1. 파일 시스템 사용 시 발생할 수 있는 문제점으로 가장 옳지 않은 것은?

① 데이터 중복(data redundancy)
② 데이터 고립(data isolation)
③ 동시 접속 이상(concurrent-access anomalies)
④ 트랜잭션 고립(transaction isolation)

2. 관계형 데이터베이스와 데이터베이스 관리 시스템(DBMS)에 대한 <보기>의 설명 중 옳은 것을 모두 고른 것은?

<보기>

ㄱ. 관계형 데이터베이스는 데이터를 행과 열로 구성된 테이블 형태로 논리적으로 표현하며, 실제 저장 구조는 DBMS 제품이나 저장 엔진에 따라 달라질 수 있다.
ㄴ. 시스템 카탈로그에는 테이블, 뷰, 제약 조건 및 권한 등에 관한 메타 데이터가 저장된다. 이 시스템 카탈로그는 DBMS가 아니라 사용자가 일반 업무 테이블과 같은 방식으로 직접 생성·관리해야 한다.
ㄷ. 데이터 조작 기능(DML)은 저장된 데이터의 조회·삽입·수정·삭제뿐 아니라 테이블 구조의 생성·변경까지 포함한다.
ㄹ. DBMS는 접근 권한 관리, 백업, 병행 제어 및 장애 회복 기능 등을 제공한다.

① ㄱ, ㄴ　② ㄱ, ㄹ
③ ㄱ, ㄷ, ㄹ　④ ㄴ, ㄷ, ㄹ

3. <보기>는 두 트랜잭션 T_1과 T_2가 동시에 실행되는 과정을 나타낸다. <보기>에서 T_1의 두 SELECT문의 조회 결과가 서로 다를 때 발생한 현상은?

<보기>

T_1	T_2
SELECT 봉급 FROM 직원 WHERE 아이디 = '123'; 실행결과: 40000	
	UPDATE 직원 SET 봉급 = 70000 WHERE 아이디 = '123'; COMMIT;
SELECT 봉급 FROM 직원 WHERE 아이디 = '123'; 실행결과: 70000	

① 반복 불가능 읽기(non-repeatable read)
② 팬텀 읽기(phantom read)
③ 갱신 손실(lost update)
④ 오손 읽기(dirty read)

4. <보기>에서 3단계 스키마 아키텍처(three-schema architecture)에 대한 설명으로 옳은 것을 모두 고른 것은?

<보기>

ㄱ. 물리적 데이터베이스로부터 사용자 응용들을 분리시킬 수 있다.
ㄴ. 실제의 데이터는 물리적 단계에만 존재한다.
ㄷ. 단계들 간에 데이터 요구와 결과를 변환하는 과정을 데이터 독립성이라고 한다.
ㄹ. 외부 스키마는 일반적으로 표현(representation) 데이터 모델을 사용하여 구현된다.
ㅁ. 물리적 데이터 독립성이 논리적 데이터 독립성보다 달성하기 더 어렵다.

① ㄱ, ㄴ, ㄷ　② ㄱ, ㄴ, ㄹ
③ ㄴ, ㄹ, ㅁ　④ ㄱ, ㄷ, ㄹ, ㅁ

5. 조인 선택률(join selectivity)에 대한 설명으로 가장 옳지 않은 것은?

<보기>

기호	의미
$\|R\|$	릴레이션 R의 튜플(tuple) 수
$R \bowtie_c S$	조인 조건 c를 적용한 R과 S 사이의 조인
R.A=S.B	조인 조건 c
×	카티션 곱(cartesian product)

① <보기>의 조인 조건 c에서 B가 R의 기본키 A를 참조하는 외래키이고, NOT NULL 제약 조건을 가진다면 조인 결과의 크기는 $|R|$이다.
② 조인 선택률은 $|R \bowtie_c S|/|R \times S|$로 계산할 수 있다.
③ <보기>의 조인 조건 c에서 A가 R의 기본키이면 조인 선택률은 $1/|R|$보다 작거나 같다.
④ <보기>의 조인 조건 c에서 B가 R의 기본키 A를 참조하는 외래키이고, NOT NULL 제약 조건을 가진다면 조인 선택률은 $1/|R|$과 같다.

6. <보기>와 같은 순서로 키(key) 값을 입력하여, 차수(degree)가 3인 ㉠B-트리와 ㉡B^+-트리를 구성하였을 때 루트 노드(root node)에 있는 키 값을 옳게 짝지은 것은? (단, 각 노드에 저장할 수 있는 키의 최대 개수는 2개이며, B^+-트리에서 노드 분할 시 오른쪽 노드에 더 많은 키를 배치한다.)

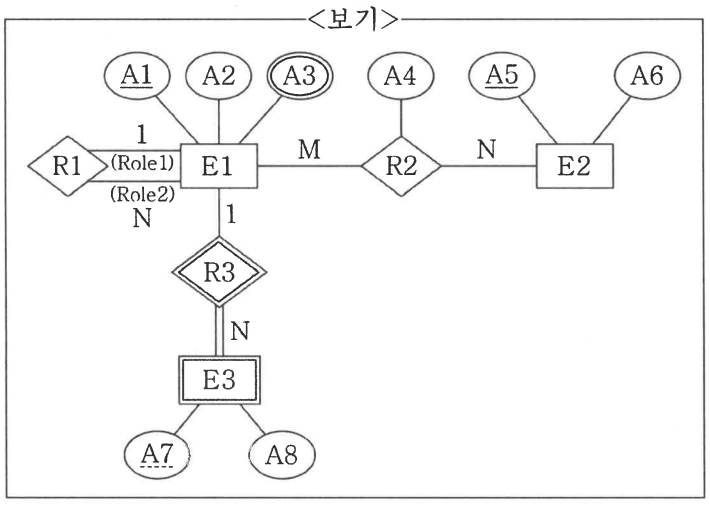
<보기>

키 입력 순서: 6, 8, 4, 5, 1, 9, 2

	㉠	㉡
①	[4, 6]	[4, 8]
②	[4, 6]	[6]
③	[5]	[6]
④	[6]	[4, 6]

7. <보기>의 개체-관계 다이어그램(E-R Diagram)을 관계형 데이터베이스 스키마(schema)로 변환한 결과로 가장 옳은 것은? (단, A1-Role#은 R1에 Role#으로 참여한 개체의 기본키를 참조하는 외래키를 의미한다.)

<보기>



① E1(A1, A2, A1-Role1), E1_A3(A1, A3), E2(A5, A6), E3(A1, A7, A8), R2(A1, A5, A4)

② E1(A1, A2), R1(A1-Role1, A1-Role2), E1_A3(A1, A3), E2(A5, A6), E3(A1, A7, A8), R2(A1, A5, A4)

③ E1(A1, A3, A2, A1-Role1), E2(A5, A6), E3(A1, A7, A8), R2(A1, A5, A4)

④ E1(A1, A2, A1-Role2), E1_A3(A1, A3), E2(A5, A6), E3(A1, A7, A8), R2(A1, A5, A4)

8. 릴레이션 R(A, B, C, D, E, G)와 함수 종속 $F = \{ A \rightarrow BC, C \rightarrow D, D \rightarrow E, BG \rightarrow A \}$가 주어질 때 후보키의 전체 집합으로 가장 옳은 것은?

① {A, BG} ② {A, C, BG}

③ {AG, BG} ④ {AG, CG}

9. 릴레이션 R(A, B, C, D)에서 A는 기본키이며, R을 <보기>와 같이 단편화하였다. 단편 F_1, F_2, F_3으로부터 원래 릴레이션 R을 항상 정확하게 재구성하는 관계 대수식으로 가장 옳은 것은? (단, 카티션 곱(cartesian product) 수행 시 동일한 이름의 속성은 릴레이션 이름으로 구분한다. C는 NULL을 허용하지 않으며, 관계 대수는 집합 의미론을 따른다.)

<보기>

- $F_1 = \pi_{A,B}(\sigma_{C='X'}(R))$
- $F_2 = \pi_{A,C,D}(\sigma_{C='X'}(R))$
- $F_3 = \sigma_{C \neq 'X'}(R)$

① $\pi_{F1.A,B,C,D}(F_1 \times F_2) \cup F_3$

② $(F_1 \bowtie F_3) \cup (F_2 \bowtie F_3)$

③ $(F_1 \bowtie F_2) \cup F_3$

④ $(F_1 \bowtie F_2) - F_3$

10. Steal/no-force 방식의 즉시 갱신(immediate update) 전략을 사용하는 데이터베이스 시스템에서 <보기>의 로그 기록을 사용하여 회복 과정을 수행한 내용으로 가장 옳은 것은? (단, 데이터 A, B, C의 초깃값은 각각 10, 30, 50이다.)

<보기>

<T1, start>
<T1, A, 10, 20>
<T2, start>
<T2, B, 30, 40>
<checkpoint, {T1, T2}>
<T3, start>
<T1, C, 50, 60>
<T1, commit>
<T3, A, 20, 80>
<T2, B, 40, 90>
<T2, commit>
-- 시스템 장애 발생 --

① 트랜잭션 T2는 검사점(checkpoint) 이전에 시작되었으므로 REDO 대상에서 제외되며 UNDO를 수행해야 한다.

② 로그 <T3, A, 20, 80>의 갱신 내용은 즉시 갱신 기법에 따라 디스크에 반영되었을 수 있으므로, A를 초깃값인 10으로 되돌려야 한다.

③ 회복 시스템이 지연 갱신(deferred update) 기법을 사용했다면, 회복 완료 후 B의 최종 상태는 30이 된다.

④ 모든 회복 절차가 완료된 후, 데이터의 최종 상태는 A = 20, B = 90, C = 60이다.

11. <보기 1>은 학생의 수강 정보를 저장하는 ENROLLMENT 릴레이션이다. <보기 2>의 조건을 참고할 때, 관계형 데이터 모델의 키에 대한 설명으로 가장 옳지 않은 것은?

<보기 1>

ENROLLMENT

EnrollmentID	StudentID	CourseID	Grade
E001	S101	C001	A
E002	S101	C002	B+
E003	S102	C001	A-
E004	S103	C003	B
E005	S102	C004	A

<보기 2>

- EnrollmentID는 수강 정보, StudentID는 학생 정보, CourseID는 과목 정보를 의미한다.
- EnrollmentID는 각 수강 정보를 유일하게 구분한다.
- 한 학생은 여러 과목을 수강할 수 있으며 한 과목은 여러 학생이 수강할 수 있다.
- 동일한 학생이 동일한 과목을 중복하여 수강하지 않는다.
- EnrollmentID, StudentID, CourseID에는 NULL 값이 없다.

① EnrollmentID는 후보키이다.

② (StudentID, CourseID)는 각 튜플을 유일하게 식별할 수 있는 후보키이다.

③ StudentID는 후보키가 아니다.

④ (EnrollmentID, StudentID)는 각 튜플을 유일하게 식별할 수 있는 후보키이다.

12. <보기 1>은 회사의 부서, 직원 및 업무 정보를 관리하는 관계형 데이터베이스의 일부이다. 밑줄 친 속성은 기본키이며, 외래키에는 <보기 2>와 같은 참조 동작이 설정되어 있다. <보기 3>의 SQL문을 (가)부터 (다)까지 순서대로 실행했을 때의 실행 결과로 가장 옳지 않은 것은? (단, 모든 SQL문은 정상적으로 실행된다고 가정한다.)

<보기 1>

DEPARTMENT

DeptID	DeptName
10	Research
20	Sales

EMPLOYEE

EmpID	EmpName	DeptID
101	Kim	10
102	Lee	10
103	Park	20
104	Choi	20

TASK

TaskID	EmpID	TaskName
T1	101	Analysis
T2	102	Design
T3	102	Report
T4	103	Sales
T5	104	Support

<보기 2>

```
EMPLOYEE.DeptID
    REFERENCES DEPARTMENT(DeptID)
    ON UPDATE CASCADE
    ON DELETE CASCADE

TASK.EmpID
    REFERENCES EMPLOYEE(EmpID)
    ON UPDATE CASCADE
    ON DELETE CASCADE
```

<보기 3>

(가) UPDATE DEPARTMENT SET DeptID = 30 WHERE DeptID = 10;
(나) DELETE FROM EMPLOYEE WHERE EmpID = 102;
(다) DELETE FROM DEPARTMENT WHERE DeptID = 20;

① (가)를 실행하면 EmpID가 101과 102인 직원의 DeptID가 30으로 연쇄적으로 변경된다.

② (가)를 실행해도 TASK 테이블의 EmpID 값은 연쇄적으로 변경되지 않는다.

③ (나)를 실행하면 TASK 테이블에서 1개의 튜플이 연쇄적으로 삭제된다.

④ (다)까지 모두 실행한 후에는 각 테이블에 1개의 튜플만 남는다.

13. <보기>는 관계 R에 관한 설명이다. 관계 R을 분해한 결과 중, 각 분해 관계가 제3정규형(3NF)을 만족하는 것으로 가장 옳은 것은? (단, R은 제2정규형을 만족하며, 제시된 함수 종속성과 이들로부터 추론되는 종속성만 고려한다.)

<보기>

- 공공기관은 민원 처리 현황을 관계 R로 관리한다.
- 관계: R(민원번호, 담당자번호, 담당자명, 부서코드, 부서명, 처리기한)
- 기본키: 민원번호
- 함수 종속성:
  민원번호 → 담당자번호, 처리기한
  담당자번호 → 담당자명, 부서코드
  부서코드 → 부서명

① R1(민원번호, 담당자번호, 담당자명, 처리기한), R2(담당자번호, 부서코드), R3(부서코드, 부서명)
② R1(민원번호, 담당자번호, 부서코드, 처리기한), R2(담당자번호, 담당자명), R3(부서코드, 부서명)
③ R1(민원번호, 담당자번호, 처리기한), R2(담당자번호, 담당자명, 부서코드, 부서명)
④ R1(민원번호, 담당자번호, 처리기한), R2(담당자번호, 담당자명, 부서코드), R3(부서코드, 부서명)

14. 데이터베이스 관리자 DBA는 문서 테이블의 소유자이다. <보기>의 SQL문을 순서대로 모두 실행한 후, 마지막 REVOKE문으로 인해 SELECT 권한을 잃는 사용자를 모두 고른 것은? (단, 처음에는 DBA 이외의 사용자가 문서 테이블에 대한 권한을 가지고 있지 않으며, 표준 SQL의 권한 철회 규칙을 따른다.)

<보기>

```
DBA> GRANT SELECT ON 문서 TO Kim WITH GRANT OPTION;
DBA> GRANT SELECT ON 문서 TO Lee WITH GRANT OPTION;
Kim> GRANT SELECT ON 문서 TO Park WITH GRANT OPTION;
Lee> GRANT SELECT ON 문서 TO Park WITH GRANT OPTION;
Park> GRANT SELECT ON 문서 TO Choi WITH GRANT OPTION;
Kim> GRANT SELECT ON 문서 TO Han WITH GRANT OPTION;
Han> GRANT SELECT ON 문서 TO Yoon;
Park> GRANT SELECT ON 문서 TO Min;
DBA> REVOKE SELECT ON 문서 FROM Kim CASCADE;
```

① Kim, Han
② Kim, Han, Yoon
③ Kim, Park, Han, Yoon
④ Kim, Park, Choi, Han, Yoon, Min

15. instructor 릴레이션이 <보기 1>과 같을 때, <보기 2>의 SQL문을 수행한 결과는?

<보기 1>

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

<보기 2>

```
select count(*) from instructor T, instructor S
where T.salary>S.salary and T.salary>90000 and
      S.dept_name = 'Comp. Sci.';
```

① 1 ② 3
③ 5 ④ 7

16. 데이터베이스 스키마(schema)가 <보기>와 같다. SQL 문장 중에서 실행 결과가 나머지와 다른 것은? (단, instructor는 강사 정보를, teaches는 강의 담당 내역을 저장하며, 강의를 담당하지 않는 강사가 적어도 한 명은 존재한다고 가정한다.)

<보기>

instructor = (id, name, dept_name, salary)
teaches = (id, course_id, section_id, semester, year)

① select name from instructor natural join teaches;
② select name from instructor, teaches where instructor.id = teaches.id;
③ select name from instructor natural left outer join teaches;
④ with temp(id, name) as (select id, name from instructor) select name from temp join teaches using (id);

17. <보기 1>의 '판매' 릴레이션에서 기본키는 (상품번호, 지점)이다. <보기 2>의 '상품현황'은 결과를 별도로 저장하지 않는 일반 뷰이다. <보기 3>과 같이 질의 Q1을 실행한 후, 삭제문과 삽입문을 순서대로 실행하고 질의 Q2를 실행할 때, Q1과 Q2의 결과를 옳게 짝지은 것은?

<보기 1>

판매

상품번호	지점	수량
P1	서울	3
P1	부산	2
P2	서울	4
P2	부산	1
P3	서울	6

<보기 2>

```
CREATE VIEW 상품현황(상품번호, 지점수, 총수량) AS
    SELECT 상품번호, COUNT(DISTINCT 지점), SUM(수량)
    FROM 판매
    GROUP BY 상품번호;
```

<보기 3>

```
-- 질의 Q1
    SELECT 상품번호
    FROM 상품현황
    WHERE 지점수 = 2 AND 총수량 >= 5
    ORDER BY 상품번호;

-- 삭제문
    DELETE FROM 판매
    WHERE 상품번호 = 'P2' AND 지점 = '부산';

-- 삽입문
    INSERT INTO 판매(상품번호, 지점, 수량)
    VALUES ('P3', '부산', 1);

-- 질의 Q2
    SELECT 상품번호
    FROM 상품현황
    WHERE 지점수 = 2 AND 총수량 >= 5
    ORDER BY 상품번호;
```

	Q1	Q2
①	P1, P2	P1, P2
②	P1, P2	P1, P3
③	P1, P3	P1, P2
④	P1, P2, P3	P1, P3

18. 분산 데이터베이스의 투명성(transparency)에 대한 설명으로 가장 옳지 않은 것은?

① 위치 투명성은 사용자가 어느 사이트에 데이터가 저장되는지 몰라도 접근할 수 있게 한다.
② 중복 투명성은 동일 데이터가 여러 사이트에 복제되어도 사용자가 이를 의식하지 않게 한다.
③ 단편화 투명성은 릴레이션이 여러 조각으로 나뉘어 저장되어도 전체 릴레이션처럼 사용할 수 있게 한다.
④ 분산 투명성은 사용자가 항상 모든 단편의 물리적 위치를 명시해야 함을 의미한다.

19. <보기>는 데이터 항목 A, B, C에 대한 트랜잭션의 잠금 상태이다. 현재 발생한 교착 상태의 모든 순환을 하나의 트랜잭션만 철회하여 제거하려고 한다. 철회해야 할 트랜잭션에 해당하는 것은? (단, 공유 잠금(shared lock) S끼리는 동시에 허용되며, 배타 잠금(exclusive lock) X는 다른 모든 잠금과 충돌한다.)

<보기>

트랜잭션	현재 보유한 잠금	새로 요청한 잠금
T1	S(A), X(B)	X(C)
T2	S(A), S(C)	X(B)
T3	S(C)	X(A)

① T1
② T2
③ T3
④ 어느 트랜잭션을 철회해도 모든 순환이 제거된다.

20. 한 문화 콘텐츠 플랫폼은 데이터의 특성과 이용 방식에 따라 <보기>와 같이 NoSQL 데이터베이스를 활용하려고 한다. NoSQL의 특징에 대한 설명으로 가장 옳지 않은 것은?

<보기>

- 사용자 식별자를 키로 하여 로그인 상태를 저장한다.
- 콘텐츠 메타 데이터는 JSON 문서로 저장하며, 콘텐츠 유형에 따라 저장되는 항목이 다르다.
- 대량의 이용 기록은 행별로 저장하되, 기록의 종류에 따라 각 행을 구성하는 속성이 다르다.
- 사용자와 콘텐츠 간의 팔로우 및 추천 관계를 관리한다.

① 로그인 상태를 키-값(key-value) 데이터베이스에 저장하면 키를 이용하여 값 전체를 검색할 수 있지만, 값의 일부나 값의 내용을 조건으로 검색하려면 별도의 처리가 필요할 수 있다.

② 콘텐츠 메타 데이터를 문서(document) 기반 데이터베이스에 저장하는 경우 동일한 컬렉션에 포함된 모든 문서는 반드시 동일한 구조와 항목을 가져야 한다.

③ 이용 기록을 컬럼(column) 기반 데이터베이스에 저장하면 행 키로 각 행을 식별하고, 컬럼 패밀리 내 컬럼의 구성은 행마다 다르게 할 수 있다.

④ 팔로우 및 추천 관계를 그래프(graph) 기반 데이터베이스에 저장하면 데이터는 노드에 저장하고 데이터 간의 관계는 간선으로 표현할 수 있다.