

데이터베이스론

문 1. 관계 데이터 모델의 키와 제약 조건에 대한 설명으로 옳은 것만을 모두 고르면?

- ㄱ. $R(A, B, C)$ 릴레이션에서 기본키가 복합키 (A, B) 이고 B 가 외래키라면, 참조 무결성에 의해 B 는 널값을 가질 수 있다.
 ㄴ. 주어진 릴레이션 R 의 속성들의 부분집합 X 에 대해, 어떤 튜플도 동일한 값을 가지지 않는다면, 이러한 속성의 집합 X 를 그 릴레이션의 슈퍼키라고 한다.
 ㄷ. 참조 무결성 제약은 참조할 수 없는 외래키 값을 가져서는 안 된다는 것을 의미한다.
 ㄹ. 대체키와 외래키는 유일성과 최소성을 모두 만족해야 한다.

- ① ㄱ, ㄴ
 ② ㄱ, ㄷ
 ③ ㄴ, ㄷ
 ④ ㄷ, ㄹ

문 2. 릴레이션 스키마의 표현은 릴레이션명(속성명1:도메인1, 속성명2:도메인2, ..., 속성명n:도메인n)이다. 이 표현을 따르는 릴레이션 스키마 $A(a:int, x:int, c:int)$ 와 $B(b:int, x:int, d:int)$ 에 대한 관계 대수식 $\pi_{a,b}(\sigma_{a>10}(A \bowtie_{A.x=B.x} B))$ 와 동등한 관계대수식은?

- ① $\pi_{a,b}(\sigma_{a>10}(A) \times B)$
 ② $\sigma_{a>10}(\pi_{a,b}(A) \bowtie_x B)$
 ③ $\pi_{a,b}(A \bowtie_{a\sigma_{a>10}}(B))$
 ④ $\pi_{a,b}(\sigma_{a>10}(A) \bowtie_{A.x=B.x} B)$

문 3. 스키마가 사원(사원번호, 직급, 보너스)인 사원 테이블의 인스턴스가 <보기 1>과 같을 때, <보기 2>의 ㉠, ㉡에 들어갈 말로 옳게 짝 지은 것은?

— <보기 1> —

사원번호	직급	보너스
1000	과장	500
1001	과장	500
1002	사원	200

— <보기 2> —

이 사원 테이블에서 보너스는 직급별로 결정된다. 이때 직급과 보너스 속성은 모두 기본키인 사원번호에 의해 결정되지만 사원번호가 직급을 결정하고 직급이 보너스를 결정하는 (㉠) 관계를 가지고 있으므로, 이 테이블은 (㉡)이라고 할 수 있다.

- ㉠ ㉡
- ① 완전 함수 종속 제 2 정규형
 ② 완전 함수 종속 제 3 정규형
 ③ 이행적 함수 종속 제 2 정규형
 ④ 이행적 함수 종속 제 3 정규형

문 4. 데이터베이스 시스템에서 데이터 저장 요구량이 빠르게 증가하고 있어서 많은 수의 디스크가 요구된다. 다수의 디스크 드라이브를 사용하여 저장 용량을 늘리고 읽기와 쓰기를 병렬로 수행하기도 하며, 디스크의 고장에 대비하기 위해 RAID(Redundant Arrays of Independent Disks)를 구성하여 활용한다. 이와 같은 RAID에서 1 TByte 디스크 드라이브 6개를 이용하여 RAID를 구성할 때, 구성된 RAID의 저장 용량이 가장 작은 구성 방법은?

- ① 레벨 0 ② 레벨 1
 ③ 레벨 5 ④ 레벨 6

문 5. 다음은 산업통계 테이블이다. 이 테이블을 대상으로 아래 결과를 출력하고자 한다. 이를 위한 SQL 질의문은?

산업통계			
대분류	소분류	종사자수	평균연봉
제조업	반도체	10000	4000
제조업	스마트폰	5000	5000
제조업	가전	7000	6000
IT	SW개발	6000	4000
IT	게임개발	3000	3000
서비스업	미용	4000	3500
서비스업	유통	9000	4500

SQL 실행결과

대분류	소분류	종사자수	평균연봉
제조업	반도체	10000	4000
IT	SW개발	6000	4000
서비스업	유통	9000	4500

- ① SELECT *
 FROM 산업통계
 WHERE 종사자수 >= 6000;
 ② SELECT *
 FROM 산업통계
 WHERE 종사자수 = (SELECT MAX(종사자수)
 FROM 산업통계);
 ③ SELECT A.*
 FROM 산업통계 A,
 (SELECT 대분류, MAX(종사자수) AS 종사자수
 FROM 산업통계
 GROUP BY 대분류) B
 WHERE A.종사자수 = B.종사자수 AND A.대분류 = B.대분류;
 ④ SELECT 대분류, 소분류, MAX(종사자수), 평균연봉
 FROM 산업통계
 GROUP BY 대분류;

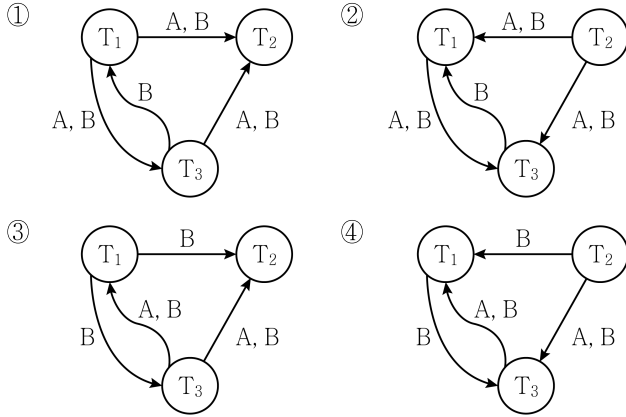
문 6. 버킷 용량 $C=2$ 인 버킷들의 주소 공간이 $0 \sim (N-1)$ 인 해시에서 키를 K 로 하는 해시 함수 $h(K) = K \bmod N$ 이고, N 값은 5이다. 이때 일련의 K 값들이 3, 5, 7, 9, 11, 13, 18, 14 순서로 삽입된 후 4번 버킷의 값은? (단, 버킷에서의 충돌은 충돌 발생 버킷 주소 i 에 대하여 $(i+1) \bmod N$ 을 통해 지정하는 선형탐색 개방 주소법을 이용하여 해결한다)

- ① 5, 14 ② 3, 13
 ③ 9, 14 ④ 9, 18

- 문 7. 다음 트랜잭션 스케줄의 선행그래프(precedence graph)로 옳은 것은? (단, t_i 는 시간 단위, T_j 는 트랜잭션, $R(A)$ 는 A항목 읽기, $W(A)$ 는 A항목 쓰기를 나타낸다)

시간	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
----	-------	-------	-------	-------	-------	-------	-------	-------	-------	----------

T_1	R(A)	W(A)				R(B)		W(B)		
T_2					R(A)					R(B)
T_3			R(A)	W(A)			R(B)		W(B)	



- 문 8. 다음 SQL 구문에 대한 설명으로 옳지 않은 것은?

```
CREATE ASSERTION NO_DESIGNERS_IN_SEOUL
AS CHECK
( NOT EXISTS
  ( SELECT *
    FROM EMPLOYEE E, DEPARTMENT D
    WHERE E.DEPTNO = D.DEPTNO
      AND E.JOB = 'DESIGNER'
      AND D.LOC = 'SEOUL' )
);
```

- ① 명시된 이벤트가 발생할 때마다 DBMS가 자동적으로 수행하는 구문이다.
- ② 대상 테이블과 관련된 SQL의 갱신문 수행 시 위의 구문이 검사된다.
- ③ 위의 SQL이 불필요할 경우 DROP문으로 제거할 수 있다.
- ④ 위의 SQL에서 명세하고 있는 조건에 따라 DB연산이 수행되지 않을 수 있다.

- 문 9. XML Schema와 DTD에 대한 비교 중 옳지 않은 것으로만 묶은 것은?

	구분	XML Schema	DTD
ㄱ	구조	복잡함	상대적으로 간결함
ㄴ	문법	전용문법	XML문법
ㄷ	Namespace	지원함	지원하지 못함
ㄹ	확장성	문자열 치환을 통한 확장성	객체지향적인 확장성

- ① ㄱ, ㄴ ② ㄱ, ㄷ
③ ㄴ, ㄹ ④ ㄷ, ㄹ

- 문 10. 즉시 갱신(Immediate Update) 전략을 이용하는 회복 시스템에서 <보기 1>의 로그 레코드 형식으로 <보기 2>의 로그 레코드가 형성되어 있다. 복구는 REDO 단계 실행 후 UNDO 단계를 실행한다. 이 시스템에서 복구 절차에 대한 설명으로 옳지 않은 것으로만 묶은 것은?

— <보기 1> —

- T_i : 트랜잭션 번호가 i 인 트랜잭션
- $\langle T_i, \text{start} \rangle$, $\langle T_i, \text{commit} \rangle$, $\langle T_i, \text{abort} \rangle$: T_i 에 대한 시작, 완료, 철회를 기록하는 형식
- $\langle T_i, \text{rollback} \rangle$: T_i 가 실행 중 rollback이 발생함을 기록하는 형식
- $\langle T_i, \text{데이터항목, 이전값, 이후값} \rangle$: T_i 가 데이터 항목을 변경할 때 기록하는 형식
- $\langle T_i, \text{데이터항목, 이전값} \rangle$: T_i 가 철회되기 위하여 UNDO를 실행할 때 기록하는 형식
- $\langle \text{checkpoint, \{트랜잭션리스트\}} \rangle$: 검사점 실행을 기록하는 형식
- $\{\text{트랜잭션리스트}\}$: 검사점 실행시점에 실행 중에 있던 트랜잭션들의 목록을 포함하는 트랜잭션 리스트

— <보기 2> —

로그번호	로그 레코드
1	$\langle T_0, \text{start} \rangle$
2	$\langle T_0, A, 100, 200 \rangle$
3	$\langle T_1, \text{start} \rangle$
4	$\langle T_1, B, 200, 300 \rangle$
5	$\langle T_0, C, 300, 400 \rangle$
6	$\langle \text{checkpoint, } \{T_0, T_1\} \rangle$
7	$\langle T_1, D, 500, 600 \rangle$
8	$\langle T_3, \text{start} \rangle$
9	$\langle T_0, \text{rollback} \rangle$
10	$\langle T_4, \text{start} \rangle$
11	$\langle T_0, C, 300 \rangle$
12	$\langle T_3, E, 600, 700 \rangle$
13	$\langle T_0, A, 100 \rangle$
14	$\langle T_4, F, 700, 800 \rangle$
15	$\langle T_0, \text{abort} \rangle$
16	$\langle T_4, G, 800, 900 \rangle$
17	$\langle T_3, H, 900, 950 \rangle$
18	$\langle T_4, \text{commit} \rangle$
--고장 발생--	

- ㄱ. 복구가 시작되면 checkpoint 로그 레코드를 찾아 UNDO 대상 트랜잭션 리스트를 $\{T_0, T_1\}$ 으로 초기화한다.
- ㄴ. T_0 의 rollback을 처리하기 위하여 생성된 $\langle T_0, C, 300 \rangle$, $\langle T_0, A, 100 \rangle$ 은 복구과정에서 UNDO 단계 연산이 된다.
- ㄷ. REDO 단계를 완료하면 UNDO 대상 트랜잭션 리스트는 $\{T_1, T_3\}$ 로 구성된다.
- ㄹ. REDO 단계에서 로그 레코드 $\langle T_1, D, 500, 600 \rangle$, $\langle T_0, C, 300 \rangle$, $\langle T_3, E, 600, 700 \rangle$, $\langle T_0, A, 100 \rangle$, $\langle T_4, F, 700, 800 \rangle$, $\langle T_4, G, 800, 900 \rangle$, $\langle T_3, H, 900, 950 \rangle$ 은 REDO 연산의 대상이 된다.
- ㅁ. REDO 단계에서는 REDO 연산에 따른 로그 레코드를 로그에 기록한다.
- ㅂ. UNDO 단계에서는 $\langle T_3, H, 900 \rangle$, $\langle T_3, E, 600 \rangle$, $\langle T_3, \text{abort} \rangle$, $\langle T_1, D, 500 \rangle$, $\langle T_1, B, 200 \rangle$, $\langle T_1, \text{abort} \rangle$ 를 로그에 기록한다.

- ① ㄱ, ㅂ ② ㄴ, ㄹ
③ ㄷ, ㄹ ④ ㄹ, ㅂ

문 11. 데이터베이스에 대한 설명으로 옳지 않은 것은?

- ① 데이터 종속성이란 응용프로그램과 데이터 간의 상호의존 관계를 의미하며, 이는 응용프로그램이 접근하려는 데이터의 구성 방법에 맞게 작성되어야 함을 의미한다.
- ② 데이터 중복성이란 한 시스템 내에 내용이 같은 데이터가 중복 저장 관리되는 것을 의미하며, 경제성 문제는 발생하나 보안성은 향상된다.
- ③ 논리적 데이터 독립성이란 기존 응용프로그램에 영향을 주지 않고 데이터베이스의 논리적 구조를 변경시킬 수 있는 능력을 의미한다.
- ④ 데이터 일관성이란 여러 개의 데이터가 모두 하나의 사실을 나타낸다면 논리적으로 그 내용이 모두 동일해야 함을 의미한다.

문 12. 트랜잭션의 특성에 대한 설명으로 옳지 않은 것은?

- ① S은행 계좌에서 K은행 계좌로 이체하는 도중에 오류가 발생하면 수행된 트랜잭션을 취소하여 원래의 상태로 만들어야 한다.
- ② S은행 계좌와 K은행 계좌 금액의 합은 S계좌에서 K계좌로 이체한 후에도 그 금액의 합이 같아야 한다.
- ③ S은행 계좌에서 K은행 계좌로의 이체가 성공적으로 완료된 후에는 항상 이체가 완료된 상태를 유지해야 한다.
- ④ S은행 계좌에서 K은행 계좌로 이체하는 트랜잭션을 수행하는 동안 다른 트랜잭션이 관여할 수 없으나 참조할 수는 있다.

문 13. 다음의 사원 테이블에 대한 SQL 질의문의 결과는? (단, 열 A, B는 모두 INTEGER 타입이고, 값이 표시되어 있지 않은 열은 NULL이다)

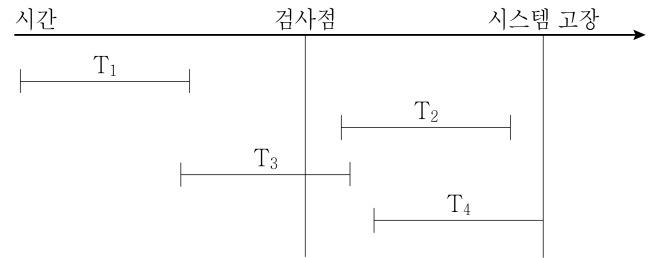
사원	
A	B
1000	500
1000	1000
2000	1000
	4000

사원 테이블에 대한 SQL 질의문

```
SELECT COUNT( A ), COUNT( DISTINCT A ),
       MAX( A+B )
FROM 사원;
```

- ① 3 2 3000
- ② 3 2 4000
- ③ 4 3 3000
- ④ 4 3 4000

문 14. 다음은 트랜잭션의 수행 중에 장애가 발생할 경우의 회복(Recovery) 기법을 적용하여 트랜잭션의 상태를 복구시키는 사례이다. 로그는 그대로 기록을 유지하면서, 회복 관리자가 정하는 일정한 시간 간격으로 검사점을 생성하는 Checkpoint 회복기법을 적용하는 경우에 트랜잭션들의 상태를 설명한 것으로 옳지 않은 것은?



- ① T₁은 회복작업을 할 필요가 없다.
- ② T₂는 트랜잭션 전체에 대하여 UNDO 연산을 수행해야 한다.
- ③ T₃는 검사점 이후에 발생한 변경 부분에 대해서만 REDO 연산을 수행해야 한다.
- ④ T₄는 트랜잭션 전체에 대하여 UNDO 연산을 수행해야 한다.

문 15. 다음은 테이블을 생성하는 SQL 구문이다. 이 테이블에 대한 연산을 설명하는 것으로 옳지 않은 것은?

```
CREATE TABLE 직원 (
    직원번호 NUMBER NOT NULL,
    직원이름 CHAR(10) UNIQUE,
    부서번호 NUMBER
        CHECK(부서번호 IN (1,2,3,4)) DEFAULT 1,
    PRIMARY KEY(직원번호),
    FOREIGN KEY(부서번호) REFERENCES
        부서(부서번호) ON DELETE CASCADE );
```

- ① 동명이인의 직원 이름을 추가했더니 입력되지 않았다.
- ② 부서가 없는 신입사원을 입력했더니 부서번호가 널값이 되었다.
- ③ 부서 테이블에서 특정 부서를 삭제했더니 해당 부서에 속한 직원들도 삭제되었다.
- ④ 직원 번호에 동일한 번호를 입력했더니 처리되지 않았다.

문 16. 두 트랜잭션 스케줄 s_1 , s_2 에 대하여, s_1 의 비연쇄(Cascadeless) 스케줄 여부와 s_2 의 s_1 에 대한 뷰 동치 여부로 옳게 짝 지어진 것은? (단, r_i , w_i , c_i 는 각각 트랜잭션 T_i 의 읽기, 쓰기, 완료 연산을 의미한다)

s_1 : $r_1(a) \ r_1(b) \ w_1(a) \ w_1(b) \ c_1 \ r_2(b) \ r_2(c) \ w_2(b) \ c_2 \ r_3(c) \ r_3(b) \ c_3$
 s_2 : $r_1(a) \ r_1(b) \ r_2(b) \ r_2(c) \ r_3(c) \ w_1(a) \ w_1(b) \ w_2(b) \ r_3(b) \ c_1 \ c_2 \ c_3$

s_1 은 비연쇄 스케줄인가?

s_2 는 s_1 과 뷰 동치인가?

- ① 비연쇄 스케줄 뷰 동치
- ② 비연쇄 스케줄 뷰 동치 아님
- ③ 비연쇄 스케줄 아님 뷰 동치
- ④ 비연쇄 스케줄 아님 뷰 동치 아님

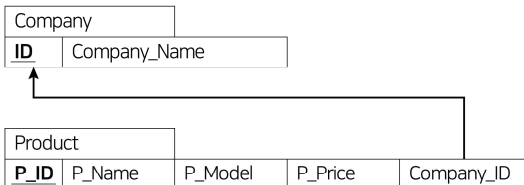
문 17. 다음 시스템 카탈로그(System Catalog)에 대한 설명으로 옳은 것으로만 묶은 것은?

- ㄱ. 시스템 그 자체에 관련이 있는 다양한 객체에 관한 정보를 포함하는 시스템 데이터베이스이기 때문에 일반 사용자는 SQL을 이용하여 내용을 검색해 볼 수 없다.
 ㄴ. 개체들은 기본 테이블, 뷰, 인덱스, 데이터베이스, 패키지, 접근 권한 등이 있다.
 ㄷ. 모든 데이터베이스에서 요구하는 정보는 동일하므로 데이터베이스의 종류에 관계없이 동일한 구조로 필요한 정보를 제공한다.
 ㄹ. 카탈로그에 저장된 정보를 메타 데이터(Meta-Data)라고도 한다.
 ㅁ. 트랜잭션 관리자를 통하여 사용자의 요구를 효율적인 형태로 변환하고 질의를 처리하는 좋은 전략을 모색할 수 있다.

- ① ㄱ, ㄴ ② ㄱ, ㄷ
 ③ ㄴ, ㄹ ④ ㄴ, ㅁ

문 18. 다음은 제품 데이터베이스의 스키마 다이어그램과 인스턴스를 나타낸 것이다. 데이터베이스를 구성하는 이 두 테이블을 대상으로 아래 결과를 출력하고자 한다. 이를 위한 SQL 질의문은?

Schema Diagram



Database Instance

P_ID	P_Model	P_Price	Company_ID	ID	Company_Name
1	CPU	150000	1	1	Intel
2	VGA	180000	2	2	AMD
3	SSD	136000	4	3	Nvidia
4	HDD	130000	5	4	Samsung
5	HDD	110000	5	5	WD
6	VGA	350000	2		
7	CPU	270000	2		
8	SSD	96000	4		
9	SSD	115000	5		
10	CPU	335000	2		
11	VGA	560000	2		
12	VGA	595000	2		
13	CPU	283500	1		
14	CPU	650300	1		
15	VGA	780000	1		
16	VGA	850000	1		

SQL 실행 결과

P_ID	P_Model	P_Price	Company_ID
6	VGA	350000	2
11	VGA	560000	2
12	VGA	595000	2
15	VGA	780000	1
16	VGA	850000	1

- ① SELECT *
 FROM Product Pro
 WHERE Pro.P_Model = 'VGA' AND Pro.P_Price IN
 (SELECT P.P_Price
 FROM Product P LEFT JOIN Company C
 ON (C.ID = P.Company_ID)
 WHERE C.Company_Name = 'AMD');
- ② SELECT *
 FROM Product Pro
 WHERE Pro.P_Model = 'VGA' AND Pro.P_Price NOT IN
 (SELECT P.P_Price
 FROM Product P LEFT JOIN Company C
 ON (C.ID = P.Company_ID)
 WHERE C.Company_Name = 'AMD');
- ③ SELECT *
 FROM Product Pro
 WHERE Pro.P_Model = 'VGA' AND Pro.P_Price > ANY
 (SELECT P.P_Price
 FROM Product P LEFT JOIN Company C
 ON (C.ID = P.Company_ID)
 WHERE C.Company_Name = 'AMD');
- ④ SELECT *
 FROM Product Pro
 WHERE Pro.P_Model = 'VGA' AND Pro.P_Price > ALL
 (SELECT P.P_Price
 FROM Product P LEFT JOIN Company C
 ON (C.ID = P.Company_ID)
 WHERE C.Company_Name = 'AMD');

문 19. 다음 데이터베이스 시스템의 질의최적화에 대한 설명으로 옳지 않은 것으로만 묶은 것은?

- ㄱ. DBMS는 SQL 질의문을 내부 표현(internal representation) 형태로 변환하여 질의최적화를 한다.
 ㄴ. DBMS는 질의최적화 과정에서 데이터사전에 저장된 정보를 활용한다.
 ㄷ. 규칙기반 최적화 기법(rule-based optimization)이 질의 실행계획(query execution plan)을 생성하는 데 비용이 거의 소요되지 않기 때문에 최근에 많이 활용되고 있다.
 ㄹ. 비용기반 최적화 기법(cost-based optimization)은 생성 가능한 모든 후보 질의실행계획 중에서 비용이 가장 적게 드는 질의실행계획을 생성한다.
 ㅁ. 조인 방법(join method) 중에서 해시 조인방법은 비동등 조인 질의(non-equi join query)에 적용하지 않는다.

- ① ㄱ, ㄴ ② ㄴ, ㄷ
 ③ ㄷ, ㄹ ④ ㄹ, ㅁ

문 20. 트랜잭션 고립 수준 중 REPEATABLE READ 모드에서 발생할 수 있는 문제를 모두 고르면?

- ㄱ. 유령데이터 읽기
 ㄴ. 반복불가능 읽기
 ㄷ. 오손읽기

- ① ㄱ ② ㄴ
 ③ ㄱ, ㄴ ④ ㄴ, ㄷ