2장 데이터 입출력 구현

029 데이터베이스 개요

- 데이터 저장소 – 데이터들을 논리적인 구조로 조직화하거나 물리적인 공간에 구축한 것을 의미

논리 데이터 저장소 – 데이터 및 데이터 간의 연관성, 제약조건을 식별하여 논리적인 구조로 조직화한 것

물리 데이터 저장소 – 논리 데이터 저장소를 소프트웨어가 운용될 환경의 물리적 특성을 고려하여 실제 저장장치에 저장한 것

데이터베이스 – 여러 사람에 의해 공동으로 사용될 데이터를 중복을 배제하여 통합하고 저장장치에 저장하여 항상 사용할 수 있도록 운영하는 운영 데이터

통합된 데이터 / 저장된 데이터 / 운영 데이터 / 공용 데이터

DBMS – 사용자의 요구에 따라 정보를 생성해주고 데이터베이스를 관리해주는 소프트웨어

DBMS 필수 기능 - 정의 기능 / 조작 기능 / 제어 기능

데이터의 독립성 - 논리적 독립성 / 물리적 독립성

스키마 – 데이터베이스의 구조와 제약조건에 관한 전반적인 명세를 기술한 것

외부 스키마 / 개념 스키마 / 내부 스키마

030 데이터베이스 설계

고려사항 – 무결성 / 일관성 / 회복 / 보안 / 효율성 / 데이터베이스 확장

설계 순서 : 요구 조건 분석 -> 개념적 설계 -> 논리적 설계 -> 물리적 설계 -> 구현

요구 조건 분석 – 데이터베이스를 사용할 사람들로부터 필요한 용도를 파악하는 것

개념적 설계 – 정보의 구조를 얻기 위해 현실 세계의 무한성과 계속성을 이해하고 현실 세계에 대한 인식을 추상적 개념으로 표현하는 과정

논리적 설계 – 현실 세계에서 발생하는 자료를 컴퓨터가 이해하고 처리할 수 있는 물리적 저장장치에 저장할 수 있도록 변환하기 위해 특정 DBMS가 지원하는 논리적 자료 구조로 변환시키는 과정

물리적 설계 – 논리적 구조로 표현된 데이터를 디스크 등의 물리적 저장장치에 저장할 수 있는 물리적 구조의 데이터로 변환하는 과정

데이터베이스 구현 – 논리적 설계와 물리적 설계에서 도출된 데이터베이스 스키마를 파일로 생성하는 과정

031 데이터 모델의 개념

- 현실 세계의 정보들을 컴퓨터에 표현하기 위해서 단순화, 추상화하여 체계적으로 표현한 개념적 모형

데이터 모델 구성 요소 : 개체, 속성, 관계

데이터 모델 종류 : 개념적 데이터 모델, 논리적 데이터 모델, 물리적 데이터 모델

데이터 모델에 표시할 요소 : 구조, 연산, 제약 조건

개념적 데이터 모델 – 현실 세계에 대한 인간의 이해를 돕기 위해 현실 세계에 대한 인식을 추상적 개념으로 표현하는 과정

논리적 데이터 모델 – 개념적 모델링 과정에서 얻은 개념적 구조를 컴퓨터가 이해하고 처리할 수 있는 컴퓨터 세계의 환경에 맞도록 변환하는 과정

데이터 모델에 표시할 요소 : 구조 / 연산 / 제약 조건

032 데이터 모델의 구성 요소

개체 – 데이터베이스에 표현하려는 것, 사람이 생각하는 개념이나 정보 단위 같은 현실 세계의 대상체

속성 – 데이터베이스를 구성하는 가장 작은 논리적 단위

속성의 특성에 따른 분류 – 기본 속성 / 설계 속성 / 파생 속성

속성의 개체 구성 방식에 따른 분류 – 기본키 속성 / 외래키 속성 / 일반 속성

관계 – 개체와 개체 사이의 논리적인 연결

관계의 형태 – 일 대 일 / 일 대 다 / 다 대 다

관계의 종류 – 종속 관계 / 중복 관계 / 재귀 관계 / 베타 관계

033 E-R(개체-관계) 모델

- 개체와 개체 간이 관계를 기본 요소로 이용하여 현실 세계의 무질서한 데이터를 개념적인 논리 데이터로 표현하기 위한 방법

데이터를 개체, 관계, 속성으로 묘사

E-R 다이어그램 : 사각형 / 마름모 / 타원 / 이중 타원 / 밑줄 타원 / 복수 타원 / 관계 / 선, 링크

034 관계형 데이터베이스의 구조 / 관계형 데이터 모델

관계형 데이터베이스 - 2차원적인 표를 이용해서 데이터 상호 관계를 정의하는 데이터베이스

튜플 : 릴레이션을 구성하는 각각의 행

속성 : 데이터베이스를 구성하는 가장 작은 논리적 단위

도메인 : 하나의 애트리뷰트가 취할 수 있는 같은 타입의 원자값들의 집합

관계형 데이터 모델 - 2차원적인 표를 이용해서 데이터 상호 관계를 정의하는 DB구조를 말함

035 관계형 데이터베이스의 제약 조건 – 키

키 – 데이터베이스에서 조건에 만족하는 튜플을 찾거나 순서대로 정렬할 때 기준이 되는 속성

키의 종류 : 후보키 / 기본키 / 대체키 / 슈퍼키 / 외래키

후보키 – 릴레이션 구성하는 속성들 중에서 튜플을 유일하게 식별하기 위해 사용되는 속성들의 부분집합 / 유일성, 최소성

기본키 – 후보키 중에서 특별히 선정된 주기

대체키 – 후보키가 둘 중 이상일 때 키본키를 제외한 나머지 후보키를 의미

슈퍼키 – 한 릴레이션 내에 있는 속성들의 집합으로 구성된 키

외래키 – 다른 릴레이션의 기본키를 참조하는 속성 또는 속성들의 집합

036 관계형 데이터베이스의 제약 조건 – 무결성

무결성 – 데이터베이스에 저장된 데이터 값과 그것이 표현하는 현실 세계의 실제값이 일치하는 정확성을 의미

종류 – 개체 / 참조 / 도메인 / 사용자 정의 / NULL / 고유 / 키 / 관계

037 관계대수 및 관계해석

관계대수 – 원하는 정보와 그 정보를 검색하기 위해 어떻게 유도하는가를 기술하는 절차적인 언어

순수 관계 연산자 – Select / Project / Join / Division

일반 집합 연산자 – 합집합 / 교집합 / 차집합 / 교차곱

관계해석 – 관계 데이터의 연산을 표현하는 방법

038 이상 / 함수적 종속

이상 – 테이블이란 일부 속성들의 종속으로 인해 데이터의 중복이 발생하고 이 중복으로 인해 테이블 조작 시 문제가 발생하는 현상을 의미

삽입 이상 – 테이블에 데이터를 삽입할 때 의도와는 상관없이 원하지 않는 값들로 인해 삽입할 수 없게 되는 현상

삭제 이상 – 테이블에서 한 튜플을 삭제할 때 의도와는 상관없는 값들도 함께 삭제되는 현상

갱신 이상 – 테이블에서 튜플에 있는 속성 값을 갱신할 대 일부 튜플의 정보만 갱신되어 정보에 불일치성이 생기는 현상

함수적 종속 – 속성 X의 값 각각에 대하여 시간에 관계없이 항상 속성 Y의 값이 오직 하나만 연관되어 있을 때 Y는 X에 함수적 종속 또는 X가 Y를 함수적으로 결정한다고 함

039 정규화

- 테이블의 속성들이 상호 종속적인 관계를 갖는 특성을 이용하여 테이블을 무손실 분해하는 과정

제 1정규형 – 테이블에 속한 모든 속성의 도메인이 원자 값만으로 되어 있는 정규형

제 2정규형 – 테이블이 제 1정규형이고, 기본키가 아닌 모든 속성이 기본키에 대하여 완전 함수적 종속을 만족하는 정규형

제 3정규형 – 테이블이 제 2정규형이고, 기본키가 아닌 모든 속성이 기본키에 대해 이행적 함수적 종속을 만족하지 않는 정규형

BCNF – 테이블에서 모든 결정자가 후보키인 정규형

제 4정규형 – 테이블에 다중 값 종속이 존재할 경우 테이블의 모든 속성이 A에 함수적 종속 관계를 만족하는 정규형

제 5정규형 – 테이블의 모든 조인 종속이 테이블의 후보키를 통해서만 성립되는 정규형

040 반정규화

- 시스템의 성능을 향상하고 개발 및 운영의 편의성 등을 높이기 위해 정규화된 데이터 모델을 의도적으로 통합, 중복, 분리하여 정규화 원칙을 위배하는 행위

테이블 통합 – 두 개의 테이블이 조인되어 사용되는 경우가 많을 경우 성능 향상을 위해 하나의 테이블로 만들어 사용하는 것

테이블 분할 - 테이블을 수직 또는 수평으로 분할하는 것

중복 테이블 추가 – 작업의 효율성을 향상시키기 위해 테이블을 추가하는 것

중복 속성 추가 – 조인해서 데이터를 처리할 때 데이터를 조회하는 경로를 단축하기 위해 자주 사용하는 속성을 하나 더 추가하는 것

041 시스템 카탈로그

- 다양한 객체에 관한 정보를 포함하는 시스템 데이터베이스

메타 데이터 – 시스템 카탈로그에 저장된 정보

데이터 디렉터리 – 데이터 사전에 수록된 데이터에 접근하는 데 필요한 정보를 관리 유지하는 시스템

042 트랜잭션 분석 / CRUD 분석

트랜잭션 – 데이터베이스의 상태를 변환시키는 하나의 논리적 기능을 수행하기 위한 작업의 단위 또는 한꺼번에 모두 수행되어야 할 일련의 연산들을 의미

트랜잭션 특성 – 원자성 / 일관성 / 독립성, 격리성, 순차성 / 영속성, 지속성

CRUD 분석 – 프로세스와 테이블 간의 CRUD 매트릭스를 만들어서 트랜잭션을 분석하는 것

CRUD 매트릭스 – 프로세스와 데이터 간의 관계를 분석하는 분석표

트랜잭션 분석 – CRUD 매트릭스를 기반으로 테이블에 발생하는 트랜잭션 양을 분석하여 테이블에 저장되는 데이터의 양을 유추하고 DB의 용량 산정 및 구조의 최적화를 목적으로 함

043 인덱스

- 데이터 레코드를 빠르게 접근하기 위해 <키 값, 포인터> 쌍으로 구성되는 데이터 구조

인덱스의 종류 – 트리 기반 인덱스 / 비트맵 인덱스 / 함수 기반 인덱스 / 비트맵 조인 인덱스 / 도메인 인덱스

클러스터드 인덱스 / 넌클러스터드 인덱스

044 뷰 / 클러스터

뷰 – 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위해 하나 이상의 기본 테이블로부터 유도된 이름을 가지는 가상 테이블

- 저장장치 내에 물리적으로 존재하지 않지만 사용자에게는 있는 것처럼 간주됨

클러스터 – 데이터 저장시 데이터 액세스 효율을 향상시키기 위해 동일한 성격의 데이터를 동일한 데이터 블록에 저장하는 물리적 저장 방법

- 데이터의 분포도가 넓을수록 유리함

- 데이터 분포도가 넓은 테이블을 클러스터링하면 저장 공간을 절약할 수 있음

045 파티션

- 대용량의 테이블이나 인덱스를 작은 논리적 단위인 파티션으로 나누는 것

데이터 처리는 테이블 단위 / 데이터 저장 – 파티션 단위

범위 분할 – 지정한 열의 값을 기준으로 분할

해시 분할 – 해시 함수를 적용한 결과 값에 따라 데이터 분할

조합 분할 – 범위 분할로 분할한 다음 해시 함수를 적용하여 다시 분할하는 방식

046 분산 데이터베이스 설계

데이터베이스 용량 설계 – 데이터가 저장될 공간을 정의하는 것

테이블에 저장될 데이터양과 인덱스, 클러스터 등이 차지하는 공간 등을 예측하여 반영해야 함

분산 데이터베이스 – 논리적으로는 하나의 시스템에 속하지만 물리적으로는 네트워크를 통해 연결된 여러 개의 사이트에 분산된 데이터베이스를 말함

분산 데이터베이스의 목표 – 위치 투명성 / 중복 투명성 / 병행 투명성 / 장애 투명성

분산 설계 방법 – 테이블 위치 분산 / 분할 / 할당

047 데이터베이스 이중화 / 서버 클러스터링

데이터베이스 이중화 – 시스템 오류로 인한 데이터베이스 서비스 중단이나 물리적 손상 발생 시 복구하기 위해 동일한 데이터베이스를 복제하여 관리하는 것

- 하나 이상의 데이터베이스가 항상 같은 상태를 유지하므로 데이터베이스에 문제가 발생하면 복제된 데이터베이스를 이용하여 즉시 문제를 해결할 수 있음

데이터베이스 이중화의 분류 – Eager 기법 / Lazy 기법

데이터베이스 이중화 구성 방법 – 활동-대기 방법 / 활동-활동 방법

클러스터링 – 두 대 이상의 서버를 하나의 서버처럼 운영하는 기술

클러스터링 종류 – 고가용성 클러스터링 / 병렬 처리 클러스터링

RTO(목표 복구 시간) – 비상상태 또는 업무 중단 시점으로부터 복구되어 가동될 때까지의 소요 시간을 의미

RPO - 비상상태 또는 업무 중단 시점으로부터 데이터를 복구할 수 있는 기준점을 의미

048 데이터베이스 보안

- 데이터베이스의 일부 또는 전체에 대해서 권한이 없는 사용자가 액세스하는 것을 금지하기 위해 사용되는 기술

암호화 – 데이터를 보낼 때 송신자가 지정한 수신자 이외에는 그 내용을 알 수 없도록 평문을 암호문으로 변환하는 것

암호화 과정 / 복호화 과정

암호화 기법 – 개인키 암호 방식 / 공개키 암호 방식

접근통제 – 데이터가 저장된 객체와 이를 사용하려는 주체 사이의 정보흐름을 제한하는 것

접근통제 정책 / 접근통제 매커니즘 / 접근통제 보안모델

접근통제 기술 – 임의 접근통제 / 강제 접근통제 / 역할기반 접근통제

접근통제 정책 – 신분 기반 정책 / 규칙 기반 정책 / 역할 기반 정책

접근통제 매커니즘 – 정의된 접근통제 정책을 구현하는 기술적인 방법

접근통제 보안 모델 – 기밀성 모델 / 무결성 모델 / 접근통제 모델

접근통제 조건 – 접근통제 매커니즘의 취약점을 보완하기 위해 접근통제 정책에 부가하여 적용할 수 있는 조건

값 종속 통제 / 다중 사용자 통제 / 컨텍스트 기반 통제

감사 추적 – 사용자나 애플리케이션이 데이터베이스에 접근하여 수행한 모든 활동을 기록하는 기능

049 데이터베이스 백업

- 전산 장비의 장애에 대비하여 데이터베이스에 저장된 데이터를 보호하고 복구하기 위한 작업

로그 파일 – 데이터베이스이 처리 내용이나 이용 상황 등 상태 변화를 시간의 흐름에 따라 모두 기록한 파일

데이터베이스 복구 알고리즘 – NO-UNDO / UNDO/NO-REDO / UNDO/REDI / NO-UNDO/NO-REDO

백업종류 – 복구 수준에 따라 운영체제를 이용하는 물리 백업과 DBMS 유틸리티를 이용하는 논리 백업이 있음

050 스토리지

- 단일 디스크로 처리할 수 없는 대용량의 데이터를 저장하기 위해 서버와 저장장치를 연결하는 기술

DAS – 서버와 저장장치를 전용 케이블로 직접 연결하는 방식

NAS – 서버와 저장장치를 네트워크를 통해 연결하는 방식

SAN – 서버와 저장장치를 연결하는 전용 네트워크를 별도로 구성하는 방식

051 논리 데이터 모델의 변환

엔티티를 테이블로 변환

- 논리 데이터 모델에서 정의된 엔티티를 물리 데이터 모델의 테이블로 변환하는 것

슈퍼타입/서브타입을 테이블로 변환

- 논리 데이터 모델에서 이용되는 형태이므로 물리 데이터 모델을 설계할 때는 슈퍼타입/서브타입을 테이블로 변환해야 함

슈퍼타입 기준 테이블 변환

- 서브타입을 슈퍼타입에 통합하여 하나의 테이블로 만드는 것

서브타입 기준 테이블 변환

- 슈퍼타입 속성들을 각각의 서브타입에 추가하여 서브타입들을 개별적인 테이블로 만드는 것

개별타입 기준 테이블 변환

- 슈퍼타입과 서브타입들을 각각의 개별적인 테이블로 변환하는 것

속성을 컬럼으로 변환

- 논리 데이터 모델에서 정의한 속성을 물리 데이터 모델의 컬럼으로 변환

일반 속성 변환 / Primary UID를 기본키로 변환 / Primary UID를 기본키로 변환 / Secondary를 유니크키로 변환

관계를 외래키로 변환

- 논리 데이터 모델에서 정의된 관계는 기본키와 이를 참조하는 외래키로 변환함

052 자료구조

- 자료를 기억장치의 공간 내에 저장하는 방법과 자료 간의 관계, 처리 방법 등을 연구 분석하는 것을 말함

선형 구조 – 배열 / 선형 리스트 – 연속 리스트, 연결 리스트 / 스택 / 큐 / 데크

비선형 구조 – 트리 / 그래프

배열 – 크기와 형이 동일한 자료들이 순서대로 나열된 자료의 집합

연속 리스트 – 배열과 같이 연속되는 기억장소에 저장되는 자료구조

연결 리스트 – 자료들을 임의의 기억공간에 기억시키되 자료 항목의 순서에 따라 노드의 포인터 부분을 이용하여 서로 연결시킨 자료 구조

스택 – 리스트 한쪽 끝으로만 자료의 삽입, 삭제 작업이 이루어지는 자료 구조

큐 – 리스트의 한쪽에서는 삽입 작업이 이루어지고 다른 한쪽에서는 삭제 작업이 이루어지는 자료 구조

그래프 – 정점과 간선의 두 집합으로 이루어지는 자료 구조

053 트리

- 정점과 선분을 이용하여 사이클을 이루지 않도록 구성한 그래프의 특수한 형태]

노드 : 트리의 기본 요소로서 자료 항목과 다른 항목에 대한 가지를 합친 것

근 노드 : 트리의 맨 위에 있는 노드

디그리 : 각 노드에서 뻗어나온 가지의 수

단말 노드 = 입 노드 : 자식이 하나도 없는 노드

비단말 노드 : 자식이 하나라도 있는 노드

조상 노드 : 임의의 노드에서 근 노드에 이르는 경로상에 있는 노드들

자식 노드 : 어떤 노드에 연결된 다음 레벨의 노드들

부모 노드 : 어떤 노드에 연결된 이전 레벨의 노드들

형제 노드 : 동일한 부모를 갖는 노드들

깊이 : 트리에서 노드가 가질 수 있는 최대의 레벨

숲 : 여러 개의 트리가 모여 있는 것

트리의 디그리 : 노드들의 디그리 중에서 가장 많은 수

054 이진 트리

- 차수가 2 이하인 노드들로 구성된 트리

운행법 – 트리를 구성하는 각 노드들을 찾아가는 방법

Preorder 운행법 – 이진 트리를 Root -> Left -> Right 순으로 운행하며 노드들을 찾아가는 방법

Inorder 운행법 – 이진 트리를 Left -> Root -> Right 순으로 운행하며 노드들을 찾아가는 방법

Postorder 운행법 – 이진트리를 Left -> Right -> Root 순으로 운행하며 노드들을 찾아가는 방법

수식의 표기법 – 이진 트리를 만들어진 수식을 인오더, 프리오더. 포스터오더로 운행하면 각각 중위, 전위, 후위 표기법이 됨

전위 표기법 : 연산자 -> Left -> Right

중위 표기법 : Left -> 연산자 -> Right

후위 표기법 : Left -> Right -> 연산자

055 정렬

삽입 정렬 – 이미 순서화된 파일에 새로운 하나의 레코드를 순서에 맞게 삽입시켜 정렬하는 방식

선택 정렬 – 최소값을 찾아 첫 번째 레코드 위치에 놓고 다시 최소값을 찾아 두 번째 레코드 위치에 놓는 방식을 반복하여 정렬하는 방식

버블 정렬 – 인접한 두 개의 레코드 키 값을 비교하여 그 크기에 따라 레코드 위치를 서로 교환하는 정렬 방식

쉘 정렬 – 매개변수 값으로 서브파일을 구성하고 각 서브파일으르 삽입 정렬 방식으로 순서 배열하는 과정

퀵 정렬 – 키를 기준으로 작은 값은 왼쪽, 큰 값은 오른쪽 서브 파일에 분해시키는 과정

힙 정렬 – 전이진 트리를 이용한 정렬 방식

2-Way 합병 정렬 – 이미 정렬되어 있는 두 개의 파일을 한 개의 파일로 합병하는 정렬 방식

기수 정렬 = Bucket Sort – 큐를 이용하여 자릿수별로 정렬하는 방식