

제 1 절. 데이터 모델링의 이해

1. 모델링의 이해

모델링의 정의 : 현실세계를 추상화/단순화/명확화 하여 모델로 나타내는 일.

- 추상화 : 현실을 일정한 형식에 맞추어 표현하는 것.
- 모델링의 세 가지 관점
 - 데이터 관점(what) : 무엇을 모델링
 - 프로세스 관점(How) : 무엇을 해야하는지 모델링
 - 상관 관점(Data vs Process) : 업무처리에 따라 데이터의 영향을 모델링(interaction)

2. 데이터 모델의 기본 개념의 이해

데이터 모델링 : 정보시스템 구축 위한 데이터 관점 업무 분석기법 / 현실세계의 데이터를 특정 표기법으로 표현/ DB 를 구축하기 위한 분석,설계의 과정

데이터 모델 기능 : 시스템 가시화, 명세화, 구조화, 문서화, 다양한 관점 위해 특정 세부사항 on/off, 구체화

3. 데이터 모델링의 중요성 및 유의점

데이터 모델링이 중요한 이유 : 파급효과, 복잡한 정보 요구사항의 간결한 표현, 데이터 품질

- 데이터 품질 유의점 : 중복, 비유연성, 비밀관성

4. 데이터 모델링의 3단계 진행

- 개념적 데이터 모델링 : 추상화 수준 높음, 업무중심, 포괄, 전사적, EA 수립시 사용
- 논리적 데이터 모델링 : 시스템으로 구축하고자 하는 업무에 대해 Key, 속성, 관계 등을 정확히 표현, 재사용성 높음
- 물리적 데이터 모델링 : 실제 DB 에 이식 가능하도록 성능, 저장등 물리적 성격 고려하여 설계

5. 프로젝트 생명주기(Life Cycle) 에서 데이터 모델링

- 분석단계 : 논리/개념적 데이터 모델링
- 설계단계 : 물리적 데이터 모델링

6. 데이터 모델링에서 데이터독립성의 이해

- 데이터 독립성 필요하다.
- 데이터 독립성 구성요소
 - 외부 스키마 : View 단계, 개개 사용자들이 보는 개인적 DB스키마
 - 개념 스키마 : 모든 사용자, 모든 응용시스템 관점을 통합한 DB
 - 내부 스키마 : DB의 물리적구성

7. 데이터 모델링의 중요한 세 가지 개념

- 어떤 것(Thing) / 성격(Attributes) / 관계(Relationships)
- 데이터 모델링에서의 용어
 - 어떤 것을 나타내는 용어 : Entity(복수), Instance(단수)
 - 성격을 나타내는 용어 : Attribute(복수), Attribute Value(단수)

- 관계를 나타내는 용어 : Relationship(복수), Pairing(단수)

8. 데이터 모델링의 이해관계자

- 개발자, DBA, 모델러, 업무전문가

9. ERD의 이해

- ERD(E-R Diagram) : Entity-Relationship Diagram
- 표기법 : IE/Crow's Foot 이 짝
- 작성법 : Entity 도출 => Entity 배치 => 관계 설정 => 관계명 기술 => 참여도 기술 => 필수여부 기술
- 중요 Entity 일수록 왼쪽상단
- 관계차수(Cardinality) 표현 (1:1, 1:M, M:M)
- 필수일 경우 o 표시 (Crow's Foot)

10. 좋은 데이터 모델의 요소

- 완전성 / 중복배제 / 업무규칙 / 데이터의 재사용 / 의사소통 / 통합성 이 잘 나타난 모델

제 2 절. 엔터티(Entity)

1. 엔터티의 개념

- Entity : 업무에 필요하고 유용한 정보를 저장, 관리하기 위한 집합적인 것

2. 엔터티와 인스턴스에 대한 내용과 표기법

- ERD에서 엔터티 네모로 그림

3. 엔터티의 특징

- 업무에서 필요로 하는 정보
- 유일한 **식별자**에 의해 식별이 가능해야 함
 - instance 각각을 유일하게 구분 가능한 식별자 필요
- 영속적으로 존재하는 인스턴스의 집합 (instance 2개 이상)
- 업무 프로세스에 의해 이용
- 반드시 속성이 있어야 함
- 다른 엔터티와 최소 한 개 이상의 관계 필요
 - 통계, 코드, 로그 등을 나타내는 엔터티는 예외

4. 엔터티의 분류

- 유무형에 따라
 - 유형 엔터티 : 물리적형태 있음
 - 사건 엔터티 : 업무 수행에 따른 발생 (주문, 청구)
 - 개념 엔터티 : 관리가능한 개념적 정보 (보험상품)
- 발생시점에 따라
 - 기본(키) 엔터티 : 독립 생성, 부모역할, 고유한 주식별자 (사원, 부서)
 - 중심 엔터티 : 기본 엔터티로 부터 발생, 업무중심역할 (접수, 계약)
 - 행위 엔터티 : 프로세스 진행하며 도출 (주문내역)

5. 엔터티의 명명

- 현업 용어 사용
- 약어 가능한 배제
- **단수명사**
- 각 엔터티 이름은 유일
- 의미대로 자연스럽게

제 3 절. 속성(Attribute)

1. 속성의 개념

- 인스턴스에서 관리하고자 하는 의미상 더 이상 분리되지 않는 최소의 데이터 단위
- 엔터티는 속성들에 의해 설명됨.

2. 엔터티, 인스턴스와 속성, 속성값에 대한 내용과 표기법

- 한 개의 엔터티는 2 개 이상의 인스턴스 집합
- 한 개의 엔터티는 두 개 이상의 속성
- 한 개의 속성은 한 개의 속성값을 갖는다.

3. 속성의 특징

- 해당 업무 필요한 정보
- 정규화 이론에 근간, 정해진 주식별자에 함수적 종속성(FD)를 가져야 함
- 한 속성에는 하나의 속성값.

4. 속성의 분류

- 기본 속성 : 업무로부터 추출한 모든 속성. (이름, 날짜, 가격 ...)
- 설계 속성 : 업무를 규칙화하기 위해 새로 정의한 속성 (약품코드)
- 파생 속성 : 다른 속성에 영향을 받아 발생하는 속성. 계산된 값. (전체 용기 수)

5. 도메인

- 각 속성이 가질 수 있는 값의 범위. (학점 : 0 ~ 4.5 실수 값)

6. 속성의 명명

- 업무에서 사용하는 이름
- 서술식 속성명 사용하지 않음.
- 약어 사용 가급적 제한
- 전체 데이터 모델에서 유일성 확보하는것이 좋음.

제 4 절. 관계

1. 관계의 개념

- 인스턴스 사이의 논리적인 연관성으로서 존재 또는 행위로서 서로에게 관련성이 부여된 상태

2. 관계의 분류

- 존재에 의한 관계와 행위에 의한 관계로 구분 가능
 - 존재에 의한 관계 : "사원은 부서에 항상 속해있다"
 - 행위에 의한 관계 : "주문은 고객이 주문을 할 때 발생된다"
- 클래스 다이어그램에서는 연관관계(실선) / 의존관계(점선) 로 구분하여 표기
- ERD 에서는 구분없이 단일화 된 관계로 표기

3. 관계의 표기법

- 관계명 / 관계 차수(Cardinality) / 관계 선택사항 (필수 or 선택)
- 관계 차수
 - 1:1, 1:M, M:M
 - 1 일 경우 실선위 ' | ' 표시, Many 일 경우 까마귀 발 표시
- 관계 선택사항
 - 모든 인스턴스가 반드시 관계를 가지는지 아닌지
 - FK에 NULL 을 허용할지 말지
 - 없어도 되는경우 실선위 ' o ' 표시. 0을 의미한다고 생각하면 됨 (없는경우 1부터)

4. 관계의 정의 및 읽는 방법

- 관계 체크사항
 - 두 개의 엔터티 사이에 연관규칙이 존재하는가?
 - 두 개의 엔터티 사이에 정보의 조합이 발생하는가?
 - 업무기술서, 장표에 관계 연결에 대한 규칙이 있는가?
 - 업무기술서, 장표에 관계 연결을 가능하게 하는 동사가 있는가?
- 관계 읽기
 - 한 엔터티를 기준(Source)로 잡고 읽음.
 - (각각의/하나의) + (Source E.) + 은/는 + (관계차수) + (Target E.) + 이/에 + (선택사항) + 속한다.
 - 각각의 사원은 한 부서에 항상 속한다.
 - 각 부서에는 여러 사원이 때때로 소속된다.

제 5 절. 식별자

1. 식별자(Identifiers) 개념

- 하나의 엔터티에 구성되어 있는 여러 속성 중 엔터티를 대표하는 속성
- 하나의 엔터티는 반드시 하나의 유일한 식별자가 존재
- **식별자는 엔터티내에서 인스턴스들을 구분할 수 있는 구분자**

2. 식별자의 특징

- 유일성 : 주식별자에 의해 엔터티 내의 모든 인스턴스들이 유일하게 구분
- 최소성 : 주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수
- 불변성 : 지정된 주식별자의 값은 자주 변하지 않아야 함
- 존재성 : 주식별자가 지정되면 반드시 값이 있어야 함 (Not Null)

3. 식별자 분류 및 표기법

- 식별자 분류
 - 대표성 여부 : 주식별자 / 보조식별자
 - 스스로 생성 여부 : 내부식별자 / 외부식별자(FK)
 - 속성의 수 : 단일식별자 / 복합식별자
 - 대체 여부 : 본질식별자(대체불가 유일) / 인조식별자(다른 식별자들을 더해서 만든 경우)

4. 주식별자 도출기준

- 업무에서 자주 이용되는 속성 (사원번호)
- 명칭, 내역같이 이름으로 기술되는 것들은 지양 (WHERE "신당동장독대를뛰쳐나온떡볶이총각의맛있는프로포즈")
- 복합으로 주식별자 구성할 경우 너무 많지 않게 (WHERE ... AND ... AND ... AND ... AND ..)
 - 인조식별자 생성 고려

5. 식별자관계와 비식별자관계에 따른 식별자

- 자식 엔터티에서는 부모로 부터 받은 외부식별자(FK)를 자신의 주식별자(PK) 로 사용할지 또는 부모와 연결이 되는 속성으로만 사용할 지 결정함.
- 식별자관계 : 부모로부터 받은 식별자를 자식엔터티의 주식별자로 이용
 - NULL 값이 오면 안되므로 반드시 부모 엔터티 생성 후 자식 엔터티 생성.
 - 강한 연결관계 / 실선 / 반드시 부모엔터티 종속
- 비식별자관계 : 부모엔터티로부터 속성을 받았지만 자식엔터티의 주식별자로 사용하지 않고 일반 속성으로만 사용하는 경우
 - 자식 엔터티에서 받은 속성이 반드시 필수가 아니어도 됨(NULL), 부모 없이 자식만 생성 가능
 - 약한 연결관계 / 점선 / 자식 주식별자구성을 독립적으로 구성