학습 목표

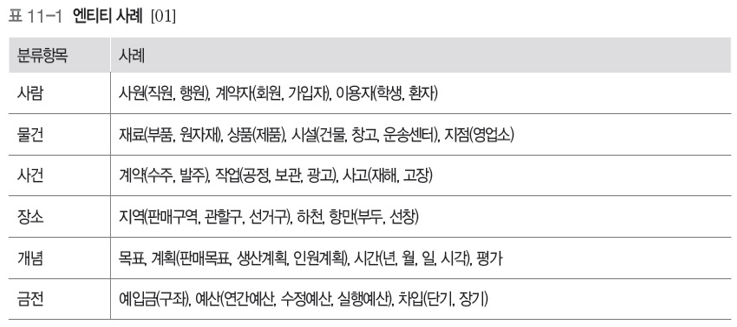
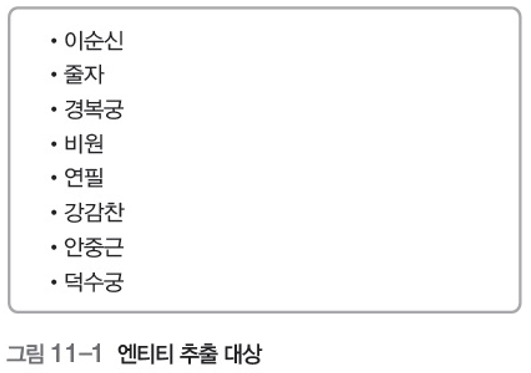
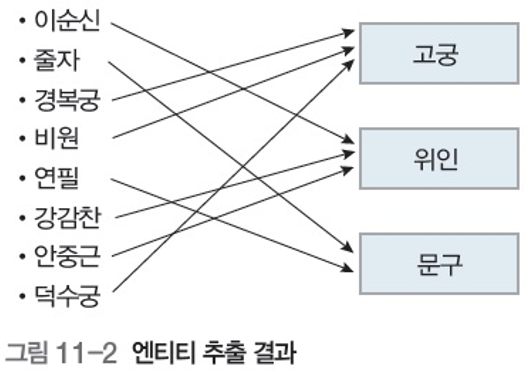
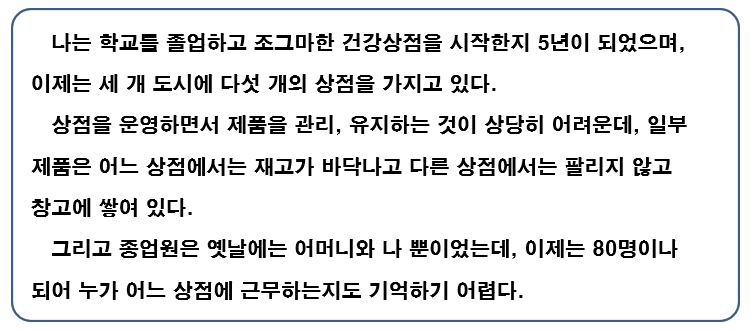
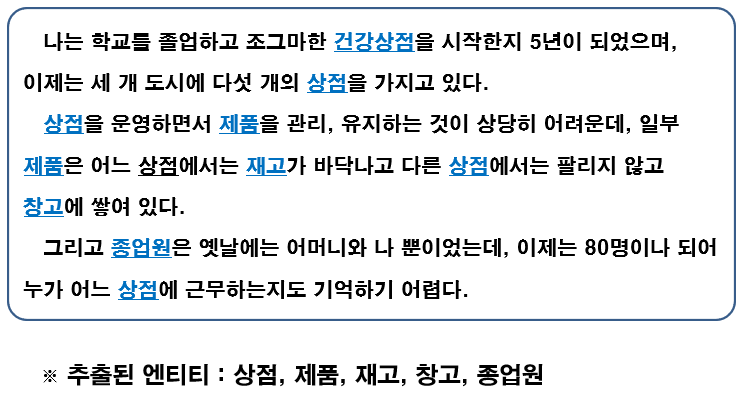
관계형 DB 설계 과정을 단계별로 학습한다.

엔티티 정의 및 관계 정의 방법을 익혀 ER 다이어그램을 작성할 수 있도록 학습한다.

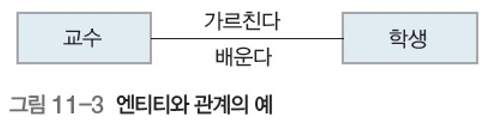
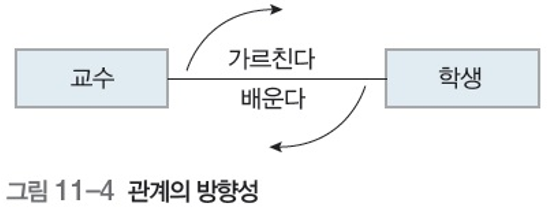
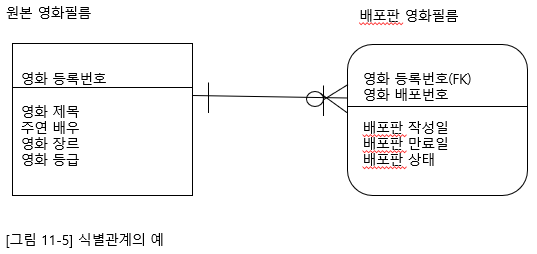
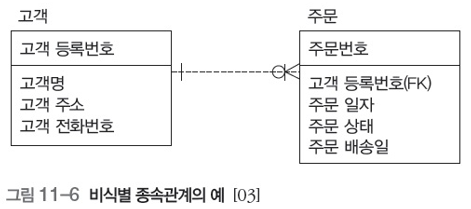
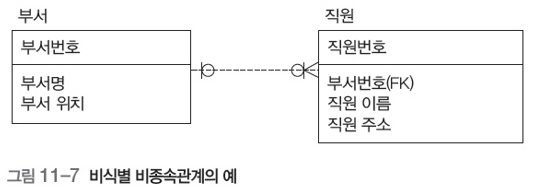
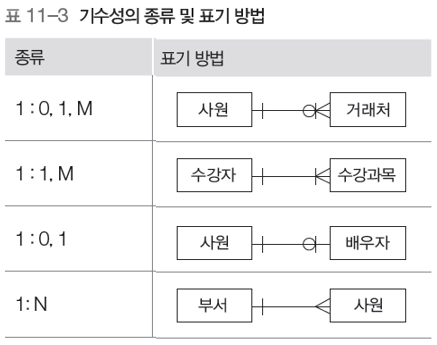
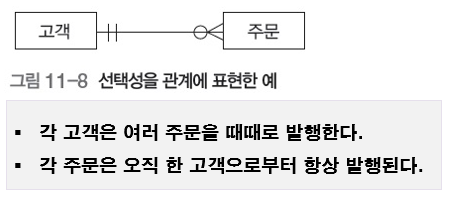
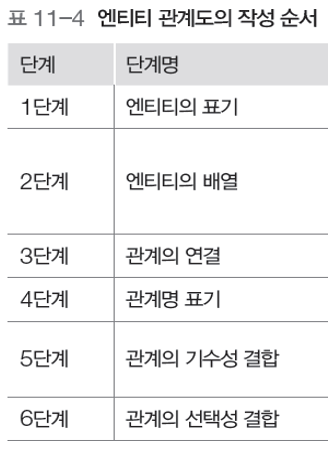
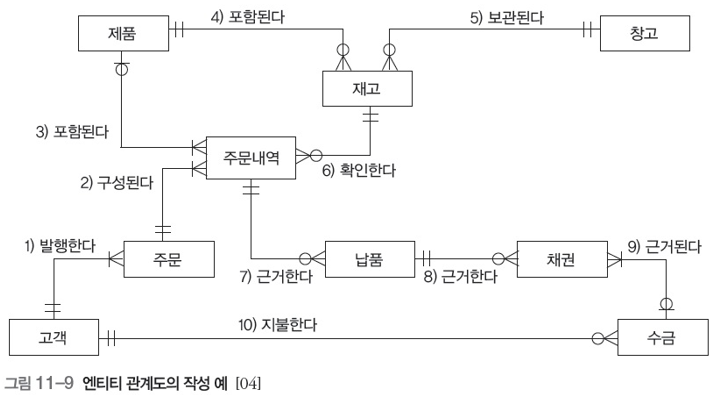
식별자 정의 및 속성 정의 과정을 통해 DB 스키마의 설계 방법을 학습한다.

정규화의 개념을 이해하고 실제 사례를 통해 적용해 본다.

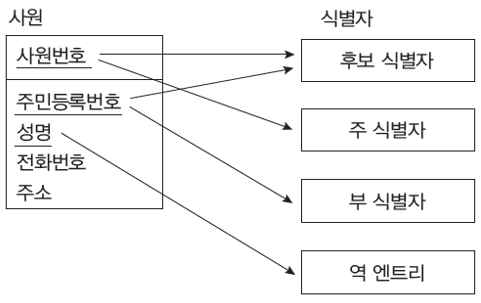
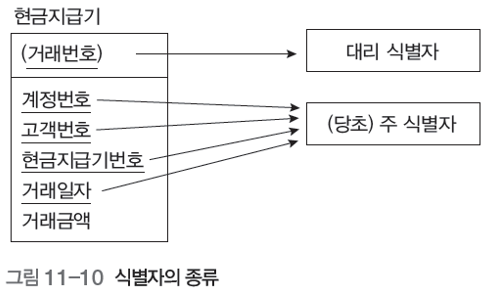
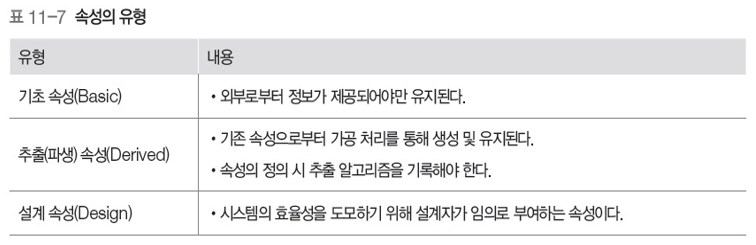
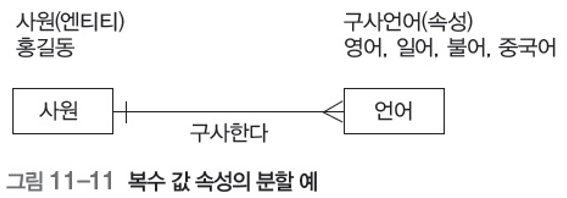
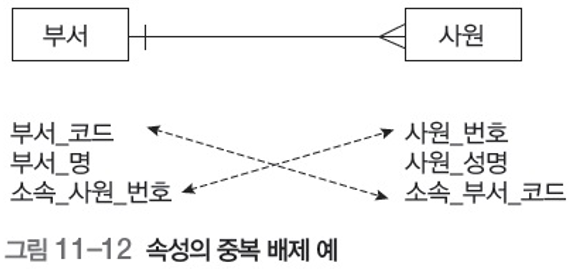
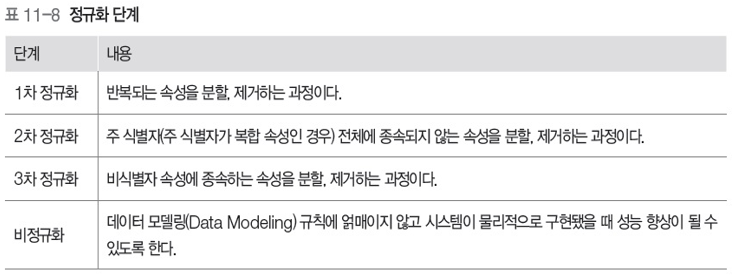
**11.1 엔티티 정의**

1. 엔티티의 개념과 조건
   * 엔티티의 정의에 필요한 충족 조건
     1. 업무에 유용한 정보를 제공해야 함
     2. 명확한 속성 유형이 하나 이상 존재해야 함
     3. 각각의 인스턴스를 구분할 수 있어야 함
     4. 엔티티는 최소한 하나 이상의 다른 엔티티와 관계를 가져야 함
2. 엔티티 추출
   * 엔티티로 추출될 수 있는 대상 데이터의 범위
     1. 정보전략계획의 산출물인 전사 데이터
     2. 정보수집 공정의 산출물
     3. 현행 사용 장표 및 각종 서식
     4. 현행 정보시스템의 데이터 구조 분석 결과
   * 엔티티 추출 기준
     1. 현업에서 사용하는 명사형 어휘를 추출
     2. 엔티티 조건의 충족 여부를 기준으로 판단
     3. 상상력으로 창조하지 말고 존재 여부를 눈으로 확인한 후 추출
   * 엔티티 추출 시 주의사항
     1. 상상력을 동원하여 엔티티를 창조하지 않음
     2. 존재 여부를 확인한 후 추출
     3. 엔티티를 분리 유형
        1. 기업에서 발생, 관리되는 자료는 데이터베이스, 그룹웨어, 패키지로 분류 가능
        2. 엔티티의 적정성 검증
        3. 유용성, 식별자, 속성 존재여부, 엔티티 간의 식별 가능 및 관계 존재여부 등
3. 엔티티 정의
   * 엔티티명 정의 시 규칙
     1. 현업용어(업무적인 용어)를 사용
     2. 단수명사 사용
     3. 약어 사용 금지
     4. 유일한 명칭 사용
     5. 필요 시 수식어 사용 가능
   * 엔티티 정의 시 포함해야 할 사항
     1. 엔티티명: 현업 사용자 어휘를 사용하며 모델에서 유일한 엔티티명을 사용
     2. 엔티티 종류 설계
        1. 독립 엔티티: 스스로 정보를 제공할 수 있는 엔티티
        2. 종속 엔티티: 다른 독립 엔티티에 종속되어 정보를 제공하는 엔티티
     3. 엔티티 정의: 왜 이러한 정보가 필요한지 식별, 이해도 증진, 중복을 피하기 위해 반드시 정의
   * 엔티티 정의 사례
     1. 엔티티명: 고객
     2. 정의: 당사의 상품 및 제품을 과거 3년 이내에 구매한 사실이 있는 개인 또는 법인
     3. 종류: 독립 엔티티
     4. 동의어: 거래처
     5. 발생건수: 5,000건/년
     6. 성장률: 20%/년
   * 엔티티의 사례  
     
   * 실습하기-1: 엔티티 추출  
     
   * 실습하기-1: 엔티티 추출 결과  
     
   * 실습하기-1: 엔티티 추출 (1)  
     
   * 실습하기-2: 엔티티 추출 (2)  
     

**11.2 관계 정의**

1. 관계의 개념과 추출
   * 관계
     1. 두 엔티티를 연결하고자 하는 업무적인 이유, 업무규칙, 제약 등
     2. 두 엔티티 사이에서 논리적인 연결 관계를 의미  
        
   * 관계 추출 시 유용한 기준
     1. 현업에서 사용하는 동사형 어휘
     2. 한 장표에 나타나는 엔티티들
2. 관계의 방향성, 관계요소, 명명 기준
   * 방향성: 두 엔티티들 중 주체로 참여하는 엔티티 입장에서 능동형 동사를 사용
   * 관계요소: 하나의 관계는 양방향의 관계요소로 구성  
     
   * 명명 기준
     1. 두 개의 관계요소명으로 표기
     2. 사용자 용어를 사용
     3. 현재 시제 동사를 사용
     4. 방향성에 따라 능동형 또는 수동형으로 표현
     5. 방향성을 고려하여 시계방향으로 표현
3. 관계의 종류
   * 식별관계(Identifying Relationship)
     1. 부모 엔티티의 주 식별자는 관계를 통해 자식 엔티티로 자동적으로 이주
     2. 자식은 자신의 각 인스턴스를 식별하기 위해 부모에 종속적
     3. 부모 없이 존재할 수 없음  
        
   * 비식별관계(Non-Identifying Relationship)
     1. 비식별 종속관계
        1. 부모의 주 식별자는 자식의 non-key 영역으로 이주
        2. 자식은 자식의 각 인스턴스의 식별을 위해 부모 엔티티에 독립적
        3. 부모 없이 존재할 수 없음  
           
     2. 비식별 비종속관계
        1. 부모의 주 식별자는 자식의 non-key 영역으로 이주
        2. 자식은 자신의 각 인스턴스의 식별을 위해 부모 엔티티에 독립적
        3. 부모 없이 존재할 수도 있음  
           
4. 관계의 속성
   * 기수성: 관계에 참여하는 각 엔티티가 얼마나 많이 참여할 수 있는가의 관계 비율
   * 기수성의 종류
     1. 1 : 0, 1, M(one vs. zero, one or more)
     2. 1 : 1, M(one vs. one or more)
     3. 1 : 0, 1(one vs. zero or one)
     4. 1 : N(one vs. numbers)
   * 기수성의 표현  
     
   * 선택성: 관계되는 엔티티 존재조건으로 관계연결의 여부가 미치는 영향의 표현 방식
   * 선택성의 종류
     1. 항상(Always)
     2. 때때로(Sometimes)  
        
   * 엔티티 관계도 작성  
     
   * 엔티티 관계도 작성 사례  
     

**11.3 식별자 정의**

1. 식별자의 개념과 종류
   * 식별자의 개념
     1. 한 엔티티 내의 특정 인스턴스를 구분할 수 있는 단일속성 또는 속성 그룹
   * 식별자 종류
     1. 후보 식별자(Candidate key)
     2. 주 식별자(Primary key)
     3. 부 식별자(Alternate key)
     4. 대리 식별자(Surrogate key)
     5. 역 엔트리(Inversion key)
   * 후보 식별자
     1. 엔티티의 각 인스턴스를 유일하게 식별하기 위해 제공되는 속성이나 속성 그룹  
        예) 사원의 후보 식별자: ‘사원번호’, ‘주민등록번호’
     2. 후보 식별자는 다시 주 식별자나 부 식별자로 구분
   * 주 식별자
     1. 엔티티의 각 인스턴스를 유일하게 식별하는 데 가장 적합한 것
   * 주 식별자를 선택할 때 고려할 요소
     1. 주 식별자는 효율적이어야 함
     2. 주 식별자는 Null 값을 포함할 수 없음
     3. 주 식별자는 정적(Static)으로 유지되어야 함
     4. 업무적으로 활용도가 높아야 함
   * 부 식별자
     1. 후보 식별자 가운데 주 식별자로 선택되지 않은 식별자
     2. ‘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
     1. 하나 또는 그 이상의 속성이 하나 또는 그 이상의 엔티티의 인스턴스에 접근하는 데 자주 사용될 때 선택
     2. 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
     1. 입력규칙: 자식 엔티티에 한 건의 인스턴스가 입력 시 적용하는 업무규칙
     2. 삭제규칙: 부모 엔티티에 한 건의 인스턴스가 삭제 시 적용하는 업무규칙
2. 속성 정의
   * 속성의 기본 개념
     1. 엔티티의 특징을 나타내기 위한 요소
     2. 식별자 역할을 하는 Key 영역의 속성과 Key를 제외한 정보를 담고 있는 Non-Key 영역의 속성으로 구분
   * 속성의 도출
     1. ‘그것(엔티티)에 대해 무엇을 알고 싶습니까?’ 라는 질문에 대한 답을 찾음
     2. 수집된 사용 장표의 항목들을 찾음
     3. 기존 시스템의 항목들을 분석
   * 속성의 명명
     1. 현업의 표준용어를 사용
     2. ‘엔티티명\_수식어(필요시)\_영역명’과 같이 명명
     3. 소유격 사용을 배제
     4. 약어 사용을 배제
     5. 핵심 단어로 구성
     6. 모델에서 유일한 속성 이름을 갖도록 명명
   * 속성 영역(도메인)
     1. 속성이 가질 수 있는 값의 범위
     2. 명(또는 이름), 주소
     3. 번호, 금액, 길이, 무게, 속도 부피, 수량, 비율
     4. 일자, 시각, 기간, 요일
3. 속성의 정의 방법
   * 속성의 유형  
     
   * 복수 값 속성의 분할
     1. 특정 엔티티에 대한 속성 값이 둘 이상 나타나는 경우 이 속성을 분할
     2. 다른 엔티티로 독립시킴  
        
   * 속성의 중복 배제
     1. 속성은 모형 내에서 오직 하나의 엔티티에 속해야 함  
        
4. 정규화(Normalization)
   * 정규화란?
     1. 복잡한 데이터 구조를 단순화시켜 안정적인 구조로 변환하는 과정
     2. 함수적 종속성(A 속성값에 대해 오직 B 속성 하나의 값만 관련되는 성질)이라는 이론에 근거
   * 정규화의 목적
     1. 정보의 중복을 최소화
     2. 정보의 일관성 확보, 무결성 극대화
     3. 정보구조의 안정성 최대화
     4. 정보모형의 단순화
     5. 정보의 신뢰도 증대
   * 정규화 단계별 주요 기능 및 목적  
     
   * 제 1 정규형
     1. 자료저장소의 모든 속성들이 원자값(더 이상 분할되지 않는 값)을 가짐  
        
   * 제 1 정규형의 문제
     1. 삭제이상, 삽입이상, 갱신이상
        1. 아직 제 1 정규형이 논리화된 상태가 아니기 때문에 발생
   * 제 2 정규형
     1. 자료저장소가 제 1 정규형
     2. 모든 속성들이 주 식별자에 대해 완전 함수적 종속성을 가짐  
        
   * 제 2 정규형의 문제
     1. 삭제이상: 주 식별자가 아닌 속성이 주 식별자에 대해 이행적으로 종속성을 갖기 때문
   * 제 3 정규형
     1. 자료저장소가 제 2 정규형
     2. 모든 속성들이 주 식별자에 대해 비이행적으로 함수적 종속성을 가짐
   * 비정규형
     1. 데이터 모델링 규칙에 얽매이지 않고 성능 향상을 목적으로 구축