

데이터베이스론

문 1. 파일 시스템과 데이터베이스 시스템의 설명으로 옳지 않은 것은?

- ① 자료의 중복 정도는 파일 시스템이 데이터베이스 시스템보다 높다.
- ② 파일 시스템은 특정 응용 프로그램에 종속적인 반면, 데이터베이스 시스템은 특정 응용 프로그램에 종속적이지 않다.
- ③ 데이터베이스 시스템의 정보가 중앙 집중화되고 원격지의 사용자에게 접근이 허용되면 정보가 누출될 가능성이 높을 수 있다.
- ④ 회복 처리기, 트랜잭션 처리기 등의 오버헤드로 인하여 데이터베이스 시스템은 다수의 사용자를 지원하기에 적합하지 않다.

문 2. DBMS 언어에 대한 설명으로 옳지 않은 것은?

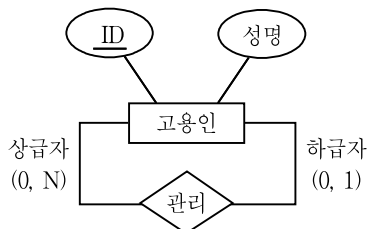
- ① DBMS에 따라 데이터 정의어(DDL: Data Definition Language)는 개념 스키마와 내부 스키마를 명시하며 두 스키마 사이의 사상을 정의하는데 사용되기도 한다.
- ② 저장구조 정의어(SDL: Storage Definition Language)는 내부 스키마를 정의하는데 사용된다.
- ③ 데이터 조작어(DML: Data Manipulation Language)는 데이터에 대한 검색, 삽입, 삭제, 수정을 위한 언어이다.
- ④ 비절차적 데이터 조작어는 절차적 데이터 조작어에 비해 상대적으로 고수준의 언어이며, 한번에 한 레코드만을 조작한다.

문 3. 관계 데이터베이스의 릴레이션 $R(A, B, C)$ 에서 “에트리뷰트 B가 A에 함수적으로 종속된다.”는 것은 에트리뷰트 A의 특정 값에 대하여 에트리뷰트 B의 유일한 값으로 대응된다는 의미이다. 이러한 함수적 종속성을 유추하는 규칙 중 옳지 않은 것은?

- ① if $A \rightarrow B$, then $AC \rightarrow BC$
- ② if $A \rightarrow B$ and $B \rightarrow C$, then $A \rightarrow C$
- ③ if $A \rightarrow B$ and $A \rightarrow C$, then $A \rightarrow BC$
- ④ if $AB \rightarrow C$, then $A \rightarrow B$ and $A \rightarrow C$

문 4. 개체-관계(ER) 다이어그램에 대한 설명으로 옳은 것은?

(단, 개체와 관계 사이에 표시된 정수 쌍은 개체와 관계 사이의 참여 관계를 각각 (최솟값, 최댓값)으로 표현한 것이다. 밑줄은 기본키이다)



- ① 관계 ‘관리’는 M:N(다 대 다) 관계를 나타낸다.
- ② 고용인으로서 상급자는 하급자를 반드시 1명 이상 가진다.
- ③ 고용인으로서 하급자는 상급자를 갖지 않을 수 있다.
- ④ 1명의 상급자는 0명 또는 여러 명의 하급자를 가질 수 있는 관계이므로, 이를 고용인이 아닌 별도의 릴레이션으로 표현해야 한다.

문 5. 다음 ER 다이어그램은 약한 개체 집합(Weak Entity Set)의 예를 나타낸다. 이 ER 다이어그램에 대한 설명으로 옳지 않은 것은? (단, Person은 개체 집합이며, Dependents는 약한 개체 집합이다. 밑줄은 기본키이며, 점선 밑줄은 부분키이다)



- ① Person과 Dependents는 일 대 다 관계이다.
- ② 모든 Dependents 개체들은 서로 다른 name 값을 갖는다.
- ③ Dependents에 속하는 모든 개체에 대해 반드시 대응되는 Person 개체가 존재한다.
- ④ Person은 Dependents에 대한 식별 개체 집합(Identifying Entity Set)이다.

문 6. 보조저장장치에 대한 설명으로 옳지 않은 것은?

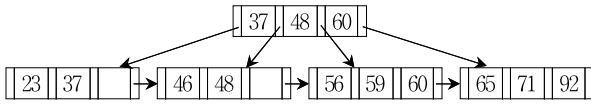
- ① 플래시메모리는 데이터를 읽는 속도와 쓰는 속도가 비슷하며, 하드디스크보다 빠른 접근속도를 제공한다.
- ② 하드디스크의 접근시간은 회전지연시간(Rotational Latency), 탐색시간(Seek Time), 그리고 전송시간(Transfer Time)으로 구별되며, 이 중에서 전송시간은 대략적으로 데이터의 크기에 비례한다.
- ③ 자기 테이프는 순차적 접근(Sequential Access)만이 가능하다.
- ④ RAID는 비교적 저렴한 디스크를 여러 개 중복하여 데이터 저장의 신뢰성을 높이도록 만들어진 저장장치이다.

문 7. A, B, C, D, E, F는 릴레이션 R의 에트리뷰트이며 그들 각각은 집합을 허용하지 않는다. 다음 각 FD가 릴레이션 R에 대한 함수적 종속성들의 집합이라 가정할 경우, 이들 중 BCNF(Boyce-Codd Normal Form)를 만족하는 것은?

- ① $FD = \{A \rightarrow BCD, B \rightarrow AEF\}$
- ② $FD = \{A \rightarrow BCD, B \rightarrow EF\}$
- ③ $FD = \{AB \rightarrow C, A \rightarrow DEF\}$
- ④ $FD = \{AB \rightarrow CDEF, C \rightarrow A\}$

문 8. 다음은 차수가 4인 B⁺-트리의 초기 상태이다. 이 B⁺-트리에 색인키 '18', '21'을 순서대로 삽입할 때 생성되는 B⁺-트리의 구조로 옳은 것은?

<B⁺-트리의 초기 상태>



- ①
-
- ②
-
- ③
-
- ④
-

문 9. 두 개 릴레이션 Student(sno, sname, address, score, dept)와 Department(dno, dname, manager)가 있을 때, 질의문 “SELECT * FROM Student s, Department d WHERE s.dept > 100;”을 수행하려고 한다. 이 질의 수행으로 생성되는 결과 릴레이션의 차수(Degree)와 카디널리티(Cardinality)는 각각 얼마인가?
(단, 릴레이션 Student의 애트리뷰트 ‘소속(dept)’은 릴레이션 Department의 애트리뷰트 ‘코드(dno)’를 외부키로 참조한다)

Student

학번 (sno)	성명 (sname)	주소 (address)	성적 (score)	소속 (dept)
9801	홍길동	서울	80	100
9802	김철수	대전	90	200
9803	이순자	강릉	90	100
9805	이원영	부산	95	200
9806	홍남순	서울	65	300

Department

코드 (dno)	학과명 (dname)	학과장 (manager)
100	정통	이순신
200	전자	강감찬
300	기계	김유신

- ① 차수 = 5, 카디널리티 = 3
 ② 차수 = 5, 카디널리티 = 2
 ③ 차수 = 8, 카디널리티 = 9
 ④ 차수 = 8, 카디널리티 = 3

문 10. 두 개의 릴레이션 Student, Enroll에 대한 합병 조인(Merge Join) 연산을 수행하려고 한다. 각 릴레이션의 레코드 개수와 디스크 상의 블록의 개수에 대한 정보가 아래와 같이 주어져 있다. 이때 두 릴레이션이 조인 애트리뷰트에 대하여 정렬이 되어 있을 경우, 합병 조인을 수행하였을 때 디스크 접근 회수는? (단, 한 번의 디스크 접근으로는 1개의 디스크 블록만을 읽어 올 수 있으며, 메모리 상에는 각 릴레이션마다 1개의 블록만을 저장할 수 있다고 가정한다)

- Student 릴레이션의 레코드 개수: 5,000
 ○ Student 릴레이션의 블록 개수: 100
 ○ Enroll 릴레이션의 레코드 개수: 8,000
 ○ Enroll 릴레이션의 블록 개수: 300

- ① 400
 ② 13,000
 ③ 30,000
 ④ 40,000,000

문 11. 사용자 A가 어떤 릴레이션에 대한 SELECT 권한을 ‘WITH GRANT OPTION’과 함께 사용자 B에게 허가하고, 사용자 B가 그 릴레이션에 대한 SELECT 권한을 ‘WITH GRANT OPTION’과 함께 사용자 C에게 허가하고, 사용자 C가 그 릴레이션에 대한 SELECT 권한을 ‘WITH GRANT OPTION’과 함께 사용자 D에게 허가했다. 이후 사용자 A가 사용자 B의 SELECT 권한을 취소하면 사용자 C와 D의 권한으로 옳은 것은?

- ① C와 D도 연쇄적으로 SELECT 권한이 취소된다.
 ② C의 SELECT 권한이 취소되고, D의 SELECT 권한이 취소되지 않는다.
 ③ C와 D 모두 SELECT 권한이 취소되지 않는다.
 ④ C의 SELECT 권한이 취소되지 않고, D의 SELECT 권한은 취소된다.

문 12. XML 문서에 대한 설명으로 옳지 않은 것은?

- ① XML은 인터넷 환경에서 구조화된 데이터의 교환, 저장에 위하여 개발된 문서 표준이다.
 ② XML 문서에 대한 질의 처리를 위해서 XPath, XQuery 등의 질의어를 사용한다.
 ③ XML 문서의 구조 형식을 정의하기 위해서 Document Type Definition(DTD)을 사용한다.
 ④ XML 문서는 개방적인 구조로 인하여 관계형 데이터베이스에 저장할 수 없으며, 일반 파일의 형태로 저장하거나 XML 전용 데이터베이스를 사용한다.

문 13. 질의 최적화 과정에서 질의문 계획(Query Plan)의 평가 및 결정 과정에 대한 설명으로 옳지 않은 것은?

- ① 질의 최적화의 마지막 단계로서 후보 프로시저 선정과정 후에 수행한다.
- ② 프로시저들의 조합으로 만들어지는 계획 중 가장 비용이 적게 드는 것을 고른다.
- ③ 가능한 후보 질의문 계획을 모두 생성하고 이를 평가하기 위해 경험적(Heuristic) 기법을 사용한다.
- ④ 각 프로시저의 실행 비용을 계산하기 위해 매개변수 비용식을 사용한다.

문 14. DBMS에서 트랜잭션의 동시성 제어(Concurrency Control)와 회복(Recovery) 기술에 대한 설명으로 옳지 않은 것은?

- ① 데이터베이스에 대한 즉시 갱신은 데이터베이스에 대한 갱신 로그를 저장함으로서 회복에 대비한다. 일반적으로 데이터베이스와 로그의 동시 손상을 피하기 위해 별도의 전용 디스크에 로그를 저장할 수 있다.
- ② 데이터베이스 갱신에 대한 로그 기록을 유지하는 것은 일반적으로 성능 저하를 가져오므로, 성능을 유지하기 위해 로그보다 데이터베이스 갱신을 먼저 쓰는 것을 WAL(Write-Ahead Logging)이라고 한다.
- ③ 검사점(Checkpoint) 회복 기법은 트랜잭션 수행 중 주기억 장치의 로그 버퍼와 데이터베이스 버퍼를 디스크에 저장한다.
- ④ 2-단계 로킹(2-Phase Locking)은 확장단계(Growing Phase)와 수축 단계(Shrinking Phase)로 구성된다. 확장단계에서는 트랜잭션이 데이터 항목에 대하여 새로운 로크를 요청할 수 있지만 해제할 수는 없으며, 수축단계에서는 반대로 로크를 해제할 수는 있지만 새로 요청할 수 없다.

문 15. 다음 트랜잭션 스케줄에서 직렬가능한 스케줄은? (단, $r_1(X)$ 와 $w_1(X)$ 는 각각 트랜잭션 T1의 데이터 X에 대한 읽기 및 쓰기 연산이다)

- ① $r_1(X); r_3(X); w_1(X); r_2(X); w_3(X);$
- ② $r_1(X); r_3(X); w_3(X); w_1(X); r_2(X);$
- ③ $r_3(X); r_2(X); w_3(X); r_1(X); w_1(X);$
- ④ $r_3(X); r_2(X); r_1(X); w_3(X); w_1(X);$

문 16. 검사점(Checkpoint) 회복 기법을 수행하기 위한 작업순서로 옳은 것은?

ㄱ. 변경된 데이터 블록을 모두 안정된 저장소로 출력한다.
 ㄴ. 로그 버퍼에 있는 모든 레코드를 안정된 저장소로 출력한다.
 ㄷ. 검사점 로그 레코드를 안정된 저장소로 출력한다.

- ① ㄱ - ㄴ - ㄷ
- ② ㄴ - ㄱ - ㄷ
- ③ ㄷ - ㄱ - ㄴ
- ④ ㄷ - ㄴ - ㄱ

문 17. 분산 데이터베이스 시스템에 대한 설명으로 옳지 않은 것은?

- ① 여러 사이트에 분산된 데이터를 동기화하고 전체 데이터베이스의 무결성을 유지하기 위해 분산 트랜잭션 기능을 사용한다.
- ② 중앙집중형 데이터베이스 시스템보다 구현이 간단하기 때문에 개발비용을 더 많이 줄일 수 있다.
- ③ 중앙집중형 데이터베이스 시스템과는 달리 각 사이트의 붕괴와 통신 회선의 고장과 같은 새로운 유형의 고장으로부터 회복하는 기능을 부가적으로 제공해야 한다.
- ④ 임의의 사이트에 존재하는 데이터베이스 카탈로그(디렉토리) 정보는 전체 분산 데이터베이스에 걸친 전역적인 정보를 유지할 수도 있고 지역적인 정보만을 유지할 수도 있다.

문 18. 데이터베이스 튜닝에 대한 설명으로 옳지 않은 것은?

- ① 데이터베이스의 성능을 높이기 위해 의도적으로 스키마를 중복이 발생하도록 변경할 수 있다.
- ② 성능이 저하될 때 이미 구축된 인덱스 대신 다른 종류의 인덱스를 선택할 수 있다. 하지만, 기존 인덱스를 단순히 재구성하는 것으로는 성능 향상에 영향을 줄 수 없다.
- ③ 성능 향상을 위해 릴레이션을 수직 또는 수평으로 분할할 수 있다.
- ④ 자주 수행되는 질의 형태의 경우 뷰를 생성하도록 한다.

문 19. SQL을 이용해서 트랜잭션의 고립수준(Isolation Level)을 지정하려 한다. 다음 조건을 만족시키기 위한 고립수준 중 가장 적절한 것은?

트랜잭션 T는 완료된 트랜잭션의 변경만을 읽고, T가 기록한 데이터는 다른 트랜잭션에 의해서 변경되지 않는 것을 보장해야 한다. 하지만, T가 읽고 있는 값은 T가 완료되기 전에 다른 트랜잭션에 의해서 변경되어도 문제가 없다.

- ① SERIALIZABLE
- ② REPEATABLE READ
- ③ READ COMMITTED
- ④ READ UNCOMMITTED

문 20. 데이터베이스를 분산시키기 위한 방법으로 수직 분할(Vertical Fragmentation), 수평 분할(Horizontal Fragmentation), 그리고 중복이 있다. 이 방법들에 대한 설명으로 옳지 않은 것은?

- ① 2단계 완료 규약(2-Phase Commit Protocol)은 수직 분할, 수평 분할 또는 중복으로 분산된 데이터베이스의 일관성을 유지하기 위한 트랜잭션의 수행 방법이다.
- ② 데이터베이스를 중복으로 분산시키면 검색과 갱신 성능이 향상된다.
- ③ 테이블 R을 $R_1, R_2, \dots, R_n$ 으로 수평 분할하였다면, 분산 데이터베이스 시스템에 주어진 선택 질의 σ 은 $\sigma(R_1) \cup \sigma(R_2) \cup \dots \cup \sigma(R_n)$ 로 수행된다. (단, $\cup$ 는 합집합 연산자를 의미한다)
- ④ 테이블 R을 $R_1, R_2, \dots, R_n$ 으로 수직 분할하였다면, $R = R_1 \bowtie R_2 \bowtie \dots \bowtie R_n$ 으로 표현될 수 있다. (단, $\bowtie$ 는 조인 연산자를 의미한다)