* 1. **데이터 모델의 이해**

**모델링 : 복잡한 현실세계를 단순화시켜 표현하는 것, 사물 또는 사건에 관한 양상(Aspect) 이나 관점(Perspective)을 연관된 사람이나 그룹을 위하여 명확하게 하는 것, 현실 세계의 추상화된 반영**

현실세계 -------(추상화 단순화 명확화)---------🡪 데이터모델

**문제) 모델링의 특징이 아닌것은?**

**모델링의 세가지 관점**

1. **데이터(Data, What) : 어떤 데이터와 관련**
2. **프로세스(Process, How) : 무엇을 해야하는지**
3. **상관(Data vs Process) : 일의 방법에 따라 데이터는 어떻게 영향받는지**

**데이터 모델링 : 정보 시스템을 구축하기 위한 데이터 관점의 업무 분석 기법, 현실세계의 데이터에 대해 약속된 표기법에 의해 표현되는 과정, 데이터베이스를 구축하기 위한 분석/설계의 과정**

**데이터 모델 제공 기능 :**

1. **가시화 : 시스템을 원하는 현재/원하는 모습으로 나타냄**
2. **명세화 : 시스템의 구조화 행동을 명세화시킴**
3. **구조화된 틀 : 시스템을 구축하는 구초화된 틀 제공**
4. **문서화 : 시스템을 구축하는 과정에서 결정한 것을 문서화시킨다**
5. **다양한 관점 : 다양한 영역에 집중하기 위해 다른 영역의 세부 사항은 숨기는 다양한 관점 제공**
6. **구체화 : 특정 목표에 따라 구체화된 상세 수준의 표현방법을 제공**

**데이터 모델링의 중요성 및 유의점**

1. **파급효과**
2. **복잡한 정보 요구사항의 간결한 표현**
3. **데이터 품질**

**데이터 모델링의 3단계 진행**

1. **현실세계의 개념 데이터 모델링(추상적) -> 개념적 구조**
2. **개념적 구조를 논리 데이터 모델링 -> 논리적 구조**
3. **논리적 구조를 물리 테이터 모델링(구체적) -> 물리 구조(데이터베이스)**
4. **개념적 데이터 모델링 : 고객이 상품 주문**
5. **논리적 데이터 모델링 : 주문번호, 주문상품명, 고객번호, 고객 주소**
6. **물리적 데이터 모델링 : 실질 데이터베이스 저장**

**데이터 독립성의 필요성**

1. **유지비용 및 데이터 복잡도 낮추고 중복데이터 줄이기 위한 목적**
2. **화면과 데이터베이스 간 서로 독립성을 유지하기 위한 목적**

**유지보수, 데이터 중복성, 데이터 복잡도 증가, 요구사항 대응 저하 -> 데이터 독립성 필요**

**데이터베이스 3단계 구조**

1. **외부 단계 외부 스키마1, 외부스키마2…**

**----------------------------------------------------------------------논리적 데이터 독립성**

1. **개념적 단계 개념 스키마**

**----------------------------------------------------------------------물리적 데이터 독립성**

1. **내부적 단계 내부 스키마**

**외부 스키마 : 사용자 관점**

**개념 스키마 : 통합 관점**

**내부 스키마 : 물리적 관점**

**문제) 데이터베이스 3단계 관점이 아닌것은?**

**논리적 데이터 독립성 : 개념 스키마가 변경되어도 외부 스키마에는 영향 x**

**물리적 독립성 : 내부 스키마가 변경되어도 외부/개념 스키마에 영향 x**

**데이터 모델링 세 가지 요소**

1. **어떤 것 : 엔터티**
2. **어떤 것 간의 연관 : 관계**
3. **어떤 것의 성격 : 속성**

데이터 모델링의 작업 순서

1. 엔터티 그린다
2. 배치
3. 관계 설정
4. 관계명 기술
5. 관계 참여도 기술
6. 관계 필수 여부 기술 ex) 주문은 반드시 고객을 가져야 한다

**엔터티(Entity) : 명사, 업무상 관리가 필요한 관심사, 저장이 되기 위한 어떤 것(thing)이다.**

**엔터티는 인스턴스의 집합이라고 할 수 있다.**

****

**엔터티 표기법**

****

**엔터티 특징**

1. **업무에서 필요로 하는 정보**
2. **식별 가능 : 인스턴스 구별하기 위함**
3. **인스턴스 집합**
4. **업무 프로세스에 의해 이용 : 업무 프로세스가 반드시 이용해야한다**
5. **속성 포함 : 반드시 속성이 포함되어야한다**

****

1. **관계의 존재 : 다른 엔터티와 최소 한 개 이상의 관계가 존재해야한다**

****

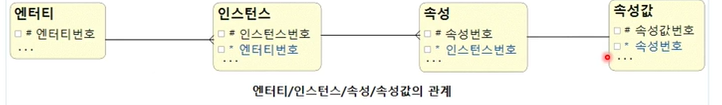
**엔터티의 분류**

1. **유무형에 따른 분류**
2. **유형 : 물리적 형태가 있고 안정적/지속적으로 활용됨 ex) 사원, 물품, 강사**
3. **개념 : 물리적 형태 존재 x, 관리해야할 개념적 정보ex) 조직, 보험상품**
4. **사건 : 업무를 수행함에 따라 발생하는 엔터티 ex) 주문, 청구, 미납**
5. **발생시점에 따른 분류**
6. **기본 : 독립적 생성 가능, 자신은 타 엔터티의 부모 역할ex) 사원, 부서, 고객, 상품, 자재**
7. **중심 : 본엔터티로부터 발생, 업무 중심역할, 데이터 양 많이 발생되며 다른 엔터티와의 관계를 통해 많은 행위 엔터티 생성ex) 계약, 사고, 예금원장, 청구, 주문, 매출**
8. **행위 : 두 개 이상의 부모엔터티로부터 발생, 자주 내용 변경됨ex) 주문목록, 사원변경 이력**

**속성(Attribute) : 업무에서 필요로 하는 인스턴스에서 관리하고자 하는 의미상 더 이상 분리되지않는 최소 데이터 단위**

**엔터티, 인스턴스, 속성, 속성값의 관계**

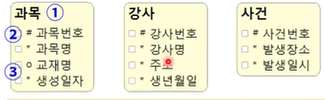
1. **한 개의 엔터티는 두 개 이상 인스턴스 집합이어야한다.**
2. **한 개의 엔터티는 두 개 이상의 속성 가짐**
3. **한 개의 속성은 한 개의 속성값을 가진다.**

****

**이름, 주소, 생년월일과 같은 각각의 값을 대표하는 이름들을 속성이라고 하고, 홍길동, 서울시 강서구, 1967년 12월31일과 같이 각각의 이름에 대한 구체적인 값을 속성값(value)라고 한다.**

**표기법 :**

1. **속성명 기재**
2. **해당 속성이 식별자인지 아닌지**
3. **필수값(\*)인지 선택값(0)인지**

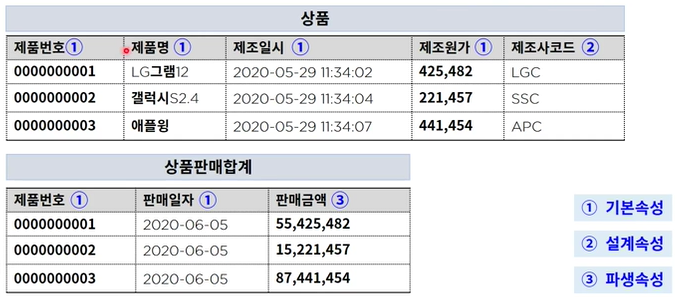
****

**특징:**

1. **엔터티와 마찬가지로 해당 업무에서 필요하고 관리하고자 하는 정보 ex)강사의 교재이름**
2. **주 식별자에 함수적 종속성을 가짐**
3. **하나의 속성에는 하나의 값만 가짐**

**특성에 따른 분류 :**

1. **기본속성(Basic) : 업무분석을 통해 바로 정의한 속성**
2. **설계속성(Designed) : 업무상 존재하지는 않지만 설계하면서 도출해낸 속성**
3. **파생속성(Derived) : 다른 속성으로부터 계산이나 변형이 되어 생성된 속성**

****

**엔터티 구성방식에 따른 분류**

1. **PK(Primary Key) : 엔터티 식별 가능**
2. **FK(Foreign Key) : 다른 엔터티와의 관계에서 포함된 속성**
3. **일반 속성 : 엔터티에 포함되어있고 PK, FK가 아닌 속성**

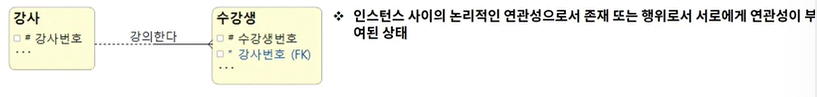
****

**도메인(Domain) : 각 속성은 가질 수 있는 값의 범위가 있는데 이를 그 속성의 도메인이라고 한다.**

Ex) 학생이라는 엔터티가 있을 때 학점이라는 속성의 도메인은 0.0에서 4.0 사이의 실수 값이며 주소라는 속성은 길이가 20자리 이내인 문자열로 정의

각 속성은 도메인 이외의 값을 갖지 못한다.

**관계(Relationship) : 엔터티의 인스턴스 사이의 논리적인 연관성으로서 존재의 형태로서나 행위로서 서로에게 연관성이 부여된 상태**

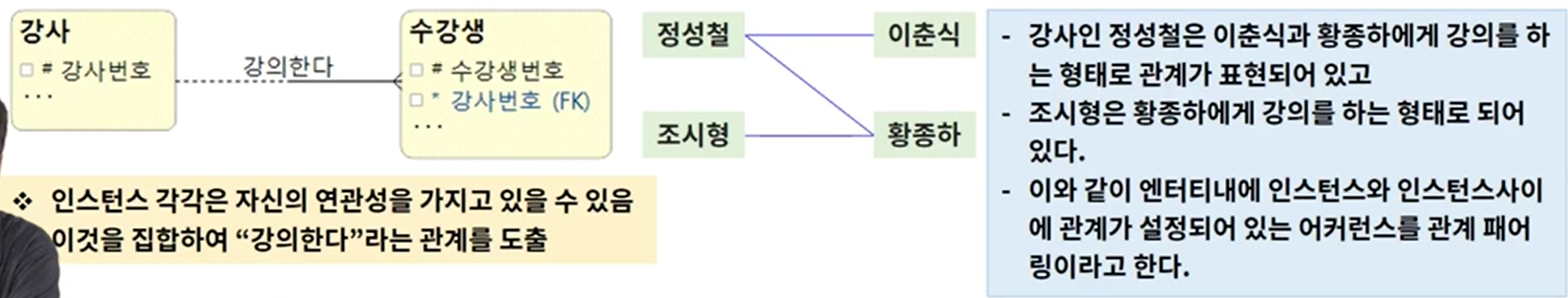
****

**관계의 페어링**

관계는 엔터티 안에 인스턴스가 개별적으로 관계를 가지는 것(페어링)이고 이것의 집합을 관계로 표현한다는 것이다.

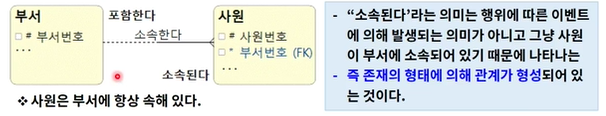
개별 인스턴스가 각각 다른 종류의 관계를 자기고 있다면 두 엔터티 사이에 두 개 이상의 관계가 형성될 수 있다.

**각각의 엔터티의 인스턴스들은 자신이 관련된 인스턴스들과 관계의 어커런스로 참여하는 형태를 관계 페어링이라고 한다.**

****

**관계의 분류**

1. **존재에 의한 관계**

****

1. **행위에 의한 관계**

****

**관계의 표기법 - 관계 차수(Cardinality) : 두 개의 엔터티 간 관계에서 참여자의 수를 표현하는 것**

가장 일반적인 관계 차수 표현방법은 1:1, 1:M, M:N이다.

**1:1**

****

**1:M**

****

**M:N**

****

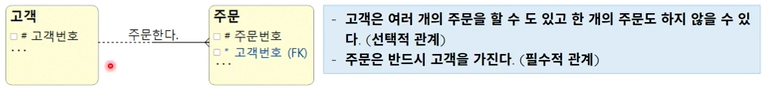
**관계의 표기법 – 관계선택사항(Optionality)**

“반드시 지하철의 문이 닫혀야하만 지하철은 출발한다”

지하철 출발과 지하철 문 닫힘은 필수적 관계 - > **필수 참여 관계(Mandatory)**

**“**안내방송 시스템”

지하철 출발과 지하철 방송과는 선택적 관계 -> **선택 참여 관계(Optional)**

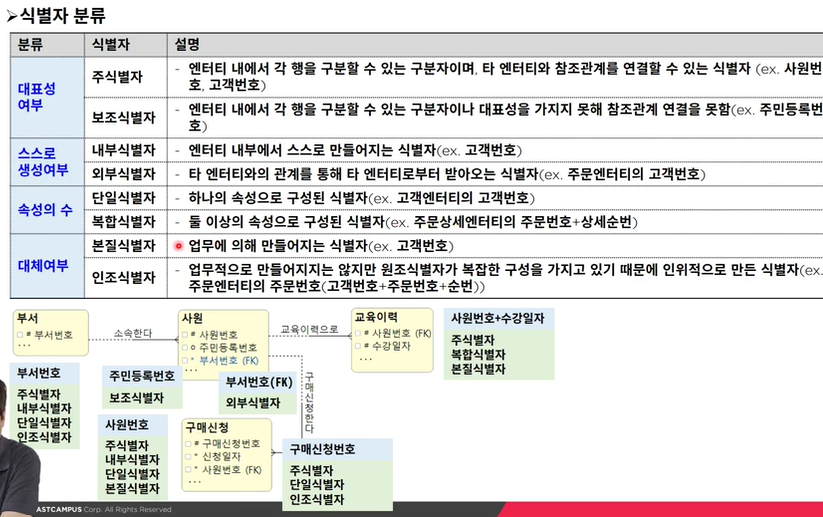


**선택적 관계는 점선표기, 필수적 관계는 실선표기**

**식별자(Identifier) : 여려개의 집합체를 담고 있는 하나의 통(엔터티)에서 각각을 구분할 수 있는 논리적인 이름이 있어야한다. 이 구분자를 식별자라고 한다.**

**특징 :**

1. **유일성 : 유일하게 구분되어야한다**
2. **최소성 :최소의 수**
3. **불변성 : 자주 변하지 않는 것**
4. **존재성 : 반드시 값이 있어야함**

****

**식별자관계와 비식별자 관계의 결정**

**외부식별자(Foreign Identifier)는 자기 자신의 엔터티에서 필요한 속성이 아니라 다른 엔터티와의 관계를 통해 자식 쪽에 엔터티에 생성되는 속성을 외부식별자라 하며, 데이터베이스 생성 시에 FK의 역할을 한다.**

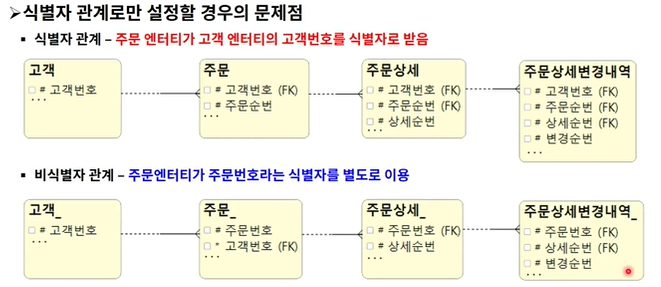
**자식 엔터티에서 부모엔터티로부터 받은 외부 식별자를 자신의 주 식별자로 사용할 것인지(식별자 관계) 또는 부모와 연결이 되는 속성으로만 이용할 것인지를 결정(비식별자 관계)해야한다.**

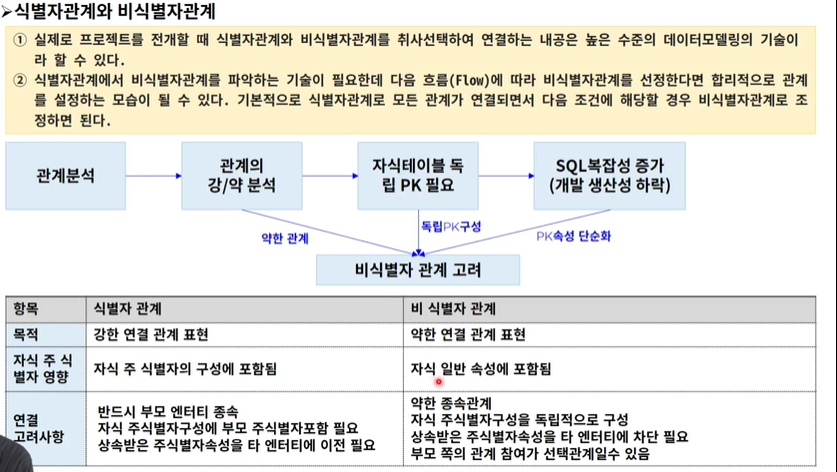
****

**식별자 관계는 반드시 부모엔터티가 생성되어야 자기 자신의 엔터티가 생성되는 경우**

**비식별자 관계는 부모 없는 자식 가능**

**식별자 관계로만 설정할 경우 문제점**

****

****

****

**: SQLD.TB\_EMP\_TEMP 테이블에는 DEPT\_CD 칼럼이 존재하며 SQLD.TP\_DEPT\_TEMP 테이블의 DEPT\_CD 칼럼을 참조해라 라는 뜻**