조에서의 데이터 독립성 2가지

논리적 독립성

- 개념 스키마가 변경되어도 외부 스키마에 영향을 미치지 않도록 지원하는 것
- 논리적 구조가 변경되어도 응용 프로그램에 영향 없음
- 사용자 특성에 맞는 변경 가능
- 통합 구조 변경 가능

물리적 독립성

- 내부 스키마가 변경되어도 외부/개념 스키마는 영향을 받지 않도록 지원하는 것
- 저장 장치의 구조 변경은 응용 프로그램과 개념 스키마에 영향 없음
- 물리적 구조 영향 없이 개념 구조 변경 가능
- 개념 구조 영향 없이 물리적인 구조 변경 가능

데이터 모델링의 3가지 요소

- 업무가 관여하는 어떤 것(Things)-엔티티(Entity)
- 어떤 것(Things)이 가지는 성격(Attributes)-속성
- 업무가 관여하는 어떤 것(Things) 간의 관계(Relationships)-관계

모델링 3요소의 단수와 집합(복수)의 명명

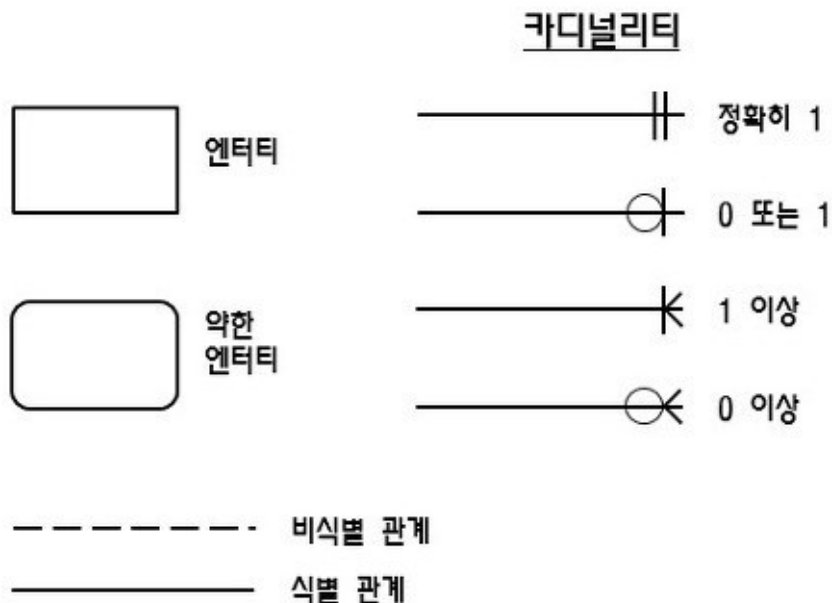
용어의 구분정의

구분	복수 / 집합 개념	개별 / 단수 개념
개념	타입 / 클래스	어커런스 / 인스턴스
어떤 것 (Thing)	엔터티 타입 (Entity Type) / 엔터티 (Entity)	엔터티 (Entity) / 인스턴스 (Instance), 어커런스 (Occurrence)
어떤 것 간의 연관 (Association between Things)	관계 (Relationship)	페어링 (Pairing)
어떤 것의 성격 (Characteristic of a Thing)	속성 (Attribute)	속성 값 (Attribute Value 1)

데이터 모델링의 이해관계자

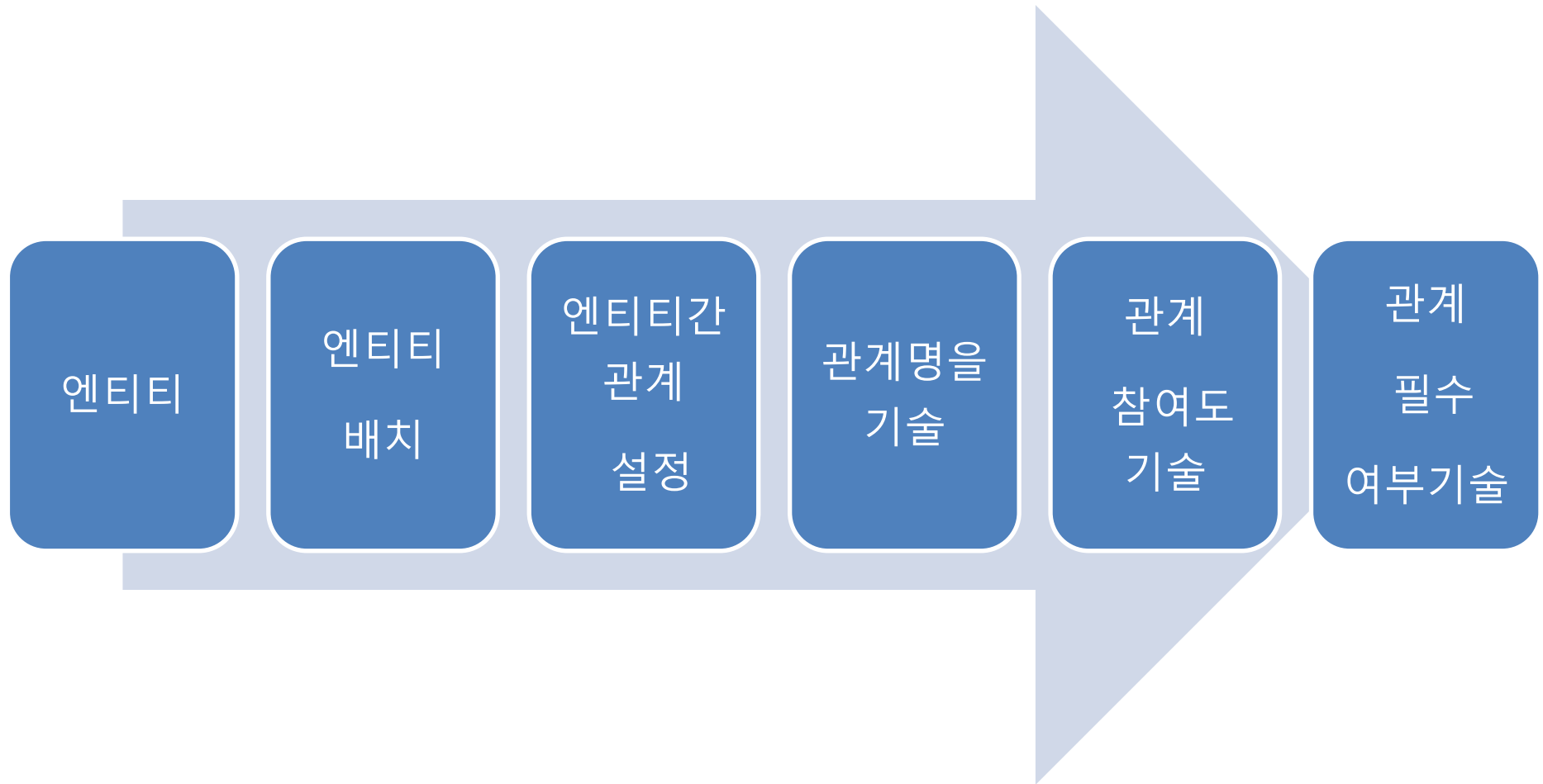
역할	설명
DBA (DataBase Administrator)	데이터베이스를 설계하고, 설치, 백업/복구, 보안, 성능 관리 등 운영 전반을 책임지는 관리자
프로젝트 개발자 (System Engineer)	실제 시스템을 코딩하고 구현하는 개발자. 데이터 모델을 기반으로 프로그램 개발을 수행
현업 업무 전문가	해당 시스템이 적용될 실제 업무 담당자. 업무 프로세스와 요구사항을 정확하게 파악하고 모델에 반영할 수 있도록 지원
전문 모델러	개념적, 논리적, 물리적 데이터 모델을 설계하는 전문가. 업무 요구사항을 구조화하고 최적의 데이터 구조로 모델링함

ERD(Entity Relationship Diagram)



- 1976 피터첸(Peter Chen)
- 업무분석에서 도출된 엔티티와 엔티티간의 관계를 도식화된 다이어그램으로 표시
- 해당업무에서 데이터의 흐름과 프로세스와의 연관성 표시
- 분석된 엔티티, 관계, 속성 정보가 표현

ERD 작업순서



엔티티 배치

가장 중요한 엔티티를 왼쪽상단에서 조금 아래쪽 중앙 배치

업무흐름의 중심이 되는 엔티티는 중앙에 배치

업무를 진행하는 중심 엔티티와 관계를 갖는 엔티티들은 중심엔티티 주위에 배치

ERD 관계명(Relationship Name) 설정 규칙의 표시

- 관계명은 관계 표현에서 중요한 요소이다.
- 관계 설정이 완료되면, 각 관계에 적절한 이름을 부여해야 한다.
- 관계명은 항상 '현재형 동사'를 사용하여 표현한다.
 - 예시: 주문한다, 소속된다, 작성한다
- 지나치게 포괄적인 단어 사용은 지양한다.
- 예: '관리한다'보다는 '등록한다', '수정한다' 등 구체적 표현 사용 권장

좋은 데이터 모델의 요소

- 데이터 재사용(Data Reusability)
 - 데이터의 통합성과 독립성을 충분히 고려하여 재사용성 향상
 - 데이터가 어플리케이션에 대해 독립적으로 설계
 - 간결한 모델의 기본적인 전제는 통합
- 의사소통(Communication)
 - 데이터분석 과정에서 업무규칙 도출
 - 업무규칙들이 엔티티, 속성, 관계등의 형태로 표현
- 통합성(Integration)
 - 동일한 데이터는 조직 전체에서 한번만 정의되고 여러 다른 영역에서 참조

완전성(Completeness)

- 업무에서 필요로 하는 모든 데이터가 데이터 모델에 정의되어야 함
- 데이터 모델 검증에서 가장 먼저 확인되어야 할 부분

중복배제(Non-Redundancy)

- 하나의 DB에 동일한 사실은 반드시 한번만 기록
- 예)하나의 테이블에 '나이' 와 '생년월일' 동시존재 불가
- 저장공간의 낭비
- 중복 데이터의 일관성 유지를 위한 추가적인 데이터조작 필요

업무규칙(Business Rules)

- 모델링에 업무규칙을 표현하고 모든사용자가 공유
- 모든사용자가 해당 규칙에 대해서 동인한 판다 및 데이터 조작

엔티티(Entity)의 개념

- 변별할 수 있는 사물 - Peter Chen(1976)
- 데이터베이스 내에서 변별 가능한 객체 - C.J Data(1986)
- 정보를 저장할 수 있는 어떤 것 - James Martin(1989)
- 정보가 저장될 수 있는 사람, 장소, 물건, 사건, 그리고 개념 등 - Thomas Bruce(1992)
- 정의들의 공통점
 - 엔티티는 사람, 장소, 물건, 사건, 개념 등의 명사에 해당
 - 엔티티는 업무상 관리가 필요한 관심사에 해당
 - 엔티티는 저장이 되기 위한 어떤 것(Thing)

엔티티(Entity)의 특징

- 반드시 해당 업무에서 필요하고 관리하고자 하는 정보이어야 한다
- 유일한 식별자에 의해 식별이 가능해야 한다
- 영속적으로 존재하는 인스턴스의 집합이어야 한다(두개 이상)
- 업무 프로세스에 의해 이용되어야 한다
- 반드시 속성이 있어야 한다
- 다른 엔티티와 최소 한 개 이상의 관계가 있어야 한다

엔티티(Entity)의 명명

- 현업 업무에서 사용하는 용어 사용
- 약어 사용 자제
- 단수명사 사용
- 모든 엔티티 에서 유일하게 이름 부여
- 엔티티 의 생성의미에 맞는 이름 부여

속성(Attribute)이란?

- 속성(Attribute)의 개념
 - 업무에서 필요로 한다
 - 의미상 더 이상 분리되지 않는다
 - 엔티티를 설명하고 인스턴스의 구성요소가 된다
- 속성(Attribute)의 특징
 - 해당 업무에서 필요하고 관리하고자 하는 정보
 - 정규화 이론에 근간하여 정해진 주식별자에 함수적 종속성을 가져야 함
 - 예)사원번호, 사원이름, 주소, 전화번호, 직책, 부서번호, 부서이름
 - 하나의 속성에는 한 개의 값을 가진다.

엔티티, 인스턴스와 속성, 속성값에 대한 내용과 표기법

구분	설명
엔터티	두 개 이상의 인스턴스를 가진 데이터 집합
인스턴스	엔터티의 실제 데이터를 의미함 (레코드, 행 단위)
속성	엔터티의 고유 성격을 표현하는 정보 항목, 2개 이상 존재
속성값	하나의 속성에는 하나의 값만 존재함 (원자성 원칙 준수)

속성(Attribute)의 특성에 따른 분류

- **기본속성**

- 업무로 부터 추출한 모든속성이 여기에 해당
- 엔터티에 가장 일반적이고 많은 속성을 차지
- 코드성, 일련번호, 계산의 결과나 다른 속성의 영향을 받아 생성된 속성 제외

- **설계속성**

- 데이터 모델링을 위해 업무를 규칙화하기 위해 속성을 새로 만들거나 변형하여 정의하는 속성
- 코드성 속성, 일련번호 속성 등

- **파생속성**

- 다른 속성에 영향을 받아 발생하는 속성
- 계산에 의해 생성된 값
- 데이터 정합성유지를 위해 유의
- 가급적 적게 정의하는 것이 바람직

속성(Attribute)의 특성에 따른 분류

➤ 속성의 분류 – 특성에 따른 분류

- ① 속성은 업무분석을 통해 바로 정의한 속성을 **기본속성(Basic Attribute)**
- ② 원래 업무상 존재하지는 않지만 설계를 하면서 도출해내는 속성을 **설계속성(Designed Attribute)**
- ③ 다른 속성으로부터 계산이나 변형이 되어 생성되는 속성을 **파생속성(Derived Attribute)**이라고 한다.

상품				
제품번호 ①	제품명 ①	제조일시 ①	제조원가 ①	제조사코드 ②
0000000001	LG그램12	2020-05-29 11:34:02	425,482	LGC
0000000002	갤럭시S2.4	2020-05-29 11:34:04	221,457	SSC
0000000003	애플워	2020-05-29 11:34:07	441,454	APC

상품판매합계		
제품번호 ①	판매일자 ①	판매금액 ③
0000000001	2020-06-05	55,425,482
0000000002	2020-06-05	15,221,457
0000000003	2020-06-05	87,441,454

① 기본속성

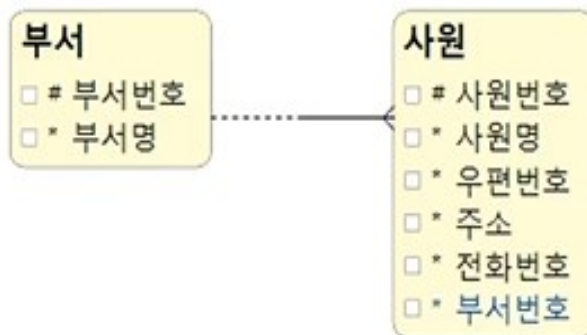
② 설계속성

③ 파생속성

속성(Attribute)의 엔터티 구성방식에 따른 분류

➤ 속성의 분류 – 엔터티 구성방식에 따른 분류

- ① 엔터티를 식별할 수 있는 속성을 PK(Primary Key)속성, 다른 엔터티와의 관계에서 포함된 속성을 FK(Foreign Key)속성, 엔터티에 포함되어 있고 PK, FK에 포함되지 않은 속성을 일반속성이라 한다.



속성분류	속성명
PK속성	부서(부서번호) 사원(사원번호)
FK속성	사원(부서번호)
일반속성	부서(부서명) 사원(사원명, 우편번호, 주소, 전화번호)

도메인(Domain)

- 속성값이 가질수 있는 범위
 - 예) 학생 엔티티의 학점 속성의 도메인은 0.0 에서 4.5 입니다.
 - 예) 학생 엔티티의 주소 속성의 도메인은 길이가 20자리 이내 입니다.
- 엔티티 내에서 속성에 대한 데이터타입과 크기 제약사항을 지정하는 것

속성(Attribute)의 명명

- 해당업무에서 사용하는 이름을 부여
- 서술식 속성 명 사용자제
- 약어사용 제한
- 전체 데이터모델에서 유일성 확보

관계(Relationship)의 개념

- 관계의 정의

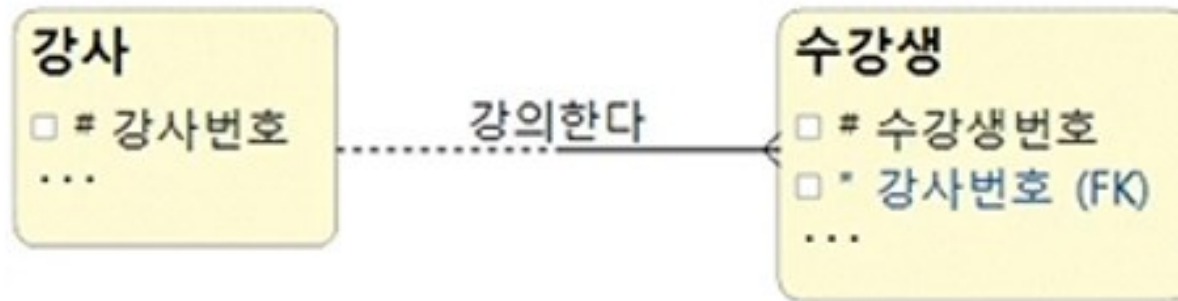
- 상호 연관성이 있는 상태
- 엔티티의 인스턴스 사이의 논리적인 연관성
- 존재의 형태로서나 행위로서 서로에게 연관성이 부여된 상태
- 엔티티와 엔티티 간 연관성을 표현
- 엔티티의 정의나 속성의 정의 및 관계정의에 따라서 다양하게 변화



- ❖ 인스턴스 사이의 논리적인 연관성으로서 존재 또는 행위로서 서로에게 연관성이 부여된 상태

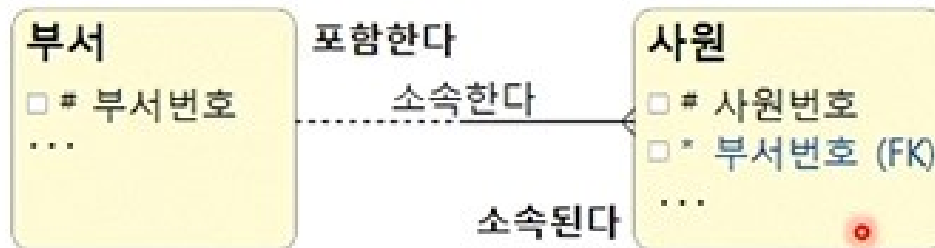
관계(Relationship)의 개념

- 관계의 패어링
 - 개별 인스턴스가 각각 다른 종류의 관계 형성 가능
 - 인스턴스와 인스턴스 사이에 관계가 설정되어 있는 어커런스



관계(Relationship)의 분류

➤ 관계의 분류 – 존재의 의한 관계



❖ 사원은 부서에 항상 속해 있다.

- “소속된다”라는 의미는 행위에 따른 이벤트에 의해 발생하는 의미가 아니고 그냥 사원이 부서에 소속되어 있기 때문에 나타나는
- 즉 존재의 형태에 의해 관계가 형성되어 있는 것이다.

➤ 관계의 분류 – 행위에 의한 관계



❖ 주문은 고객이 주문을 할 때 발생한다.

- 주문 엔터티의 주문번호는 고객이 ‘주문한다’라는 행위에 의해 발생되었기 때문에
- 두 엔터티 사이의 관계는 행위에 의한 관계가 되는 것이다

관계(Relationship)의 표기법

관계명

- 엔티티가 관계에 참여하는 형태를 지칭
- 각각의 관계는 두 개의 관계명을 가진다
- 참여자의 관점에 따라 관계이름이 능동적이거나 수동적으로 명명
- 애매한 동사는 피한다
- 현재형으로 표현
- 예) 부서와 사원의 관계(부서는 사원을 포함한다/사원은 부서에 소속된다)

관계차수(Degree/Cardinality)

- 1:1
- 1:M
- M:M

• 관계선택사양

– 필수참여관계

- 주문서와 주문목록

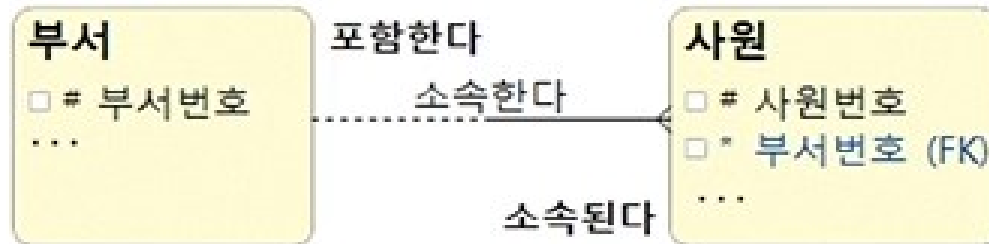
– 선택참여관계

- 목록과 주문목록
- Foreign Key로 연결될 경우 Null을 허용할 수 있는 항목
- ERD에서 선택참여하는 엔티티 쪽을 원으로 표시

관계(Relationship)의 표기법-관계명

➤ 관계의 표기법 - 관계명

- ① 엔터티가 관계에 참여하는 형태를 지칭한다.
- ② 각각의 관계는 두 개의 관계명을 가지고 있다.
- ③ 또한 각각의 관계명에 의해 두 가지의 관점으로 표현될 수 있다.

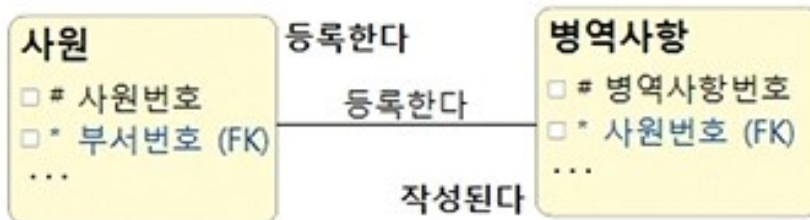


- 엔터티에서 관계가 시작되는 편을 관계시작점(The Beginning)이라고 부르고 받는 편을 관계끝점(The End)이라고 부른다.
- 관계 시작점과 끝점 모두 관계이름을 가져야 하며 참여자의 관점에 따라 관계이름이 능동적(Active)이거나 수동적(Passive)으로 명명된다.

관계(Relationship)의 표기법-관계차수

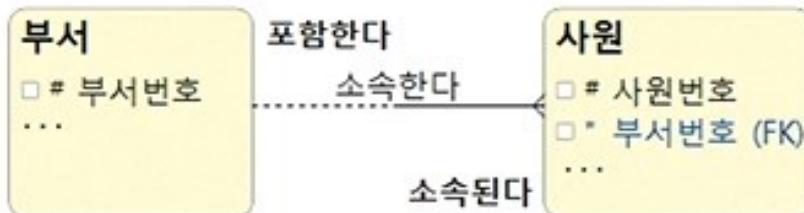
- ① 두 개의 엔터티 간 관계에서 참여자의 수를 표현하는 것을 **관계 차수(Cardinality)**라고 한다.
- ② 가장 일반적인 관계 차수 표현방법은 **1:M, 1:1, M:M**이다

▪ 관계 차수 (1:1)



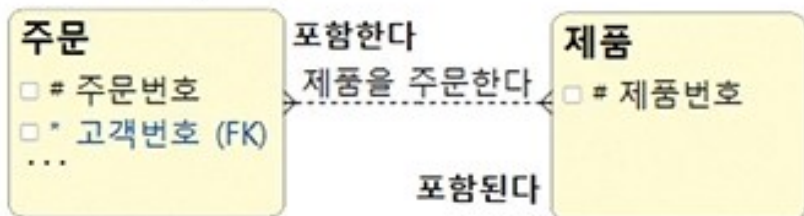
- 관계에 참여하는 각각의 엔터티는 관계를 맺는 다른 엔터티에 대해 단지 하나의 관계만을 가지고 있다.

▪ 관계 차수 (1:M)



- 한 명의 사원은 한 부서에 소속되고 한 부서에는 여러 사원을 포함한다.

▪ 관계 차수 (M:M)



- 관계에 참여하는 각각의 엔터티는 관계를 맺는 다른 엔터티에 대해 하나나 그 이상의 수와 관계를 가지고 있다

관계(Relationship)의 표기법-관계선택사양

▶ 관계의 표기법 - 관계선택사양(Optionality)

➤ 필수참여관계(Mandatory Relationship) - " 꼭 있어야 하는 관계"

•학생 - 학번

→ 학생이 존재하려면 반드시 학번이 있어야 한다.

•사원 - 부서

→ 사원은 반드시 하나의 부서에 소속되어 있어야 한다.

•주문 - 고객

→ 주문은 반드시 고객에 의해 발생해야 한다.

관계(Relationship)의 표기법-관계선택사양

선택참여관계(Optional Relationship) – "있을 수도 있고, 없을 수도 있는 관계"

- **고객 - 리뷰**

→ 고객은 리뷰를 작성할 수도 있고, 작성하지 않을 수도 있다.

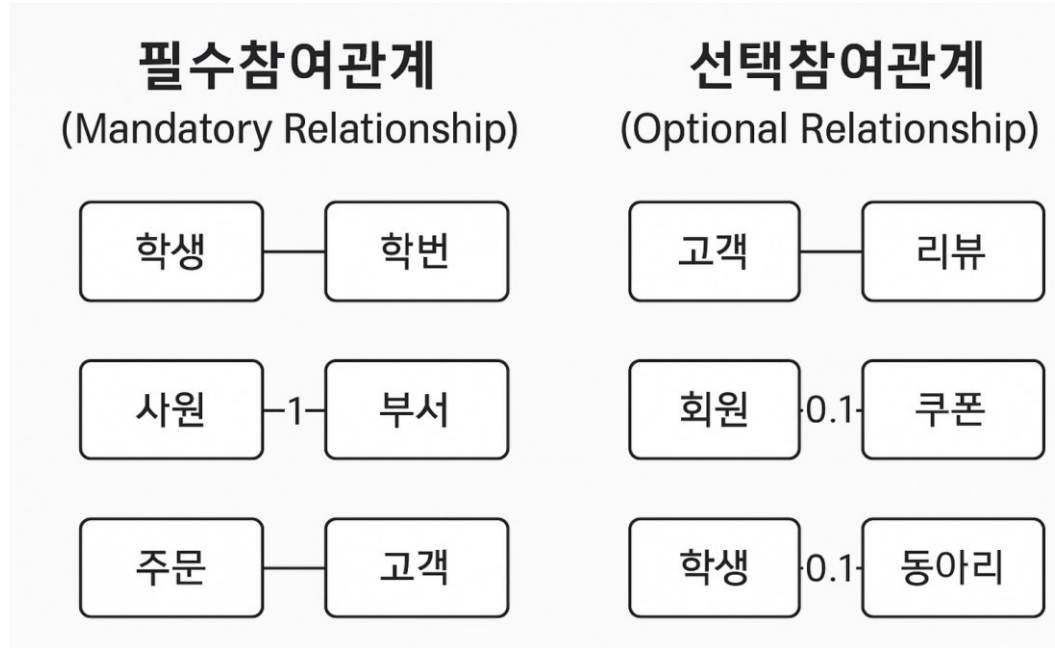
- **회원 - 쿠폰**

→ 회원은 쿠폰을 가질 수도 있고, 안 가질 수도 있다.

- **학생 - 동아리**

→ 학생은 동아리에 가입하지 않아도 된다.

관계(Relationship)의 표기법-관계선택사양



!! 필수냐, 선택이냐는 관계는 ERD 설계 시 중요한 기준이 되며,
선택참여관계는 관계선에서 '0' 표시, 필수참여관계는 '1 이상' 표시로 구분

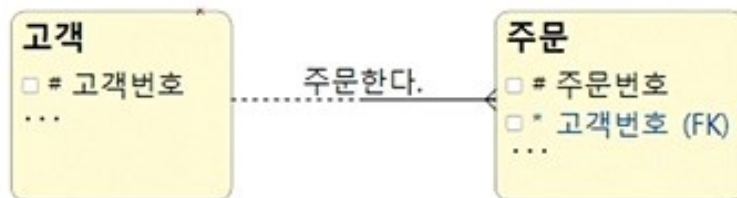
관계(Relationship)의 정의 및 읽은 방법

• 관계 체크사항

- 두 개의 엔티티 사이에 관심있는 연관규칙이 존재하는가?
- 두 개의 엔티티 사이에 정보의 조합이 발생하는가?
- 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?
- 업무기술서, 장표에 관계연결을 가능하게 하는 동사(Verd)가 있는가?

• 관계 읽기

- 기준 엔티티를 한 개 또는 각각으로 읽는다.



각각의/하나의	기준 엔터티	관계 차수	대상엔터티	필수/선택	관계명
각각의	고객은	여러 개의	주문을	때때로	주문한다.
각각의	주문은	하나의	고객을	반드시	가진다.

관계(Relationship)의 정의 및 읽은 방법

- 학생 ————— 가입한다 ► 동아리
- 동아리 ◀————— 소속된다 ————— 학생

각각의 /하나의	기준 엔터티	관계 차수	대상 엔터티	필수/선택	관계명
각각의	학생	0개 이상	동아리를	선택	가입한다
각각의	동아리	여러 명	학생을	필수	소속된다

학생은 때때로 동아리에 가입할 수도 있고, 가입하지 않을 수도 있다.

→ 선택참여 관계, 0개 이상

동아리는 반드시 하나 이상의 학생이 소속되어야 한다.

→ 필수참여 관계, 1개 이상

식별자(Identifiers)란?

식별자의 개념

- 여러 개의 집합체를 담고 있는 하나의 통에서 각각을 구분할수 있는 논리적인 이름
- 하나의 엔티티에 구성되어 있는 여러 개의 속성 중에 대표속성
- 하나의 엔티티는 반드시 하나의 유일한 식별자가 존재

식별자의 특징

- 유일성
- 최소성 - 유일성을 만족하는 최소의 수
- 불변성
- 존재성 - Not Null

식별자 분류

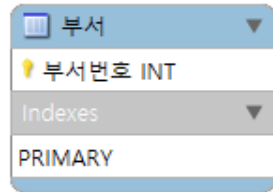
분류 기준	식별자 종류	설명	예시
대표성 여부	주식별자	엔티티 내의 각 어커런스를 고유하게 구분하며, 타 엔티티와의 참조 관계 연결에 사용됨	사원 엔티티의 사원번호
	보조식별자	엔티티 내의 각 어커런스를 고유하게 구분하지만, 대표성이 없어 타 엔티티와의 참조 관계 연결에 주로 사용되지 않음	고객 엔티티의 주민등록번호
생성 여부	내부식별자	엔티티 내부에서 스스로 생성되는 식별자	고객 엔티티의 고객번호
	외부식별자	타 엔티티와의 관계를 통해 타 엔티티로부터 받아오는 식별자	주문 엔티티의 고객번호 (고객 엔티티 참조)
속성의 수	단일식별자	하나의 속성으로 구성된 식별자	고객 엔티티의 고객번호
	복합식별자	둘 이상의 속성으로 구성된 식별자	주문상세 엔티티의 주문번호 + 상세순번
대체 여부	본질식별자	업무에 의해 만들어지는 식별자	고객 엔티티의 고객번호
	인조식별자	업무적으로 만들어지지 않는 않지만, 원조 식별자가 복잡한 구성이어서 인위적으로 만든 식별자	주문 엔티티의 주문번호 (고객번호 + 주문일자 + 순번 조합 등)

식별자 표기법

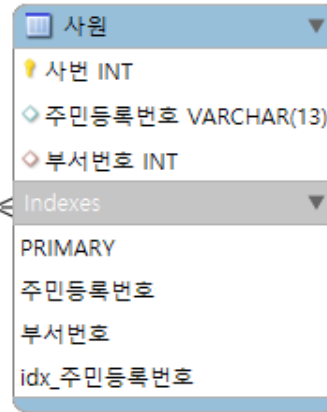
사번 : 주 식별자 (PK) + 내부 + 단일 + 본질

사번:
주 식별자 (PK, FK)
(사원 엔티티 참조)

부서번호 : 주 식별자 (PK) + 내부 + 단일 + 원조



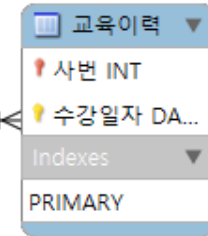
(1) -- (N)



보조식별자

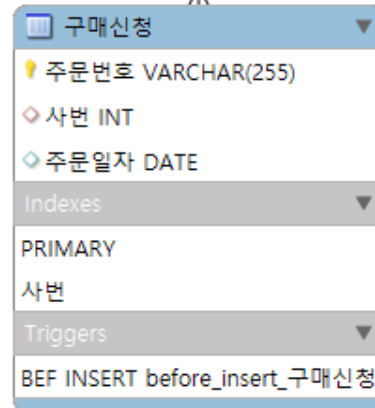
(1) -- (N)

부서번호:
외부 식별자 (FK)
(부서 엔티티 참조)



수강일자
주 식별자

(1) -- (N)



주문 번호: 주 식별자 (PK) + 단일 식별자 + 내부 식별자
인조 식별자 (주문 번호 = 사번 + 주문 일자 + 순번)
사번: 외부 식별자 (FK) (사원 엔티티 참조)
주문 일자: 주 식별자 (PK) (주문 번호 구성 요소)

식별자 관계란?

- 자식 엔터티의 주식별자가 부모 엔터티의 주식별자를 포함하고 있는 구조
- 자식 엔터티는 부모 엔터티가 존재해야만 생성 가능
- 부모 식별자는 자식의 **PK** 구성요소이자 **FK**
- 이 관계를 **식별자 관계 (Identifying Relationship)** 라고 함

비교 항목	식별자 관계 (Identifying Relationship)	비 식별자 관계 (Non-Identifying Relationship)
부모 FK 포함 여부	포함 (자식 테이블의 PK에 부모 FK가 포함됨)	별도 (자식 테이블의 PK에는 포함되지 않고, FK로만 존재)
부모 없이 생성 가능	❌ 불가능 (자식은 부모 없이는 존재할 수 없음)	가능 (자식은 부모와 독립적으로 존재할 수 있음)
자식 PK 구성	부모 PK + 자식 PK 속성 조합 (복합 키)	자식 PK 속성만으로 단독 구성 (단일 또는 복합 키)
관계 강도	강한 결합 (Strong Dependency)	느슨한 결합 (Loose Dependency)

식별자 관계

항목	설명
관계 유형	식별자 관계 (Identifying Relationship)
부모 테이블	식별자관계_부모 (부모번호 PK)
자식 테이블	식별자관계_자식 (부모번호 + 자식순번 → 복합 PK, 부모번호는 FK)
외부 식별자	부모번호 → 자식 PK의 일부이자 FK 역할 수행
구조적 특징	자식 테이블은 부모 테이블 없이는 생성될 수 없음, 관계가 강하게 결합되어 있음



비식별자 관계

항목	설명
관계 유형	비 식별자 관계 (Non-Identifying Relationship)
부모 테이블	비식별자관계_부모 (부모번호 PK)
자식 테이블	비식별자관계_자식 (자식번호 PK + 부모번호는 FK만 담당)
외부 식별자	부모번호 (자식 테이블에서 FK 역할만 하고, PK에는 포함되지 않음)
구조적 특징	자식 테이블은 부모 테이블 없어도 생성 가능 (필수 관계 아님), 독립적인 PK 가 짐



식별자 관계와 비식별자 관계

항목	식별자 관계 (강한 연결 관계)	비 식별자 관계 (약한 연결 관계)
목적	강한 연결 관계 표현	약한 연결 관계 표현
자식 주 식별자 영향	자식 주 식별자의 구성에 부모 주 식별자가 포함됨	자식 주 식별자의 구성에 부모 주 식별자가 포함되지 않음 (자식 일반 속성에 포함됨)
종속성	반드시 부모 엔터티에 종속 (강한 종속 관계)	약한 종속 관계 (부모 엔터티가 없어도 자식 엔터티가 존재 가능)
연결	자식 주 식별자 구성에 부모 주 식별자 포함 필요	자식 주 식별자 구성을 독립적으로 구성
고려 사항	상속받은 주 식별자 속성을 타 엔터티에 이전 필요	상속받은 주 식별자 속성을 타 엔터티에 이전 차단 필요, 부모 쪽의 관계 참여가 선택 관계일 수 있음

연습문제

문제 1. 다음 중 정보시스템을 모델링할 때 세 가지 관점에 해당하지 않는 것은?

- ① 업무가 어떤 데이터와 관련이 있는지 분석
- ② 업무가 실제하는 일은 무엇인지 또는 무엇을 해야 하는지 분석
- ③ 업무가 처리하는 일의 방법에 따라 데이터가 어떻게 영향을 받는지 분석
- ④ 업무를 처리할 수 있는 프로그램 구성을 어떻게 해야 하는지 분석

문제 2. 데이터 모델링의 세 가지 중요개념에 속하지 않는 것은?

- ① 업무가 관여하는 어떤것(Things)
- ② 업무가 관여하는 어떤것의 행위(Events)
- ③ 업무가 관여하는 어떤것의 성격(Attributes)
- ④ 업무가 관여하는 어떤것의 관계(Relationships)

연습문제

문제 3. 발생시점에 따라 구분할 수 있는 엔터티의 유형이 아닌 것은?

- ① 행위 엔터티(Action Entity)
- ② 중심 엔터티(Main Entity)
- ③ 개념 엔터티(Conceptual Entity)
- ④ 기본 엔터티(Basic Entity)

문제 4. 이 속성이 없어도 다른 속성을 이용하여 결과를 도출할 수 있는 특징을 가진 속성의 이름은?

- ① 설계 속성(Designed Attribute)
- ② 파생 속성(Derived Attribute)
- ③ 기본 속성(Basic Attribute)
- ④ 관계 속성(Associative Attribute)

연습문제

문제 5. 다음 중 엔터티의 특징에 포함되지 않는 것은?

- ① 반드시 해당 업무에서 필요하다고 관리하고자 하는 정보 이여야 한다.
- ② 유일한 식별자에 의해 식별이 가능해야 한다.
- ③ 엔터티는 업무 프로세스에 의해 이용되어야 한다.
- ④ 엔터티는 반드시 속성이 없어도 된다.

문제 6. 다음 설명이 나타내는 데이터 모델의 개념은 무엇인가?

학생이라는 엔터티가 있을 때 학점이라는 속성의 값의 범위는 0.0 에서 4.0 사이의 실수 값이며 주소라는 속성은 길이가 20자리 이내의 문자열로 정의할 수 있다.

- ① 도메인(Domain)
- ② 용어사전(Word Dictionary)
- ③ 속성사전(Attribute Dictionary)
- ④ 시스템카탈로그(System catalog)

연습문제

문제 7. 엔터티간 1:1, 1:M과 같이 관계의 기수성을 나타내는 것을 무엇이라 하는가?

- ① 관계명(Relationship Membership)
- ② 관계차수(Relationship Degree/Cardinality)
- ③ 관계선택성(Relationship Optionality)
- ④ 관계정의(Relationship Definition)

문제 8. 관계를 정의할 때 주요하게 체크해야 하는 사항과 거리가 먼 것은?

- ① 두 개의 엔터티 사이에 관심있는 연관규칙이 존재하는가?
- ② 두 개의 엔터티 사이에 정보의 조합이 발생하는가?
- ③ 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?
- ④ 업무기술서, 장표에 관계연결을 가능하게 하는 명사(Noun)가 있는가?

연습문제

문제 9. 식별자의 대체여부에 따라 분류하는 방식은?

- ① 주식별자 - 보조식별자
- ② 내부식별자 - 외부식별자
- ③ 본질식별자 - 인조식별자
- ④ 단일식별자 - 복합식별자

문제 10. 다음 개념에 해당하는 관계는 어떤 관계를 설명한 것인가?

부모엔티티로부터 속성을 받았지만 자식엔티티의 주식별자로 사용하지 않고 일반적인 속성으로만 사용한다.

- ① 식별자관계(Identifying Relationship)
- ② 일반속성관계(Attribute Relationship)
- ③ 비식별자관계(Non-Identifying Relationship)
- ④ 외부식별관계(Foreign Key Relationship)