제 9장 통합 분석/설계 방법론

9.1 통합 분석/설계 방법론 개요

9.2 기능모형 구축

9.3 IDEF0 모델 작성 방법

학습 목표

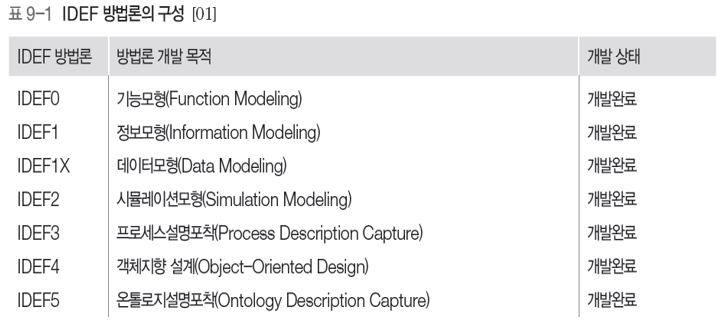
* 통합 분석(설계) (IDEF) 방법론의 등장 배경과 사용 목적, 구성에 대해 학습한다.
* 기능모형 구축 방법론인 IDEF0의 개념을 이해한다.
* IDEF0 모델링 작성 방법을 학습한다.
* IDEF0 모델링 방법을 활용한 작성 사례를 평가하고 개선할 수 있도록 실습 예제를 풀어본다.

**9.1 소프트웨어 공학**

**통합 분석/설계 방법론의 등장 배경과 사용 목적**

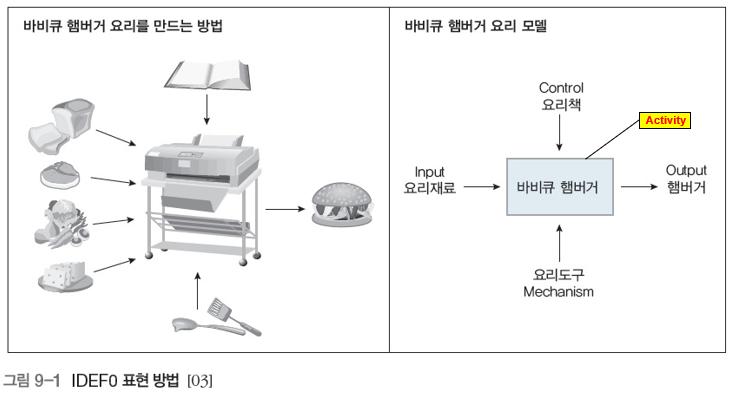
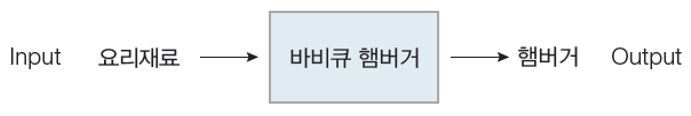
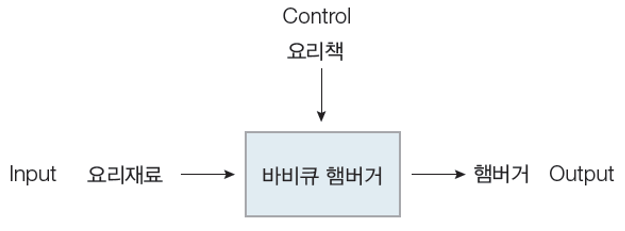
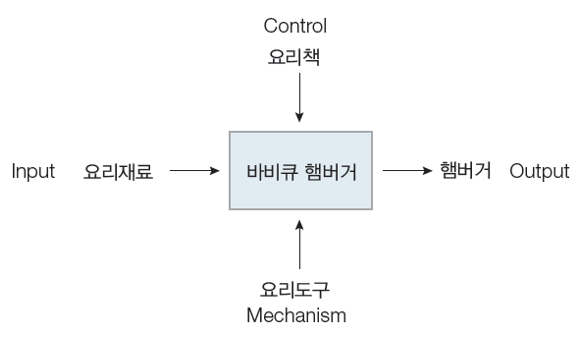
* 통합 분석/설계 방법론
  + 기업과 조직의 실체를 추상화 하여 모델링(AS-IS Model)
  + 작성된 모델의 체계적인 분석을 통해 문제점을 추출하여 개선된 기업의 모델(TO-BE Model)을 설계
* 통합 분석/설계 방법론의 사용 목적
  1. 시스템 분석, 설계, 교육, 문서화, 통합
  2. 협의(Consensus)를 위한 의사소통 수단 지원
  3. 기업의 정보시스템 구축을 위한 업무활동의 분석과 문제점 포착
  4. 기업의 활동에 관한 업무흐름의 명확한 표현

**통합 분석/설계 방법론의 구성 (개발 완료된 것만)**

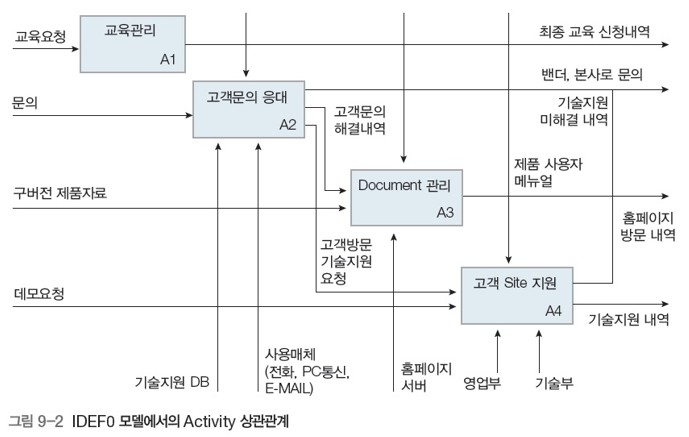


**9.2 기능 모형 구축**

**IDEF0 모델 다이어그램**

* ICOM = Input + Control + Output + Machanism  
  
* 바비큐 햄버거 요리 모델로 보는 IDEF0 표현 방법
  1. Input : 박스 좌측부터 시작하고 Activity(네모박스)에 의해 Output 변환  
     
  2. Output : 박스 우측으로 나오며 Activity의 결과로 산출되는 Concept  
     
  3. Control : 박스의 상부에서 내려가며 Activity의 수행을 통제  
     
  4. Mechanism : 박스의 하부에서부터 올라가며 Activity에 의해 사용되는 사람이나 기계와 같은 자원을 표시  
     

**IDEF0 모델에서의 Activity 상관관계**

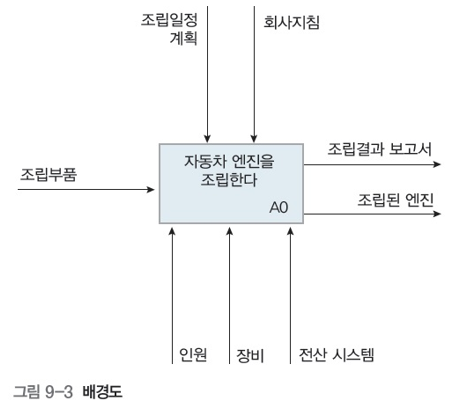
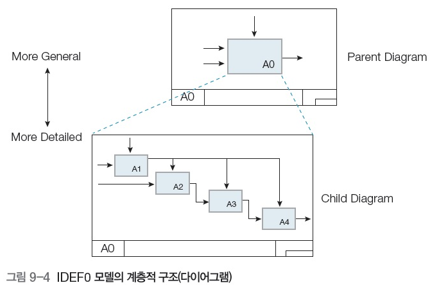
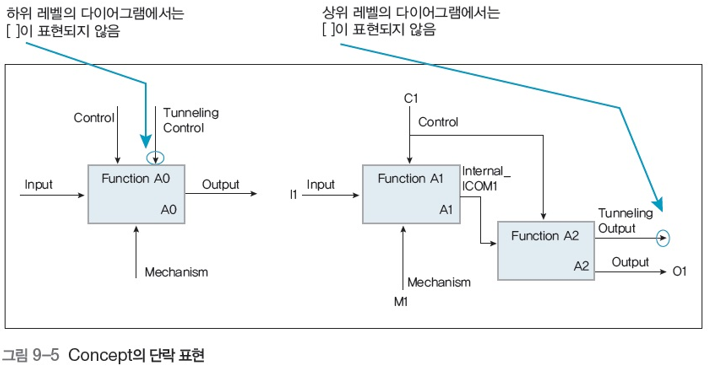


**IDEF0 모델의 특징**

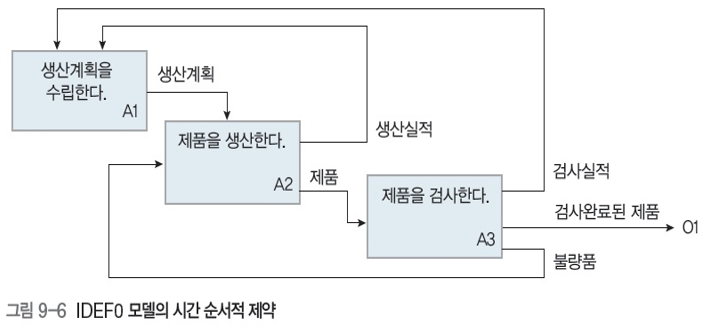
* 기업이나 조직의 활동을 추상적인 단위인 Activity로 표현
* ‘조직’과 ‘기능’을 분리시킴
* 상호협동적 팀 프로젝트에 효과적인 분석 도구로 활용

**9.3 IDEF0 모델 작성 방법**

**IDEF0 모델 표현 방법**

* 기본 표현 방법은 ‘활동’의 수직적 해체라는 개념  
  
* 계층적 구조(Diagram)
  + 하나의 모델에 대한 명백한 경계 설정
  + 목적, 관점, 주변여건 관계 일관성 제공
  + 계층적 수직적 분석방식 지원
  + 추상화 레벨에 대한 개념 제공  
    
* Concept의 단락 표현  
  

**IDEF0 모델 작성 지침**

* 일관된 목적 및 관점의 유지
  + IDEF0 모델화 활동을 시작하기 위해 작업자는 모델의 ‘목적’과 활동의 설명이 정형화 될 수 있는 ‘관점’을 우선적으로 결정해야 함
    - 목적 : 모델화 작업의 목표
    - 관점 : 모델을 구축, 검토, 파악할 경우 취해야 할 기준
* 문법적 제한 및 규약
  + 하나의 Activity가 분해될 수 있는 범위(3~6개)와 하나의 Activity에 연결될 수 있는 화살표의 수적 제한(동일한 Concept 6개 이하)을 미리 설정
  + 모델을 판독하고 그것이 2분 미만의 시간에 가능한지 살펴봐야 함
* IDEF0의 제약사항
  + 활동의 시간 순서적 제약을 명확하게 표현하지 않음  
    

**직원채용 절차에 대한 기능모형 작성 실습**

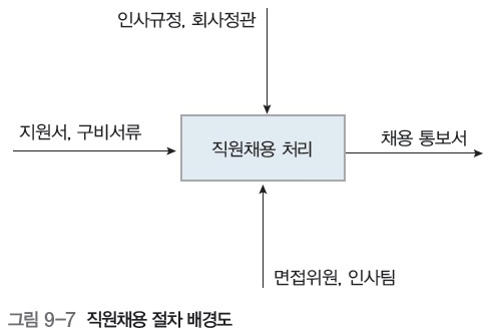
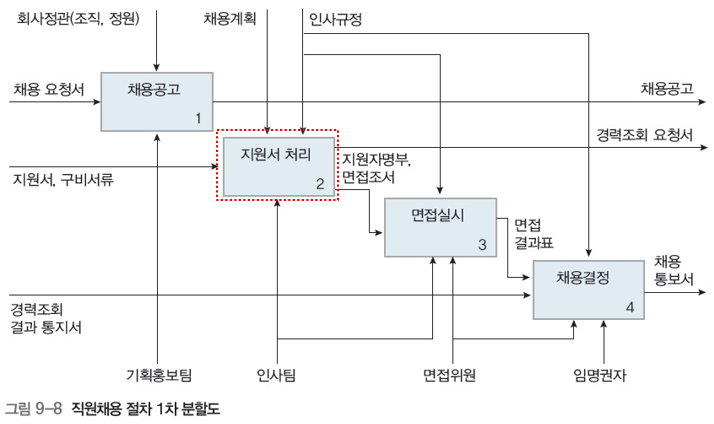
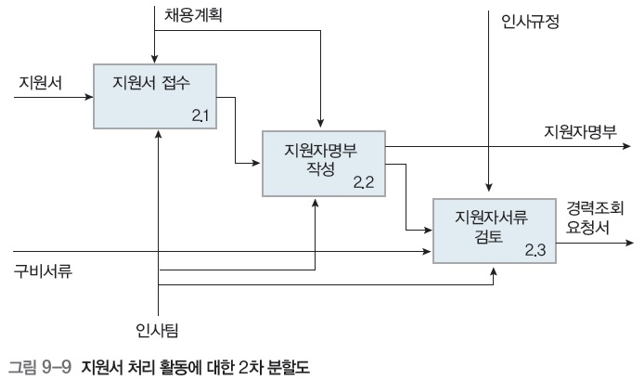
우리 회사의 일반적인 직원채용 절차는 기획홍보팀에서 채용 공고를 작성해 홈페이지에 게시한 후, 인사팀에서 지원서를 접수 받아 지원자명부를 작성한다.

지원서 접수가 종료되면 서류심사를 통해 결격사유가 있는 경우를 제외한 지원자에게 면접 일정을 통보해 면접을 실시한다.

인사담당자는 면접조서를 작성하여 면접관에게 배부하며 면접관은 면접조서의 기재사항을 참조하여 지원자에 대한 면접을 실시한다.

또한 인사담당자는 제출서류와 지원서 기재사항을 검토한 후 경력사항의 진위여부를 파악하기 위해 경력조회를 해당 부서에 요청한 후 결과를 접수하여 면접결과표에 첨부한다.

최종 심사를 거쳐 채용이 결정된 지원자에게는 채용결정 사실을 통보한다.

* 주요 기능과 세부 기능 추출
  1. 채용 공고 : 직원채용 요청서 접수, 직원채용 계획수립, 직원채용 공고
  2. 지원서 처리 : 지원서 접수, 지원자명부 작성, 지원서 기재사항 및 제출서류 검토
  3. 면접실시 : 면접조서 작성, 면접결과표 작성
  4. 채용결정 : 경력자 조회 접수, 종합토의 및 결정, 채용결정 통보
* IDEF0 방법론에 따라 작성  
  
* 직원채용 절차 1차 분할도  
  
* 지원서 처리 활동에 대한 2차 분할도  
  

제 10장 데이터 모형 구축 방법

10.1 정보공학 방법론의 개요

10.2 업무영역 분석

10.3 데이터모형 구축

학습 목표

* 정보공학 측면에서 시스템 구축 과정을 이해한다.
* 시스템 분석 과정으로서의 업무영역 분석 절차를 이해한다.
* 데이터모형 구축의 과정과 방법을 학습한다.
* 데이터모형의 구축 과정을 사례분석을 통해 익힌다.

**10.1 정보공학 방법론의 개요**

**방법론의 출현과 발전과정**

방법론이란 정보시스템을 구축하는 데 필요한 여러 작업 단계들의 ‘수행방법(Method)’과 작업 수행 시 도움이 되는 ‘기법(Technique)’ 및 ‘도구(Tool)’를 이용한 개발 경험을 바탕으로 각 작업 단계를 체계적으로 정리한 작업 수행의 표준규범이라고 정의할 수 있다.

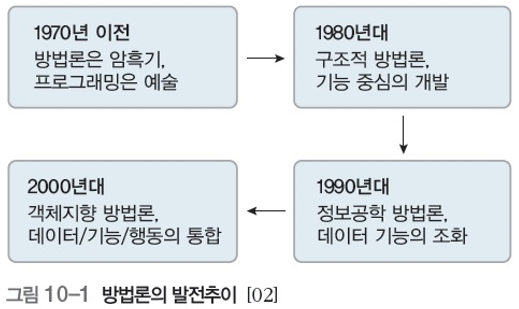
Methodology : Method + Knowledge

Technology : Technique + Knowledge

**방법론이 필요한 이유**

* 작업방법의 표준화
* 커뮤니케이션(Communication) 향상
  + 사용자와 개발자 간의 정보공유
* 정보시스템 품질 수준의 목표달성
  + 탁월하고 독특한 개인의 독창성을 제한할 수 있으므로 주의
* 프로젝트 위험의 최소화
  + 전체 공정에 대한 종합적인 관리가 가능하므로
* 주어진 기간과 비용 내에서 시스템 완성
  + 적절한 공정관리 및 도구의 사용으로 성공률이 높다

**방법론의 발전 추이**

* 기능 중심의 방법론에서 데이터와 기능의 조화를 이루는 쪽으로 발전  
  

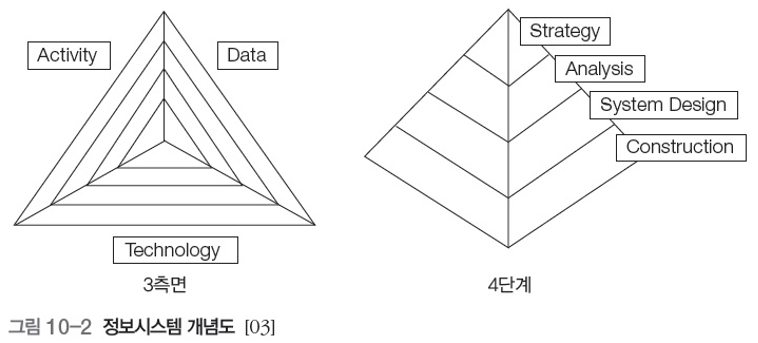
**정보공학 방법론의 개념**

* 기업 전체의 관점에서 기업 활동을 기업모델로 분석
* 이를 다이어그램 형태로 표현
* 정보시스템의 계획, 분석, 설계, 구축 등의 전 과정을 공학적으로 적용하는 방법론.  
  기업모델은 데이터 모형과 업무활동 모형이 균형 있게 고려된 모형을 의미

**정보공학 방법론과 기능 중심 방법론의 차이점**

* 전사차원의 정보체계 지원(Information Architecture)
  + 기능 중심 방법론에 비해 회사 전체를 먼저 고려한 후 하부시스템을 구축
* 경영전략 지원 중심(Business Oriented)
* 데이터와 업무활동(기능)의 균형(More Data Oriented)
  + 업무활동 지원 외에 경영전략 수립에 필요한 정보 제공이 목표 (예, ERP)

**정보시스템 개념도(피라미드형)**



**정보시스템의 3측면**

* Data : 조직이 현재 관리하거나 관리대상이 되는 모든 데이터
* Activity : 데이터를 이용한 조직의 모든 업무수행 활동
* Technology : 정보시스템 구축과 관련되는 모든 실행 기법

**정보시스템의 4단계**

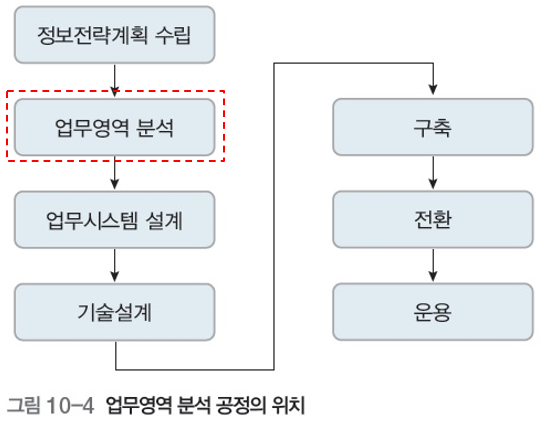
* Strategy : 기업이 필요로 하는 정보에 대한 전략적 비전을 제시하고 전략 계획을 수립
* Analysis : 기업운영에 필요한 논리모형 구축
* Design : 특별한 과정을 처리 및 수행하기 위한 절차 설계
* Construction : 데이터를 이용한 응용 프로그램 단계

**10.2 업무영역 분석(BAA)**

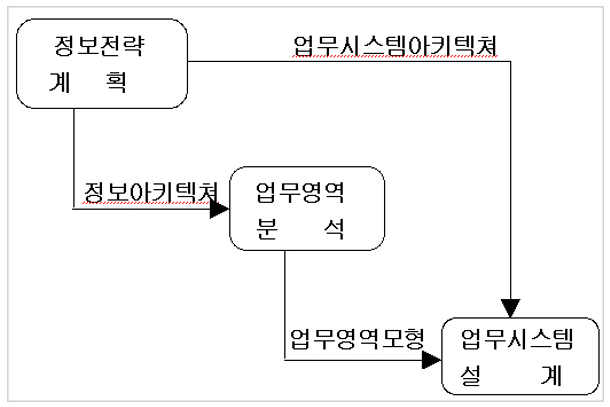
**업무영역 분석의 정의**

* 정보 전략 계획(ISP: Information Strategy Planning) 수립 후,
* 정보체계(IA: Information Architecture)를 업무영역 단위로 인계 받아,
* 상세 업무 처리 과정을 설계하여,
* 업무 시스템 설계(BSD: Business System Design)을 하기 위한 기초를 제공하는 공정  
  

**업무영역 분석 공정의 위치**



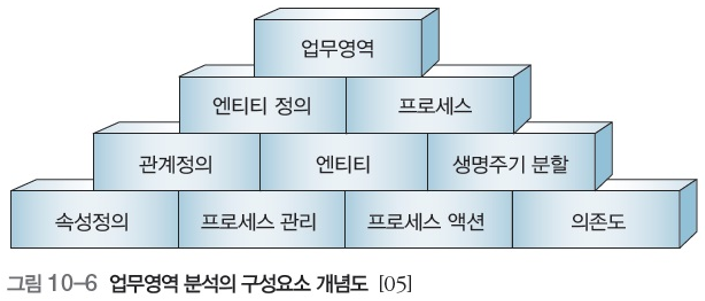
**업무영역 분석 공정 단계의 연계도**



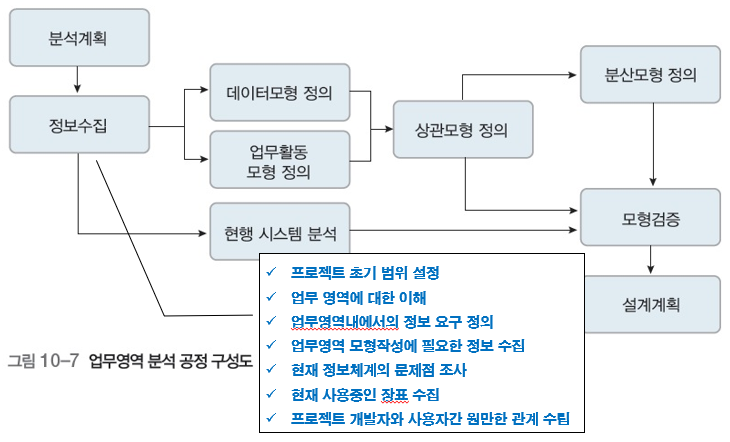
**업무영역 분석의 목적**

* 업무영역에 대한 상세한 이해
  + 정보처리의 발생 및 시스템 분석 대상 영역
* 업무영역에 대한 정보요구 파악 및 우선순위 정의
* 업무영역의 모형화
  + 데이터 모형화 -> DB설계 기초자료
  + 업무활동 모형화 -> 실행 모듈 설계 및 메뉴 구축에 활용
  + 업무활동과 데이터 간의 상관관계 모형화 -> 데이터 변동이 미치는 영향 분석 기초 자료
* 업무시스템 설계 영역 정의 -> 업무 시스템 설계 계획에 활용
* 업무시스템 설계 계획 수립 -> 시스템 설계 과정으로 진행

**업무영역 구성요소에 대한 개념도**



**업무영역 분석의 공정 구성도**



**10.3 데이터모형 구축**

**데이터모형 구축의 개념**

* 기업의 정보구조를 실체(Entity)와 관계(Relationship)를 중심으로 정해진 기호와 규칙을 사용하여 명확하고 체계적으로 표현하고 문서화하는 기법

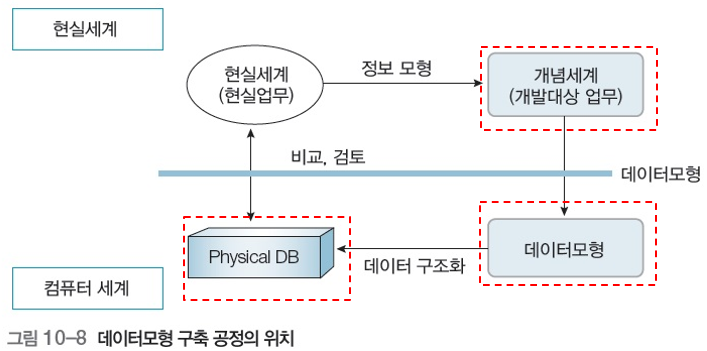
**데이터모형 구축의 목적**

* 연관 조직의 정보 요구에 대한 정확한 이해 제공
* 분석자, 개발자, 사용자 간의 의사소통 수단 제공
* 데이터 중심의 분석 방법 제공
* 변경 및 영향에 대한 분석 제공

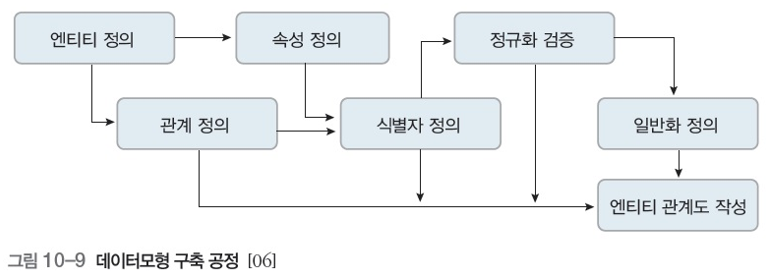
**데이터모형 구축의 종류**

* 논리적 데이터모형 구축(LDM, Logical Data Modeling)
  + 기업모델에서 나온 실체를 구체적, 상세적 정보로 변환 및 일반화하는 과정
* 물리적 데이터모형 구축(PDM, Physical Data Modeling)
  + DBMS의 기능과 성능, 데이터 분산형태를 고려하여 스키마를 생성하는 과정

**데이터모형 구축 공정의 위치**



**데이터모형 구축 공정 다이어그램**



**데이터모형 구축 작업**

* 엔티티 추가
* 엔티티 상세화 : 식별자(Identifier), 속성(Attributes), 일반화(Generalization)
* 관계 상세화 : 기수성(Cardinality), 선택성(Optionality)
* 업무규칙 정의

상세)

Cardinality : 한 릴레이션을 구성하는 튜플의 수

Tuple(관계해석) : RDB에서 관계(표) 내의 속성과 관계되는 값의 집합.

RDB 내에서 관계는 표로 저장되는데, 표의 열이 속성이고 행이 튜플이다.

비관계 파일에서의 Record와 같은 의미.

Generalization(일반화) : 어떤 정보가 필요하고 DB에 담아야 하는지 실데이터를 모형화 하는 과정

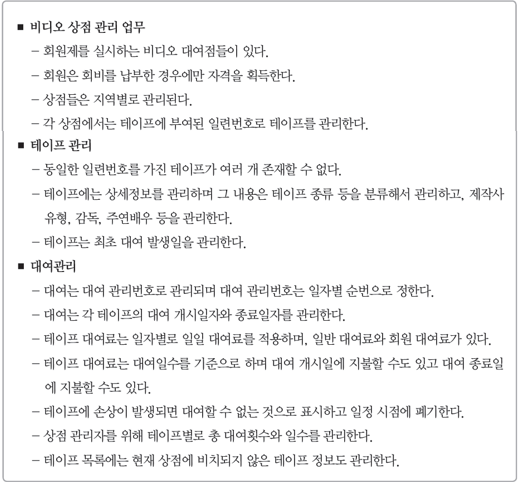
선택성(Optionality)

* 정의 : 관계로 맺어져 있는 하나 또는 두 개의 실체유형 사이의 업무규칙을 나타내는 특성
* 선택성(Optionality) : 페어링(Pairing)의 존재 여부
  + 식별 하나 또는 두 개의 실체유형 간의 관계가 성립될 때 모든 instance 간에 그 짝이 반드시 존재해야 하는지 (Mandatory), 아니면 모든 instance에 대해 그 짝이 존재하지 않아도 되는지(Optional)로 구분된다.
* 선택성의 결정 방법(과거)
  + 필수(Mandatory) : 관계가 반드시 있어야 되는 것 (학생 -> 컴퓨터공학과)
  + 선택(Optional) : 관계가 하나 이상이 있을 수도 있다 (컴퓨터공학과 -> 학생1, 학생2, …)
* 선택성의 결정 방법(현재)
  + 필수(Mandatory) : 다른 실체유형에 어떤 Row를 입력하기 전에 상대 실체유형에도 반드시 하나의 Row가 존재해야 한다. (즉, 1:1 대응)
  + 선택(Optional) : 다른 실체유형에 어떤 Row를 입력하기 전에 상대 실체유형에 어떤 Row가 존재할 필요가 없다.

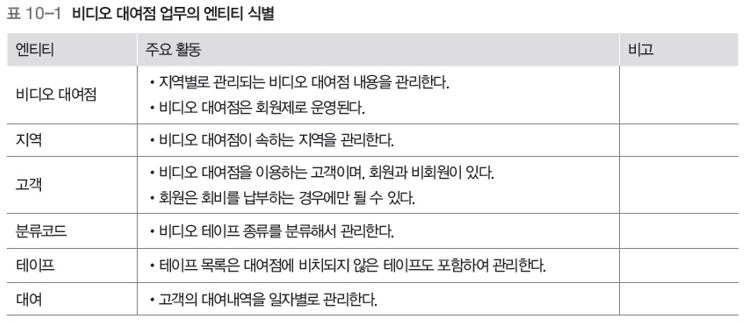
**모델링의 적정성 판단**

* 구조적 확증성 : 데이터 정의와 구성방법의 일관성이 유지되고 있는가?
* 단순성 : 사용자가 이해하기 쉽게 구성되었나?
* 비중복성 : 정보가 한군데에 한번만 존재하는가? (만일 중복되었다면 재검토가 필요)
* 공유성 : 적용업무나 기술에 특화되지 않은 다수에 의해 사용 가능한가?
* 무결성 : 정보를 사용하고 관리하는 방식에 일관성이 있는가?
* 확장성 : 새로운 요구 시 최소의 노력으로 수용 가능한가?

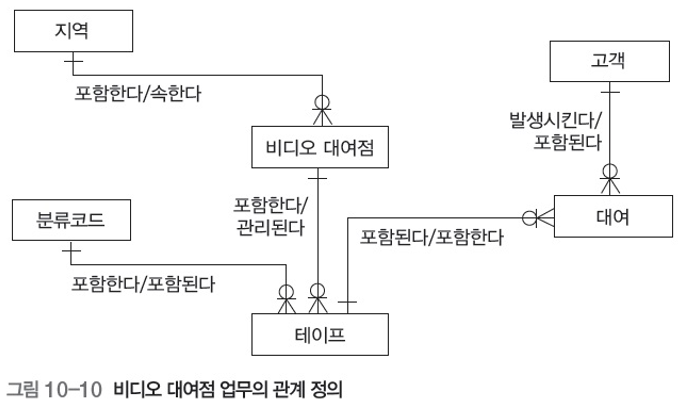
**실습하기 : 비디오 상점 업무의 데이터모형 구축**



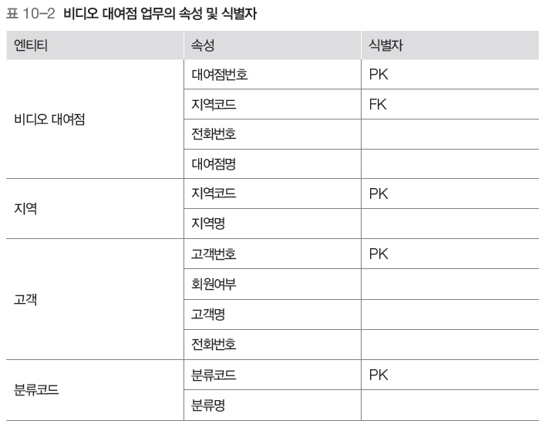
Step-1) 엔티티 식별

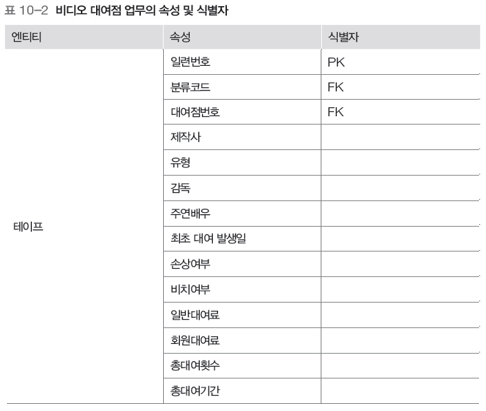


Step-2) 엔티티 간의 관계 정의



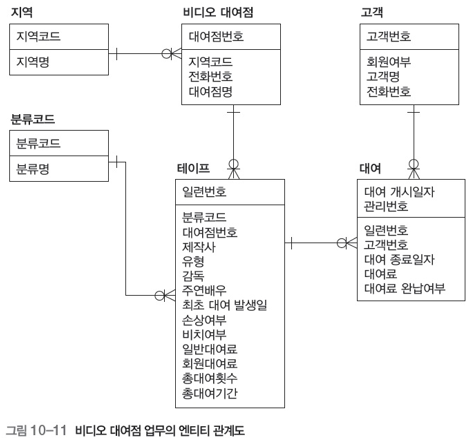
Step-3) 엔티티의 속성 및 식별자 정의







Step-4) 엔티티 관계도 작성



제 11장 관계형 DB 설계과정

11.1 엔티티 정의

11.2 관계 정의

11.3 식별자 정의

11.4 속성 정의

11.5 정규화

학습 목표

* 관계형 DB 설계 과정을 단계별로 학습한다.
* 엔티티 정의 및 관계 정의 방법을 익혀 ER 다이어그램을 작성할 수 있도록 학습한다.
* 식별자 정의 및 속성 정의 과정을 통해 DB 스키마의 설계 방법을 학습한다.
* 정규화의 개념을 이해하고 실제 사례를 통해 적용해 본다.

**11.1 엔티티 정의**

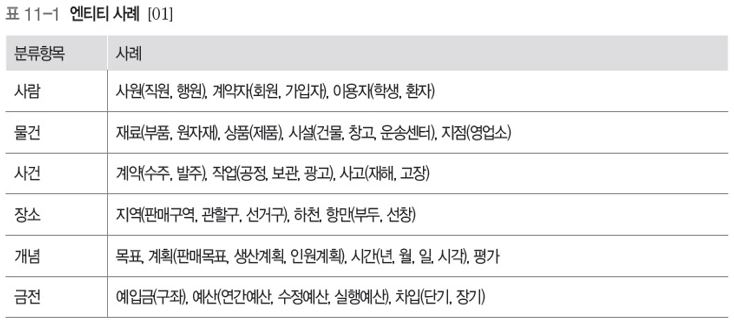
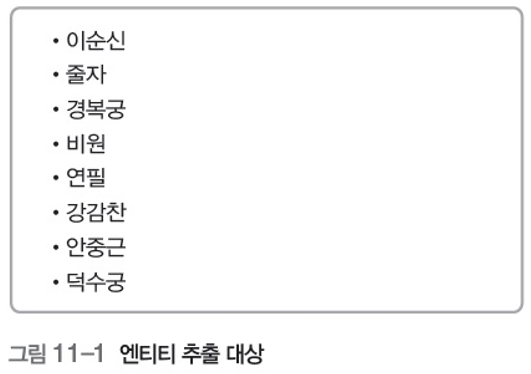
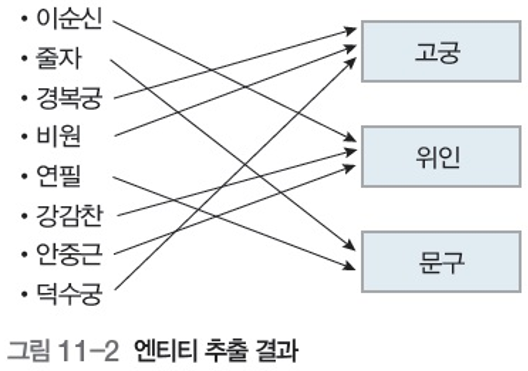
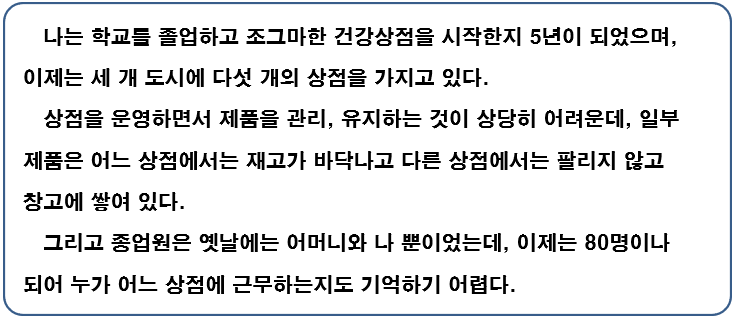
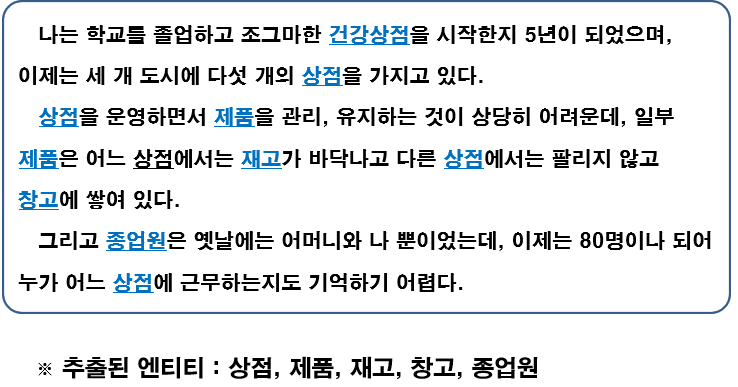
**엔티티의 개념과 조건**

* 엔티티의 정의에 필요한 충족 조건
  + 업무에 유용한 정보를 제공해야 함
  + 명확한 속성 유형이 하나 이상 존재해야 함
  + 각각의 인스턴스를 구분할 수 있어야 함
  + 엔티티는 최소한 하나 이상의 다른 엔티티와 관계를 가져야 함

**엔티티 추출**

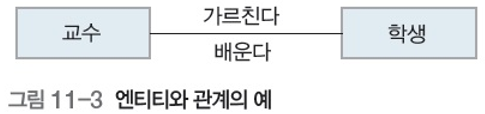
* 엔티티로 추출될 수 있는 대상 데이터의 범위
  + 정보전략계획의 산출물인 전사 데이터
  + 정보수집 공정의 산출물
  + 현행 사용 장표 및 각종 서식
  + 현행 정보시스템의 데이터 구조 분석 결과
* 엔티티 추출 기준
  + 현업에서 사용하는 명사형 어휘를 추출
  + 엔티티 조건의 충족 여부를 기준으로 판단
  + 상상력으로 창조하지 말고 존재 여부를 눈으로 확인한 후 추출
* 엔티티 추출 시 주의사항
  + 상상력을 동원하여 엔티티를 창조하지 않음
  + 존재 여부를 확인한 후 추출
  + 엔티티를 분리 유형
    - 기업에서 발생, 관리되는 자료는 데이터베이스, 그룹웨어, 패키지로 분류 가능
    - 엔티티의 적정성 검증
    - 유용성, 식별자, 속성 존재여부, 엔티티 간의 식별 가능 및 관계 존재 여부

**엔티티 정의**

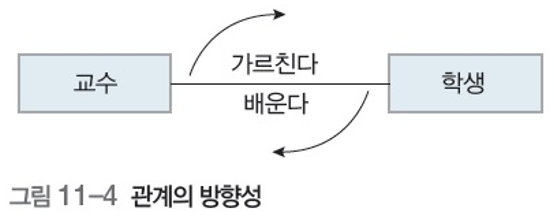
* 엔티티명 정의 시 규칙
  + 현업용어(업무적인 용어)를 사용
  + 단수명사 사용
  + 약어 사용 금지
  + 유일한 명칭 사용
  + 필요 시 수식어 사용 가능
* 엔티티 정의 시 포함해야 할 사항
  + 엔티티명 : 현업 사용자 어휘를 사용하며 모델에서 유일한 엔티티명을 사용
  + 엔티티 종류 설계
    - 독립 엔티티 : 스스로 정보를 제공할 수 있는 엔티티
    - 종속 엔티티 : 다른 독립 엔티티에 종속되어 정보를 제공하는 엔티티
  + 엔티티 정의 : 왜 이러한 정보가 필요한지 식별, 이해도 증진, 중복을 피하기 위해 반드시 정의
* 엔티티 정의 사례
  + 엔티티명 : 고객
  + 정의 : 당사의 상품 및 제품을 과거 3년 이내에 구매한 사실이 있는 개인 또는 법인
  + 종류 : 독립 엔티티
  + 동의어 : 거래처
  + 발생건수 : 5,000건 / 년
  + 성장률 : 20% / 년
* 엔티티의 사례  
  
* 실습하기-1 : 엔티티 추출  
  
* 실습하기-1 : 엔티티 추출 결과  
  
* 실습하기-2 : 엔티티 추출 (1)  
  
* 실습하기-2 : 엔티티 추출 (2)  
  

**11.2 관계 정의**

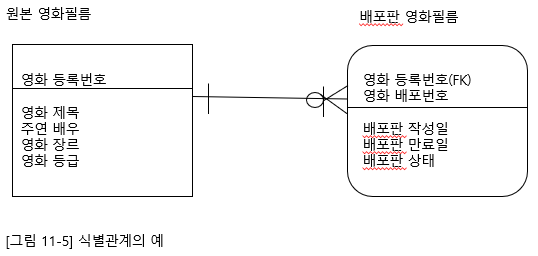
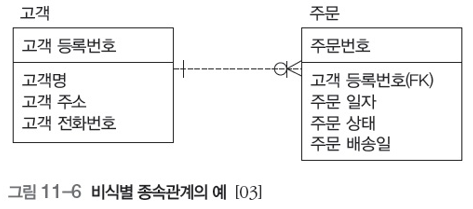
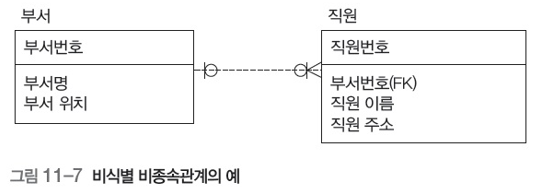
**관계의 개념과 추출**

* 관계
  + 두 엔티티를 연결하고자 하는 업무적인 이유, 업무규칙, 제약 등
  + 두 엔티티 사이에서 논리적인 연결 관계를 의미  
    
* 관계 추출 시 유용한 기준
  + 현업에서 사용하는 동사형 어휘
  + 한 장표에 나타나는 엔티티들

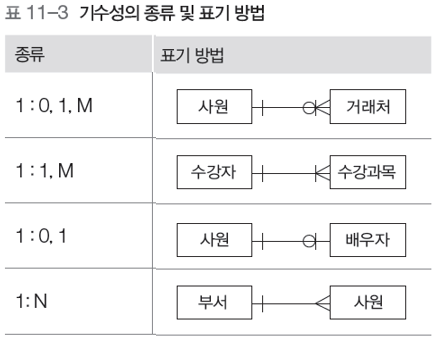
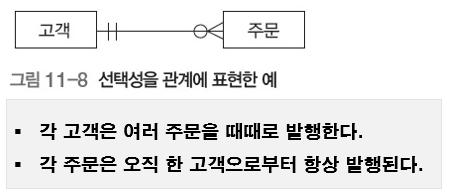
**관계의 방향성, 관계요소, 명명 기준**

* 방향성 : 두 엔티티들 중 주체로 참여하는 엔티티 입장에서 능동형 동사를 사용
* 관계요소 : 하나의 관계는 양방향의 관계요소로 구성  
  
* 명명 기준
  + 두 개의 관계요소명으로 표기
  + 사용자 용어를 사용
  + 현재 시제 동사를 사용
  + 방향성에 따라 능동형 또는 수동형으로 표현
  + 방향성을 고려하여 시계방향으로 표현

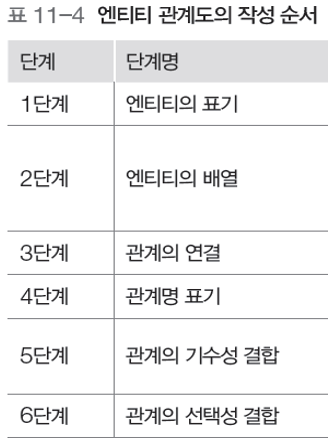
**관계의 종류**

* 식별관계(Identifying Relationship)
  + 부모 엔티티의 주 식별자는 관계를 통해 자식 엔티티로 자동적으로 이주
  + 자식은 자신의 각 인스턴스를 식별하기 위해 부모에 종속적
  + 부모 없이 존재할 수 없음  
    
* 비식별관계(Non-Identifying Relationship)
  + 비식별 종속관계
    - 부모의 주 식별자는 자식의 non-key 영역으로 이주
    - 자식은 자신의 각 인스턴스의 식별을 위해 부모 엔티티에 독립적
    - 부모 없이 존재할 수 없음  
      
  + 비식별 비종속관계
    - 부모의 주 식별자는 자식의 non-key 영역으로 이주
    - 자식은 자신의 각 인스턴스의 식별을 위해 부모 엔티티에 독립적
    - 부모 없이 존재할 수도 있음  
      

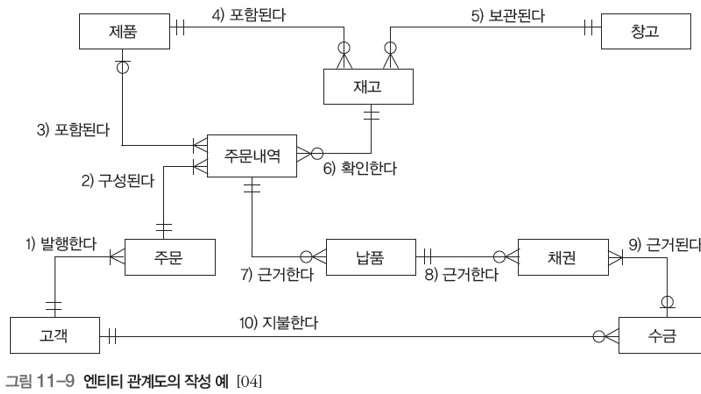
**관계의 속성**

* 기수성 : 관계에 참여하는 각 엔티티가 얼마나 많이 참여할 수 있는가의 관계 비율
* 기수성의 종류
  + 1 : 0, 1, M(one vs. zero, one or more)
  + 1 : 1, M(one vs. one or more)
  + 1 : 0, 1(one vs. zero or one)
  + 1 : N(one vs. numbers)
* 기수성의 표현  
  
* 선택성 : 관계되는 엔티티 존재조건으로 관계연결의 여부가 미치는 영향의 표현 방식
* 선택성의 종류
  + 항상(Always)
  + 때때로(Sometimes)  
    

**엔티티 관계도 작성**

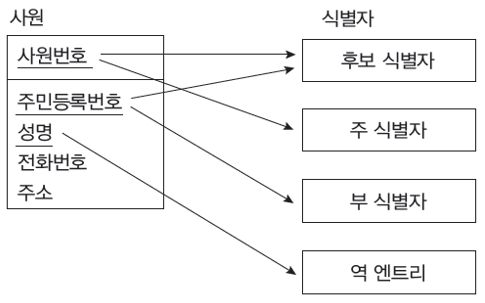
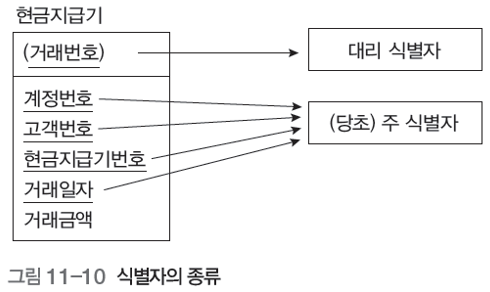


**엔티티 관계도 작성 사례**



**11.3 식별자 정의**

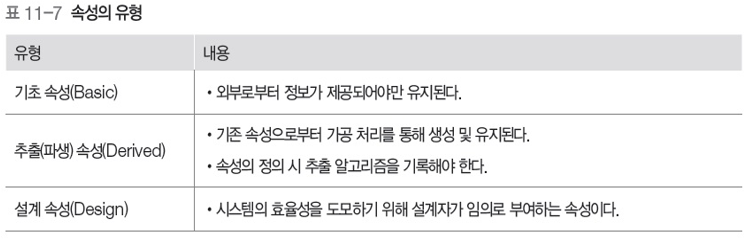
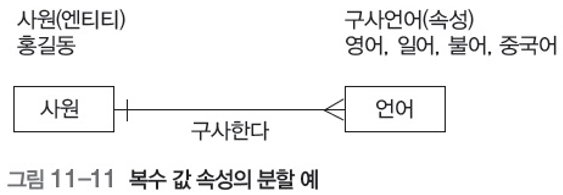
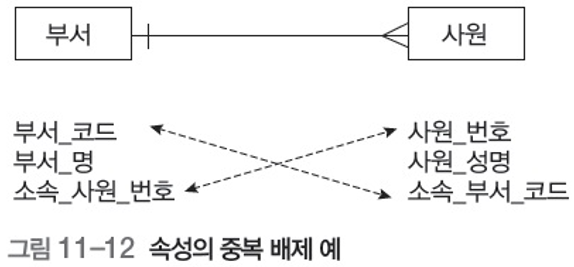
식별자의 개념과 종류

* 식별자의 개념
  + 한 엔티티 내의 특정 인스턴스를 구분할 수 있는 단일속성 또는 속성 그룹
* 식별자 종류
  + 후보 식별자(Candidate key)
  + 주 식별자(Primary key)
  + 부 식별자(Alternate key)
  + 대리 식별자(Surrogate key)
  + 역 엔트리(Inversion key)
* 후보 식별자
  + 엔티티의 각 인스턴스를 유일하게 식별하기 위해 제공되는 속성이나 속성 그룹  
    예) 사원의 후보 식별자 : ‘사원번호’, ‘주민등록번호’
  + 후보 식별자는 다시 주 식별자나 부 식별자로 구분
* 주 식별자
  + 엔티티의 각 인스턴스를 유일하게 식별하는 데 가장 적합한 것
* 주 식별자를 선택할 때 고려할 요소
  + 주 식별자는 효율적이어야 함
  + 주 식별자는 Null 값을 포함할 수 없음
  + 주 식별자는 정적(Static)으로 유지되어야 함
  + 업무적으로 활용도가 높아야 함
* 부 식별자
  + 후보 식별자 가운데 주 식별자로 선택되지 않은 식별자
  + ‘사원번호’를 주 식별자로 선택하면  
    ‘예) 사원 주민등록번호’는 부 식별자가 됨
* 대리 식별자 : 긴 복합 식별자를 주 식별자로 사용할 경우, 이를 대체하는 인위적이고, 단순한 단일 속성
* 역 엔트리
  + 하나 또는 그 이상의 속성이 하나 또는 그 이상의 엔티티의 인스턴스에 접근하는 데 자주 사용될 때 선택
  + 역 엔트리로 선택된 속성이 반드시 유일할 필요는 없음  
    예) ‘사원’을 ‘성명’으로 검색한다면 ‘성명’이라는 속성이 역 엔트리에 해당
* 식별자의 종류 사례 1)  
  
* 식별자의 종류 사례 2)  
  
* 식별자 업무규칙
  + 입력규칙 : 자식 엔티티에 한 건의 인스턴스가 입력 시 적용하는 업무규칙
  + 삭제규칙 : 부모 엔티티에 한 건의 인스턴스가 삭제 시 적용하는 업무규칙

**속성 정의**

* 속성의 기본 개념
  + 엔티티의 특징을 나타내기 위한 요소
  + 식별자 역할을 하는 Key 영역의 속성과 Key를 제외한 정보를 담고 있는 Non-Key 영역의 속성으로 ㄱ분
* 속성의 도출
  + ‘그것(엔티티)에 대해 무엇을 알고 싶습니까?’ 라는 질문에 대한 답을 찾음
  + 수집된 사용 장표의 항목들을 찾음
  + 기존 시스템의 항목들을 분석
* 속성의 명명
  + 현업의 표준용어를 사용
  + ‘엔티티명\_수식어(필요시)\_영역명’과 같이 명명
  + 소유격 사용을 배제
  + 약어 사용을 배제
  + 핵심 단어로 구성
  + 모델에서 유일한 속성 이름을 갖도록 명명
* 속성 영역(도메인)
  + 속성이 가질 수 있는 값의 범위
  + 명(또는 이름), 주소
  + 번호, 금액, 길이, 무게, 속도, 부피, 수량, 비율
  + 일자, 시각, 기간, 요일

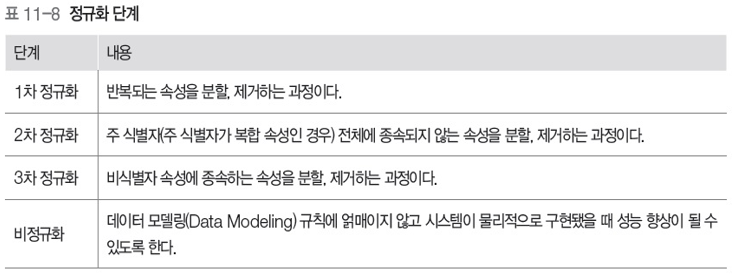
**속성의 정의 방법**

* 속성의 유형  
  
* 복수 값 속성의 분할
  + 특정 엔티티에 대한 속성 값이 둘 이상 나타나는 경우 이 속성을 분할
  + 다른 엔티티로 독립시킴  
    
* 속성의 중복 배제
  + 속성은 모형 내에서 오직 하나의 엔티티에 속해야 함  
    

**정규화(Normalization)**

* 정규화란?
  + 복잡한 데이터 구조를 단순화시켜 안정적인 구조로 변환하는 과정
  + 함수적 종속성(A 속성값에 대해 오직 B 속성 하나의 값만 관련되는 성질)이라는 이론에 근거
* 정규화의 목적
  + 정보의 중복을 최소화
  + 정보의 일관성 확보, 무결성 극대화
  + 정보구조의 안정성 최대화
  + 정보모형의 단순화
  + 정보의 신뢰도 증대

**정규화 단계**

* 정규화 단계별 주요 기능 및 목적  
  
* 제 1 정규형
  + 자료저장소의 모든 속성들이 원자값(더 이상 분할되지 않는 값)을 가짐  
    
* 제 1 정규형의 문제
  + 삭제이상, 삽입이상, 갱신이상
    - 아직 제 1 정규형이 논리화된 상태가 아니기 때문에 발생
* 제 2 정규형
  + 자료저장소가 제 1 정규형
  + 모든 속성들이 주 식별자에 대해 완전 함수적 종속성을 가짐
* 제 2 정규형의 문제
  + 삭제이상 : 주 식별자가 아닌 속성이 주 식별자에 대해 이행적으로 종속성을 갖기 때문
* 제 3 정규형
  + 자료저장소가 제 2 정규형
  + 모든 속성들이 주 식별자에 대해 비이행적으로 함수적 종속성을 가짐
* 비정규형
  + 데이터 모델링 규칙에 얽매이지 않고 성능 향상을 목적으로 구축