

데이터베이스론

2022년도 국가공무원 5급(기술) 공개경쟁채용 제2차시험

응시번호 :

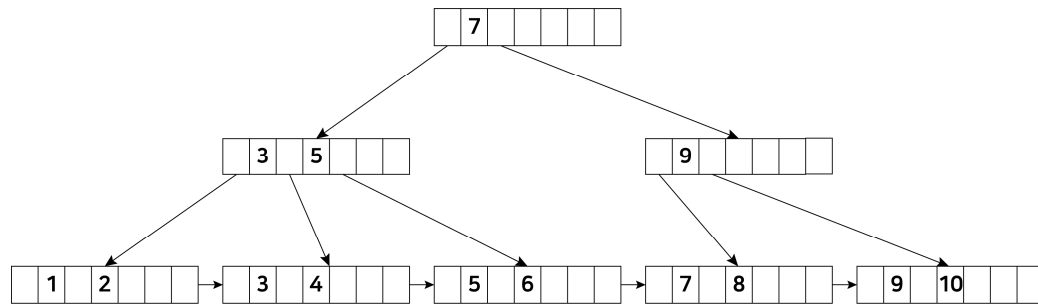
성명 :

제 1 문. 일반 대학교에서 재학생들의 수강신청 데이터를 저장할 수 있도록 기존 데이터베이스에 “수강신청”과 “수강신청내역” 테이블을 추가로 생성하려고 한다. 아래 제시된 <설계 고려사항>, <질의 빈도수 순위>, <이미 생성되어 있는 관련 테이블> 및 <새로 생성할 테이블>을 참고하여 물음에 답하시오. (단, 각 테이블의 기본키는 밑줄로 표시되어 있고, 기본키는 기본인덱스를 자동생성하는 것으로 가정한다)

(총 30점)

제 2 문. 데이터베이스의 인덱스 구조에 대한 다음 물음에 답하시오. (총 15점)

- 1) B-트리 인덱스와 B⁺-트리 인덱스의 차이점을 키값의 저장 형태 측면과 릴레이션 레코드에 접근하는 측면에서 설명하시오. (5점)
- 2) B⁺-트리 인덱스 구조와 해싱 인덱스 구조는 질의 형태에 따라 각각 장점을 가지고 있다. 두 가지 인덱스 구조의 장점을 SQL의 SELECT 문장을 예로 들어 비교하여 설명하시오. (5점)
- 3) 다음 B⁺-트리에서 키 7을 삭제한 후의 B⁺-트리를 그리시오. (단, B⁺-트리의 차수는 4이다) (5점)



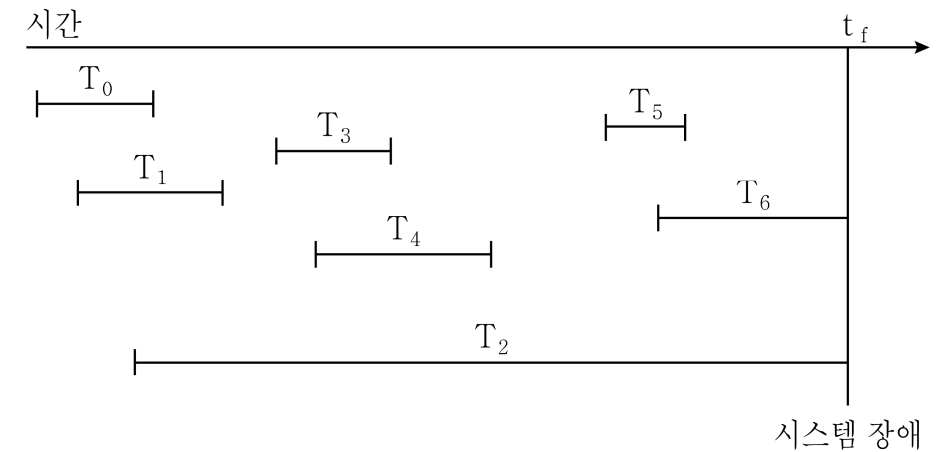
제 3 문. 릴레이션 R(A, B, C, D)에 대한 아래의 세 가지 물음 (a) ~ (c)를 주어진 함수종속 집합 F에 따라 각각 답하시오. (총 20점)

- (a) R의 후보키를 모두 구하시오.
 (b) R이 만족하는 최대 정규형을 판별하고 그 이유를 설명하시오.
 (단, 적용될 수 있는 정규형의 범위는 1NF, 2NF, 3NF, BCNF이다)
 (c) R을 함수종속성 보존 BCNF로 분해하되, 할 수 없는 경우 그 이유를 설명하시오.

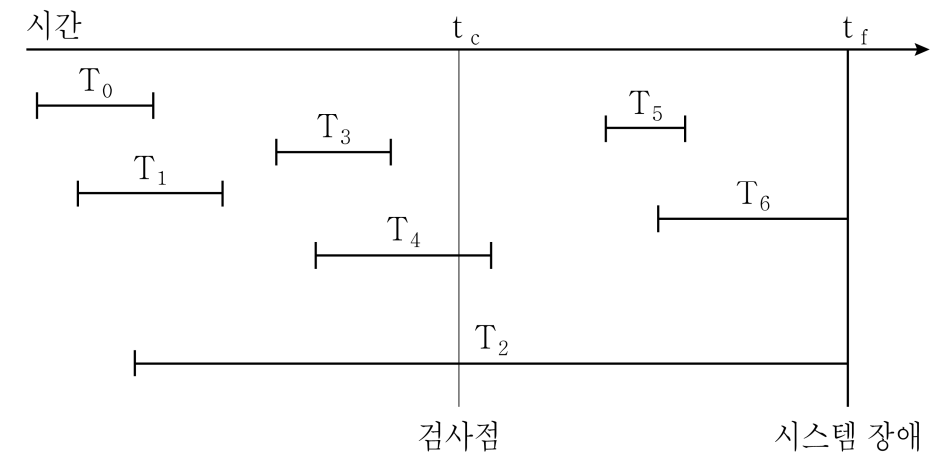
- 1) F = {A → D, B → C} (6점)
- 2) F = {A → B, A → C, BC → D} (6점)
- 3) F = {AB → C, AB → D, C → A, D → B} (8점)

제 4 문. 데이터베이스 관리시스템에 장애가 발생하여 재시작되었을 때 디스크의 로그에 [Trans-ID, commit] 로그 레코드가 존재하면 해당 트랜잭션을 재수행하고, 존재하지 않으면 트랜잭션을 취소한다. 검사점(checkpoint)은 시스템 장애가 발생한 후 재시작되었을 때 재수행하거나 취소해야 하는 트랜잭션들의 수를 줄여준다. 회복작업에 대한 다음 물음에 답하시오. (총 15점)

- 1) 다음 그림과 같이 검사점을 사용하지 않았을 때, 각 트랜잭션(T₀ ~ T₆)의 실행 상태와 필요한 회복작업에 대해 설명하시오. (7점)



- 2) 다음 그림과 같이 검사점을 사용했을 때, 각 트랜잭션(T₀ ~ T₆)의 실행 상태와 필요한 회복작업에 대해 설명하시오. (8점)



제 5 문. 데이터베이스의 테이블 생성과 초기 데이터 삽입이 다음과 같다.

<pre>CREATE TABLE USER_BUCKET (Fname VARCHAR(20), Lname VARCHAR(20), BucketID INTEGER DEFAULT 0);</pre>	<pre>CREATE TABLE BUCKET_COUNT (BucketID INTEGER, BucketCount INTEGER);</pre>
<pre>INSERT INTO BUCKET_COUNT VALUES(0, 0); INSERT INTO BUCKET_COUNT VALUES(1, 0);</pre>	

이 데이터베이스에 대한 SQL-99 트리거를 다음과 같이 생성하였다.

<pre>CREATE TRIGGER USER_BEFORE_TRIGGER BEFORE INSERT ON USER_BUCKET REFERENCING NEW ROW AS NEWTUPLE FOR EACH ROW WHEN (NEWTUPLE.Fname LIKE 'G%') SET NEWTUPLE.BucketID = 1; CREATE TRIGGER USER_AFTER_TRIGGER AFTER INSERT ON USER_BUCKET REFERENCING NEW ROW AS NEWTUPLE FOR EACH ROW UPDATE BUCKET_COUNT SET BucketCount = BucketCount + 1 WHERE BucketID = NEWTUPLE.BucketID; CREATE TRIGGER USER_TABLE_TRIGGER AFTER UPDATE OF Lname ON USER_BUCKET REFERENCING OLD TABLE AS OLDTHING NEW TABLE AS NEWTHING FOR EACH STATEMENT UPDATE BUCKET_COUNT SET BucketCount = (SELECT COUNT(*) FROM OLDTHING);</pre>

이 데이터베이스에서 트리거의 조건 평가(condition evaluation)와 조치의 실행(action execution)은 데이터베이스의 상태를 변경시키는 트랜잭션과 같은 트랜잭션 내에서 상태변경 연산의 발생 즉시 실행되는 즉시 모드(immediate mode)로 작동한다고 가정할 때, 다음 물음에 답하시오. (총 20점)

1) 아래의 INSERT문을 수행한 후의 데이터베이스 상태를 보이시오. (5점)

<pre>INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Gildong', 'Hong');</pre>

2) 아래의 INSERT문을 수행한 후의 데이터베이스 상태를 보이시오. (5점)

<pre>INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Soonmi', 'Kim');</pre>

3) 아래의 SQL문들을 수행한 후의 데이터베이스 상태를 보이시오. (5점)

<pre>BEGIN TRANSACTION INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Bora', 'Lee'); INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Gyungho', 'Park'); ROLLBACK;</pre>
--

4) 아래의 SQL문들을 수행한 후의 데이터베이스 상태를 보이시오. (5점)

<pre>INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Hyewon', 'Hwang'); INSERT INTO USER_BUCKET(Fname, Lname) VALUES('Gyungsoo', 'Oh'); UPDATE USER_BUCKET SET Lname='Yang' WHERE Fname='Gyungsoo' AND Lname='Oh';</pre>
--

인사혁신처 시험출제과장