

데이터베이스론 (7급)

(과목코드 : 029)

2024년 군무원 채용시험

응시번호 :

성명 :

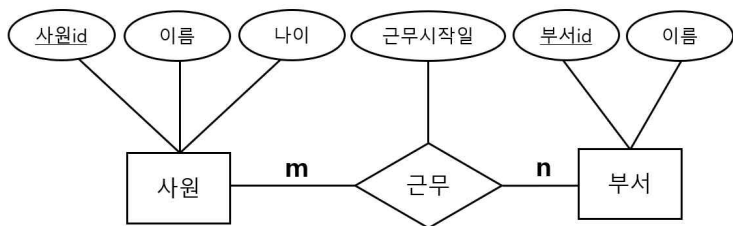
1. 다음 데이터베이스 추상화(abstraction)와 관련된 세 단계 스키마에 대한 설명 중 가장 적절하지 않은 것은?

- ① 데이터베이스 인덱스(index)는 외부 스키마(external schema)에 해당한다.
- ② 개념 스키마(conceptual schema)는 데이터베이스에 저장되는 전체 데이터를 기술한다.
- ③ 외부 스키마는 뷰(view)와 동의어로 볼 수 있다.
- ④ 외부 스키마는 사용자나 응용프로그램의 요구에 필요한 데이터에 접근하도록 만들어진 스키마이다.

2. 다음 중 관계 모델(relational model)에서 속성(attribute)과 같은 의미로 사용되는 용어들로 제시된 것은?

- ① 튜플(tuple), 행(row)
- ② 열(column), 필드(field)
- ③ 필드(field), 레코드(record)
- ④ 필드(field), 튜플(tuple)

3. 아래 개체-관계 다이어그램(E-R diagram)에 관한 설명으로 다음 중 가장 적절하지 않은 것은?



- ① 근무는 직원과 부서 사이의 개체를 의미한다.
- ② 부서id는 부서 개체를 구별해주는 식별자(identifier)이다.
- ③ 한 직원은 여러 부서에서 일할 수 있다.
- ④ 각 직원의 직원id 값은 서로 달라야만 한다.

4. 다음 중 약개체(weak entity)에 대한 설명으로 가장 적절하지 않은 것은?

- ① 부분키(partial key)를 가진다.
- ② 소유자(owner) 개체가 삭제되면 해당 약개체도 삭제된다.
- ③ 소유자 개체의 키(key)와 부분키를 복합해 식별한다.
- ④ 소유자 개체와 다대다(many-to-many) 관계를 가진다.

5. 아래 SQL문을 사용한 수강 테이블이 존재한다고 할 때, 다음 중 가장 적절하지 않은 것은?

```
CREATE TABLE 수강 ( 학번 CHAR(10),
과목번호 CHAR(10),
학점 NUMBER(3),
PRIMARY KEY (학번, 과목번호),
FOREIGN KEY 학번 REFERENCES 학생
ON DELETE CASCADE );
```

- ① 학생이 삭제되면 해당 학생의 수강 기록도 사라지게 된다.
- ② 동일한 학번이 수강 테이블에 여러 번 나올 수 없다.
- ③ 과목번호는 고정 문자열 데이터 타입을 가진다.
- ④ 학생 테이블에는 반드시 학번 속성이 존재해야 한다.

6. R1과 R2는 각각 10개, 20개의 레코드를 가지고 있다. R1 \bowtie R2 관계 대수식과 R1 \cap R2 관계 대수식의 결과 테이블이 가질 수 있는 최대 레코드 경우의 개수를 순서대로 나타낸 것은?

- ① 20, 10
- ② 30, 10
- ③ 200, 10
- ④ 200, 20

7. 다음 중 아래 SQL문에 해당하는 관계 대수식을 가장 적절하게 나타낸 것은?

```
select ename from emp, dept where sal > 1000
```

- ① $\Pi_{ename} \delta_{sal>1000} (emp \times dept)$
- ② $\Pi_{sal>1000} \delta_{ename} (emp \times dept)$
- ③ $\delta_{ename} \Pi_{sal>1000} (emp \times dept)$
- ④ $\delta_{sal>1000} \Pi_{ename} (emp \times dept)$

8. 아래와 같은 학생 테이블을 고려할 때, 다음 중 각 전공별로 해당 전공 학생의 평균 나이와 최대 나이를 구하는 SQL문으로 가장 적절한 것은?

```
학생(학번, 이름, 나이, 전공)
```

- ① `select 전공, avg(나이), max(나이) from 학생 order by 전공`
- ② `select 전공, 나이(avg, max) from 학생 group by 전공`
- ③ `select 전공, avg(나이), max(나이) from 학생`
- ④ `select 전공, avg(나이), max(나이) from 학생 group by 전공`

9. 아래의 학생 테이블의 기본키에 대하여 B+-트리 인덱스가 생성되어 있다고 할 때, 다음 중 가장 적절한 것은?

```
create table 학생 ( sid number primary key, sname varchar(20), major char(10) );
```

- ① 테이블에 하나의 B+-트리 인덱스가 이미 존재하기 때문에, major 필드를 탐색키로 하는 추가 인덱스는 생성할 수 없다.
- ② 테이블 레코드가 sid 순서에 따라 물리적으로 저장되었다면, sid를 탐색키로 하는 B+-트리 인덱스는 클러스터 인덱스 성질을 갖게 된다.
- ③ sname 필드는 가변형이므로 B+-트리 인덱스를 생성할 수 없다.
- ④ sid를 탐색키로 하는 B+-트리의 리프노드가 모두 동일한 레벨에 위치하지는 않는다.

10. 관계 R(A,B,C,D)의 키(key)가 AB라고 할 때, 다음의 함수 종속(functional dependency) 중 성립한다고 볼 수 없는 것은?

- ① $AB \rightarrow A$
- ② $D \rightarrow C$
- ③ $AB \rightarrow CD$
- ④ $ABC \rightarrow D$

11. 관계 R(A, B, C, D, E)의 함수 종속(functional dependency) 집합 FD가 아래와 같을 때, 다음 중 가장 적절한 것은?

```
FD: {AEB  $\rightarrow$  D, B  $\rightarrow$  C}
```

- ① AB는 R의 키(key)이다.
- ② R은 키가 존재하지 않는다.
- ③ R은 BCNF를 따르지 아니한다.
- ④ R은 정보 중복으로 볼 수 있는 문제가 없다.

12. 엄격 2단계 로킹 규약(strict 2-phase locking)에 따른 동시성 제어(concurrency control)를 하고, 두 개의 트랜잭션 T1, T2가 다음의 연산을 왼쪽부터 오른쪽으로 순차적으로 요청한다고 할 때, 다음 중 연산에 필요한 로크(lock)를 얻지 못해 연산을 바로 수행하지 못하고 로크(lock)를 얻을 때까지 기다리게 되는 연산은?(단, 여기서 RN(X)는 트랜잭션 TN이 X를 읽는 연산을, WN(X)는 트랜잭션 TN이 X를 쓰는 연산을 나타낸다.)

R1(A) R2(A) W1(B) W2(A) W1(C) ...

- ① R2(A)
- ② W1(B)
- ③ W2(A)
- ④ W1(C)

13. 동시성 제어(concurrency control)에 대한 설명으로 가장 적절한 것은?

- ① 2단계 로킹 규약(2-phase locking protocol)은 스케줄의 직렬성(serializability)을 보장하지 않을 수 있지만, 엄격(strict) 2단계 로킹 규약은 스케줄의 직렬성을 항상 보장한다.
- ② 로킹(locking)기반 동시성 제어에서 로킹 단위를 크게 하면 동시성 수준(concurrency level)은 낮아지게 된다.
- ③ 낙관적(optimistic) 동시성 제어기법에서는 로크(lock)를 사용하지 않으므로 동시성 제어에 들어가는 오버헤드가 없다.
- ④ 2단계 로킹 규약은 교착상태(deadlock)가 일어나지 않도록 보장해준다.

14. 다음은 고객들이 구매한 장바구니 테이블이다.

구매번호	장바구니
1	빵, 버터, 커피
2	사탕, 빵, 버터
3	사탕, 과자, 버터
4	빵, 커피, 과자
5	빵, 버터, 우유

“빵을 사면 버터도 같이 구매한다”라는 연관 규칙의 지지도(support)와 신뢰도(confidence) 값이 모두 적절한 것은?

- ① 지지도=0.75, 신뢰도=0.6
- ② 지지도=0.5, 신뢰도=0.6
- ③ 지지도=0.6, 신뢰도=0.75
- ④ 지지도=0.6, 신뢰도=0.5

15. 데이터베이스의 조인 연산(Join Operation)에 대한 설명으로 가장 적절하지 않은 것은?

- ① 자연 조인(Natural Join)은 Equi-Join 결과에서 중복되는 컬럼을 하나만 남겨두는 조인이다.
- ② 내부 조인(Inner Join)은 WHERE절이나 HAVING절 내부에 서브 쿼리를 포함하는 형태를 뜻한다.
- ③ 외부 조인(Outer Join)은 조인 조건에 만족되는 튜플 조합에 참여하지 않는 튜플도 결과에 포함시킨다.
- ④ 상호 조인(Cross Join)의 결과는 두 테이블의 튜플 수를 곱한 수만큼 생성된다.

16. 다음 중 데이터베이스 트리거(Trigger)에 대한 설명으로 가장 적절하지 않은 것은?

- ① 지정된 이벤트가 발생하면 자동으로 실행된다.
- ② 파라미터(Parameter)를 정의하고 실행 시 전달 받을 수 있다.
- ③ 지정된 테이블의 행이 삭제된 직후에도 실행될 수 있다.
- ④ 프로그램이나 사용자에게 의한 직접 실행은 불가능하다.

17. 아래의 주문 테이블에 대하여 고객ID 111 고객과 고객ID 222 고객의 주문상품 중 공통되는 주문상품을 출력하기 위해 가장 적절한 SQL 질의는? (단, 밑줄 친 속성은 기본키이다.)

주문(고객ID, 주문날짜, 주문상품, 주문수량)

- ① SELECT DISTINCT 주문상품
FROM 주문
WHERE 고객ID = 111 AND 고객ID = 222;
- ② SELECT DISTINCT 주문상품
FROM 주문 AS A, 주문 AS B
WHERE A.고객ID = 111 AND B.고객ID = 222
AND A.주문상품 = B.주문상품;
- ③ SELECT DISTINCT 주문상품
FROM 주문
WHERE 고객ID = 111
AND INTERSECT (SELECT 주문상품
FROM 주문 WHERE 고객ID = 222);
- ④ SELECT DISTINCT 주문상품
FROM 주문
WHERE 고객ID = 111
AND 주문상품 IN (SELECT 주문상품
FROM 주문 WHERE 고객ID = 222);

18. 다음은 직원 테이블과 판매 테이블에 대하여 “냉장고를 판매한 직원의 ID와 이름”을 구하는 질의이다. 질의문의 빈칸 (ㄱ), (ㄴ)에 들어갈 내용으로 가장 적절한 것은? (단, 밑줄 친 속성은 기본키이다.)

직원(직원ID, 직원이름, 소속부서)
판매(직원ID, 판매날짜, 판매시간, 상품명, 수량)

SELECT DISTINCT 직원ID, 직원이름
FROM 직원 AS E
WHERE (ㄱ) (SELECT 직원ID
FROM 판매 AS S
WHERE 상품명 = '냉장고'
AND (ㄴ));

- ① (ㄱ) EXISTS (ㄴ) 직원ID = E.직원ID
- ② (ㄱ) NOT EXISTS (ㄴ) 직원ID <> E.직원ID
- ③ (ㄱ) S.직원ID IN (ㄴ) 직원ID = E.직원ID
- ④ (ㄱ) 직원ID NOT IN (ㄴ) 직원ID <> E.직원ID

19. 아래 두 테이블 R과 S 사이에 참조 무결성 제약(Referential Integrity Constraint)이 적용되고 있으며, 테이블 R의 기본키 R.A를 테이블 S의 외래키 S.A가 참조한다고 할 때, 다음 중 테이블 R과 S에 대한 SQL 명령의 설명으로 가장 적절한 것은?

R(A, B, C, D)
S(X, Y, A)

- ① CREATE TABLE 명령으로 테이블 R을 정의할 때 참조 무결성 제약을 설정한다.
 - ② 테이블 R에 새로운 행을 추가하는 INSERT 명령은 참조 무결성에 위반될 수 있다.
 - ③ 테이블 S의 기본키 이외의 컬럼에 대한 UPDATE 명령은 참조 무결성과 무관하다.
 - ④ 테이블 S의 행을 삭제하는 DELETE 명령은 참조 무결성과 무관하다.
20. 참조 무결성이 유지되고 있는 아래 고객 테이블과 주문 테이블에 대하여, 총 고객 수와 한번 이상 주문한 고객 수를 구하고자 한다. SQL 문장의 빈칸에 들어갈 내용으로 가장 적절한 것은? (단, 밑줄 친 컬럼은 기본키이다.)

고객(고객ID, 고객이름, 연락처, 도시)
주문(주문ID, 주문날짜, 상품명, 수량, 고객ID)

SELECT (ㄱ) AS 총고객수,
(ㄴ) AS 주문고객수
FROM 고객 (ㄷ) 주문 ON
고객.고객ID = 주문.고객ID;

- ① (ㄱ) COUNT(고객.고객ID)
(ㄴ) COUNT(주문.고객ID)
(ㄷ) LEFT OUTER JOIN
- ② (ㄱ) COUNT(고객.고객ID)
(ㄴ) COUNT(주문.고객ID)
(ㄷ) RIGHT OUTER JOIN
- ③ (ㄱ) COUNT(DISTINCT 고객.고객ID)
(ㄴ) COUNT(DISTINCT 주문.고객ID)
(ㄷ) LEFT OUTER JOIN
- ④ (ㄱ) SUM(DISTINCT 고객.고객ID)
(ㄴ) SUM(DISTINCT 주문.고객ID)
(ㄷ) RIGHT OUTER JOIN

21. 아래 운항 테이블은 항공사의 운항기록 데이터의 일부를 보여주고 있다. 각 운항은 고유한 운항번호로 구별된다. 동일한 출/도착지에 대해 다른 항공기가 운항할 수도 있고, 같은 날짜에 동일한 출/도착지로 여러 편이 운항될 수도 있다. 또한, 동일한 출/도착 공항간의 운항기준거리는 동일하다. 데이터 중복이 최소화되도록 테이블을 정규화했을 때, 현재 테이블이 충족하는 정규형과 정규화 결과 테이블 스키마로서 가장 적절한 것은? (단, 밑줄 친 속성은 기본키이다.)

운항번호	날짜	항공기	출발공항	도착공항	운항기준거리
12345	2024/3/2	보잉747	ICN	ORD	10,544
23334	2024/3/3	보잉707	NRT	ICN	1,260
32432	2024/3/5	에어버스 A300	NRT	ICN	1,260
55445	2024/3/9	보잉747	ICN	SFO	9,107
90444	2024/3/9	보잉747	ICN	SFO	9,107

- ① 현재 제1정규형 만족, 제2정규형 위반
운항(운항번호, 날짜, 항공기)
공항거리(운항번호, 출발공항, 도착공항, 운항기준거리)
- ② 현재 제1정규형 만족, 제2정규형 위반
운항(운항번호, 날짜, 항공기, 출발공항, 도착공항)
공항거리(출발공항, 도착공항, 운항기준거리)
- ③ 현재 제2정규형 만족, 제3정규형 위반
운항(운항번호, 날짜, 항공기)
공항거리(운항번호, 출발공항, 도착공항, 운항기준거리)
- ④ 현재 제2정규형 만족, 제3정규형 위반
운항(운항번호, 날짜, 항공기, 출발공항, 도착공항)
공항거리(출발공항, 도착공항, 운항기준거리)

22. 다음 중 데이터베이스 트랜잭션이 지켜야 할 바람직한 특성에 대한 설명으로 가장 적절하지 않은 것은?

- ① 하나의 트랜잭션은 전체가 완전히 실행되거나 전혀 실행되지 않아야 한다.
- ② 다른 트랜잭션의 간섭 없이 정상 실행되었을 때 트랜잭션 실행 전과 후의 데이터베이스 상태에 대한 일관성이 유지되어야 한다.
- ③ 트랜잭션 실행 중의 데이터 변경 내용은 즉시 다른 트랜잭션들과 공유되어야 한다.
- ④ 완료된 트랜잭션의 데이터 변경 내용은 이후 시스템 장애가 발생하더라도 손실되어서는 안된다.

23. 아래 고객 테이블에 대해 다음의 함수 종속 (functional dependency)이 성립한다고 할 때, 보기의 설명 중 가장 적절한 것은?

고객(고객ID, 성명, 주소, 멤버십등급, 구매한도, 대출한도)

성립하는 함수 종속:
 {고객ID} → {성명, 주소, 멤버십등급}
 {멤버십등급} → {구매한도, 대출한도}

- ① 고객 테이블의 기본키로 (고객ID, 멤버십등급)이 적합하다.
- ② 멤버십등급 속성은 고객ID 속성을 참조하는 외래키(foreign key)로 지정된다.
- ③ (구매한도, 대출한도) 속성은 고객ID 속성에 함수적 종속(functionally dependent)된다.
- ④ 함수 종속 {구매한도, 대출한도} → {멤버십등급}이 성립한다.

24. 데이터베이스 트랜잭션의 동시 실행에 대한 통제가 이루어지지 않을 때, 두 트랜잭션 T1, T2의 실행이 예시하는 동시성 문제로 가장 적절한 것은?

트랜잭션 T1	트랜잭션 T2
<pre>read_item(X); X = X + 1; write_item(X); commit;</pre>	<pre>read_item(X); X = X - 1; sum = sum + X; write_item(X); commit;</pre>

- ① 갱신 손실(Lost Update)
- ② 오손 읽기(Dirty Read)
- ③ 합산 오류(Incorrect Summary)
- ④ 반복 불가 읽기(Unrepeatable Read)

25. 아래 표의 순서대로 SQL 명령이 모두 실행되었을 때, 이에 대한 설명으로 가장 적절한 것은?

순서	실행자	명령
1	계정1	GRANT SELECT ON 고객 TO 계정2 WITH GRANT OPTION;
2	계정2	GRANT SELECT ON 고객 TO 계정3 WITH GRANT OPTION;
3	계정1	GRANT SELECT ON 고객 TO 계정4;
4	계정2	GRANT SELECT ON 고객 TO 계정4;
5	계정1	REVOKE SELECT ON 고객 FROM 계정2;

- ① [계정1]은 고객 테이블에 대한 SELECT 명령의 실행 권한이 없다.
- ② [계정2]는 고객 테이블에 대한 SELECT 명령의 실행 권한을 다른 계정에 부여할 수 있다.
- ③ [계정3]은 고객 테이블에 대한 SELECT 명령의 실행 권한을 다른 계정에 부여할 수 있다.
- ④ [계정4]는 고객 테이블에 대한 SELECT 명령의 실행 권한이 있다.