

## 핵심정리

### 성능데이터모델링이란?

데이터베이스 성능  
향상을 목적으로  
설계단계의 데이터  
모델링 때부터 성능과  
관련된 사항이 데이터  
모델링에 반영될 수  
있도록 하는 것이다.

31

다음 중 성능 데이터모델링에 대한 설명으로 가장 부적절한 것은?

- ☒ ㉠ 성능이 저하된 결과를 대상으로 데이터모델 보다는 문제발생 시점의 SQL을 중심으로 집중하여 튜닝을 한다.
- ☐ ㉡ 데이터의 증가가 빠를수록 성능저하에 따른 성능개선비용은 증가한다.
- ☐ ㉢ 데이터모델은 성능을 튜닝하면서 변경이 될 수 있는 특징이 있다.
- ☐ ㉣ 분석/설계 단계에서 성능을 고려한 데이터모델링을 수행할 경우 성능저하에 따른 Rework비용을 최소화 할 수 있는 기회를 가지게 된다.

32

아래 설명을 읽고 다음 ㉠에 들어갈 단어를 작성하시오.

아 래

첫번째, 데이터모델링을 할 때 정규화를 정확하게 수행한다.  
두번째, 데이터베이스 용량산정을 수행한다.  
세번째, 데이터베이스에 발생하는 트랜잭션의 유형을 파악한다.  
네번째, 용량과 트랜잭션의 유형에 따라 반정규화를 수행한다.  
다섯번째, 이력모델의 조정, PK/FK조정, 슈퍼타입/서브타입 조정 등을 수행한다.

33

다음 중 아래에서 성능을 고려한 데이터 모델링의 순서로 가장 적절한 것은?

아 래

가. 데이터 모델링을 할 때 정규화를 정확하게 수행한다.  
나. 용량과 트랜잭션의 유형에 따라 반정규화를 수행한다.  
다. 데이터베이스 용량산정을 수행한다.  
라. 데이터베이스에 발생하는 트랜잭션의 유형을 파악한다.  
마. 성능관점에서 데이터 모델을 검증한다.  
바. 이력모델의 조정, PK/FK조정, 슈퍼타입/서브타입 조정 등을 수행한다.

- ☐ ㉠ 가-나-다-라-마-바
- ☐ ㉡ 가-나-다-라-마-마
- ☐ ㉢ 가-다-라-바-나-마
- ☒ ㉣ 가-다-라-나-바-마

34

다음 중 성능데이터 모델링을 할 때 고려사항으로 가장 부적절한 것은?

- ① 데이터 모델링의 정규화는 항상 조회 성능저하를 나타내므로 반정규화 관점에서만 성능을 고려하여 설계하도록 한다.
- ② 용량산정은 전체적인 데이터베이스에 발생하는 트랜잭션의 유형과 양을 분석하는 자료가 되므로 성능데이터 모델링을 할 때 중요한 작업이 될 수 있다.
- ③ 물리적인 데이터 모델링을 할 때 PK/FK의 칼럼의 순서조정, FK인덱스 생성 등은 성능 향상을 위한 데이터 모델링 작업에 중요한 요소가 된다.
- ④ 이력데이터는 시간에 따라 반복적으로 발생이 되기 때문에 대량 데이터일 가능성이 높아 특별히 성능을 고려하여 칼럼 등을 추가하도록 설계해야 한다.

35

아래와 같은 보관금원장 엔터티에서 관서에 대한 정보가 반정규화 되어 있기 때문에 관서정보를 조회할 때 성능저하가 발생하고 있다. 이 엔터티에 대해 몇 차 정규화가 필요한 지와 분리된 스키마 구조를 가장 바르게 짚지는 것은?

아 래

[보관금원장]

관서번호  
납부자번호

관리점번호  
관서명  
상태  
관서등록일자  
직급명  
통신번호

함수종속성(FD) :

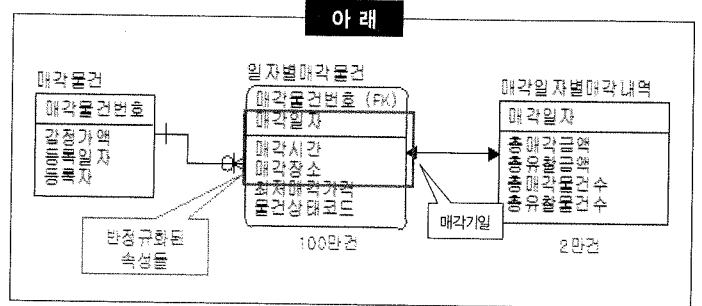
{관서번호, 납부자번호} → {직급명, 통신번호}

{관서번호} → {관리점번호, 관서명, 상태, 관서등록일자}

- ① 2차 정규화 - 정규화테이블(관서번호, 납부자번호, 관리점번호, 관서명, 상태, 관서등록일자)
- ② 3차 정규화 - 정규화테이블(관서번호, 납부자번호, 관리점번호, 관서명, 상태, 관서등록일자)
- ③ 2차 정규화 - 정규화테이블(관서번호, 관리점번호, 관서명, 상태, 관서등록일자)
- ④ 3차 정규화 - 정규화테이블(관서번호, 관리점번호, 관서명, 상태, 관서등록일자)

36

다음 중 아래 '일자별매각물건' 엔터티에 대한 설명으로 가장 적절한 것은?



- ① 1차 정규화가 필요한 엔터티로서 매각기일과 일자별매각물건으로 1:M 관계가 될 수 있다.
- ② 1차 정규화가 필요한 엔터티로서 매각기일과 일자별매각물건으로 1:1 관계가 될 수 있다.
- ③ 2차 정규화가 필요한 엔터티로서 매각기일과 일자별매각물건으로 1:M 관계가 될 수 있다.
- ④ 2차 정규화가 필요한 엔터티로서 매각기일과 일자별매각물건으로 1:1 관계가 될 수 있다.

37

아래의 데이터 모델처럼 동일한 유형의 속성이 칼럼단위로 반복되는 경우가 실제 프로젝트를 하면서 많이 발생 될 수 있다. 다음 중 아래와 같이 전제조건이 있을 때 테이블에서 나타날 수 있는 현상으로 가장 적절한 것은?

아래

전제조건 : 유형기능분류코드에 해당하는 속성들은 분포도가 양호하며, SQL Where절에서 각각의 값이 상수값으로 조건 입력될 수 있는 특징을 가진다.

모델	모델코드
모델	모델코드
제품	제품코드
품종	품종코드
종류	종류코드
A	유형기능분류코드1
B	유형기능분류코드2
C	유형기능분류코드3
D	유형기능분류코드4
E	유형기능분류코드5
F	유형기능분류코드6
G	유형기능분류코드7
H	유형기능분류코드8
I	유형기능분류코드9
모델	모델코드
제품	제품코드
품종	품종코드
종류	종류코드
...	...

- ① 조회 조건이 유형기능분류코드에 따라 반복되는 그룹이 칼럼단위로 되어 있으므로 제 1정규형이라고 할 수 있다.
- ② 유형기능분류코드에 대해 Where절에 조건으로 들어오는 값이 있으므로 PK와 이에 대한 Index만 있으면 SQL문장은 빠르게 수행될 수 있다고 할 수 있다.
- ③ 유형기능분류코드가 일반속성 안에서 반복적으로 속성이 구분되어 있기 때문에 이전종속을 수행해야 하는 제 2정규형이라 할 수 있다.
- ④ 유형기능분류코드 각각에 대하여 개별로 Index를 모두 생성할 경우 입력, 수정, 삭제 때 성능이 저하되므로 제1차 정규화를 수행한 후 인덱스를 적용하는 것이 좋다.

## 핵심정리

38

다음 중 아래 '일재고' 엔터티에 대한 설명으로 가장 적절한 것은?

아 래

일재고	보이스코드
재고일자	재고일자
초재고수량	초재고수량
1차재고수량	1차재고수량
2차재고수량	2차재고수량
3차재고수량	3차재고수량
1차재고수량	1차재고수량
2차재고수량	2차재고수량
3차재고수량	3차재고수량
1차재고수량	1차재고수량
2차재고수량	2차재고수량
3차재고수량	3차재고수량
1차재고수량	1차재고수량
2차재고수량	2차재고수량
3차재고수량	3차재고수량
1차재고수량	1차재고수량
2차재고수량	2차재고수량
3차재고수량	3차재고수량

- ☒ ① 1차 정규화가 필요한 엔터티로서 일재고와 일재고상세로 1:M의 관계가 될 수 있다.
- ☐ ② 1차 정규화가 필요한 엔터티로서 일재고와 일재고상세로 1:1의 관계가 될 수 있다.
- ☐ ③ 2차 정규화가 필요한 엔터티로서 일재고와 일재고상세로 1:M의 관계가 될 수 있다.
- ☐ ④ 2차 정규화가 필요한 엔터티로서 일재고와 일재고상세로 1:1의 관계가 될 수 있다.

중복속성에 대한 분리가 1차 정규화의 대상이 되며, 로우단위의 중복도 1차 정규화의 대상이 되지만 칼럼 단위로 중복이 되는 경우도 1차 정규화의 대상이다.

39

다음 중 아래와 같이 수강지도 엔터티를 만들었을 때 이에 해당하는 정규형과 정규화의 대상으로 가장 바르게 짝지어진 것은?

아 래

[수강지도]

학번  
과목코드  
성적  
지도교수명  
학과명

함수중속성(FD)

- 1. 학번 || 과목번호 → 성적
- 2. 학번 → 지도교수명
- 3. 학번 → 학과명

- ☒ ① 1차 정규형 - 2차 정규화 대상
- ☐ ② 2차 정규형 - 3차 정규화 대상
- ☐ ③ 3차 정규형 - 보이스코드 정규화대상
- ☐ ④ 보이스코드정규형 - 4차 정규화 대상

## 핵심정리

반정규화는 정규화된 엔터티, 속성, 관계에 대해 시스템의 성능향상과 개발(Development)과 운영(Maintenance)의 단순화를 위해 중복, 통합, 분리 등을 수행하는 데이터 모델링의 기법을 의미한다. 반정규화는 데이터를 중복하여 성능을 향상시키기 위한 기법이라고 정의할 수 있고 좀 더 넓은 의미의 반정규화는 성능을 향상시키기 위해 정규화된 데이터 모델에서 중복, 통합, 분리 등을 수행하는 모든 과정을 의미한다. 데이터 무결성이 깨질 수 있는 위험을 무릅쓰고 데이터를 중복하여 반정규화를 적용하는 이유는 데이터를 조회할 때 디스크 I/O량이 많아서 성능이 저하되거나 경로가 너무 멀어 조인으로 인한 성능저하가 예상되거나 칼럼을 계산하여 읽을 때 성능이 저하될 것이 예상되는 경우 반정규화를 수행하게 된다.

40

다음 중 데이터 모델에 대한 반정규화를 고려할 때 판단요소에 대한 설명으로 가장 적절한 것은?

- ☒ ㉠ 반정규화 정보에 대한 재현의 적시성으로 판단한다. 예를 들어, 빌링의 잔액(balance)은 다수 테이블에 대한 다량의 조인이 불가피하므로 데이터 제공의 적시성 확보를 위한 필수 반정규화 대상 정보이다.
- ☒ ㉡ 탐색 대상 데이터의 크기로 판단한다. 왜냐하면 다량 데이터에 대한 인덱스를 활용한 탐색은 Random처리의 특성으로 성능 저하가 불가피하다.
- ☒ ㉢ RDBMS는 현재 레코드 기준으로 이전 또는 이후 위치의 레코드에 대한 접근이 원천적으로 불가능하므로 반정규화를 하지 않으면 해당 정보에 대한 데이터 접근 자체가 불가능하다.
- ☒ ㉣ 반정규화 테이블은 집계 테이블에 국한하여 적용하도록 한다.

41

다음 중 하나의 테이블의 전체 칼럼 중 자주 이용하는 집중화된 칼럼들이 있을 때 디스크 I/O를 줄이기 위해 해당 칼럼들을 별도로 모아놓는 반정규화 기법으로 가장 적절한 것은?

- ☒ ㉠ 칼럼추가 - 부분칼럼 추가
- ☒ ㉡ 칼럼추가 - 중복칼럼 추가
- ☒ ㉢ 테이블추가 - 중복테이블추가
- ☒ ㉣ 테이블추가 - 부분테이블추가

## 테이블의 반정규화

기법분류	반정규화 기법
테이블병합	1:1 관계 테이블병합
	1:M 관계 테이블병합
	슈퍼/서브타입 테이블병합
테이블분할	수직분할
	수평분할
테이블추가	중복테이블 추가
	통계테이블 추가
	이력테이블 추가
	부분테이블 추가

## 칼럼의 반정규화

반정규화 기법
중복칼럼 추가
파생칼럼 추가
이력테이블 칼럼추가
PK에 의한 칼럼 추가
응용시스템 오작동을 위한 칼럼 추가

## 핵심정리

### 반정규화 절차

#### 1. 반정규화 대상조사

- 범위처리빈도수 조사
- 대량의 범위 처리 조사
- 통계성 프로세스 조사
- 테이블 조인 개수

#### 2. 다른 방법유도 검토

- 뷰(VIEW) 테이블
- 클러스터링 적용
- 인덱스의 조정
- 응용애플리케이션

#### 3. 반정규화 적용

- 테이블 반정규화
- 속성의 반정규화
- 관계의 반정규화

42

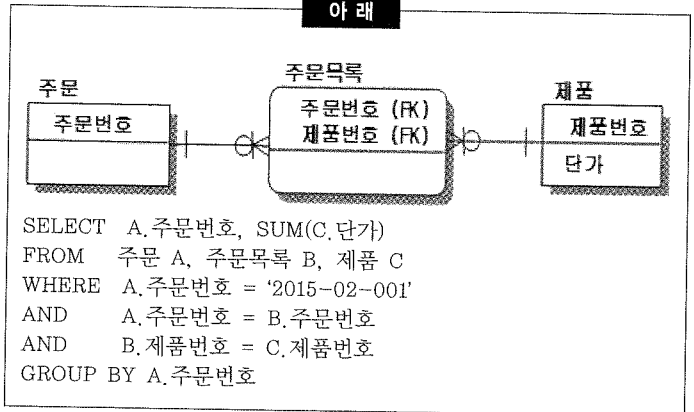
다음 중 칼럼에 대한 반정규화 기법으로 가장 부적절한 것은?

- ① 중복칼럼을 추가 - 조인감소를 위해 여러 테이블에 동일한 칼럼을 갖도록 한다.
- ② 파생칼럼을 추가한다 - 조회 성능을 우수하게 하기 위해 미리 계산된 칼럼을 갖도록 한다.
- ③ FK에 대한 속성을 추가한다 - FK관계에 해당하는 속성을 추가하여 조인성능을 높인다.
- ④ 이력테이블에 기능 칼럼을 추가한다. - 최신값을 처리하는 이력의 특성을 고려하여 기능성 칼럼을 추가한다.

43

다음 중 아래의 주문, 주문목록, 제품에 대한 데이터모델과 이를 이용하여 데이터를 조회하는 SQL문에서 조회를 빠르게 수행하기 위한 반정규화 방법으로 가장 적절한 것은?

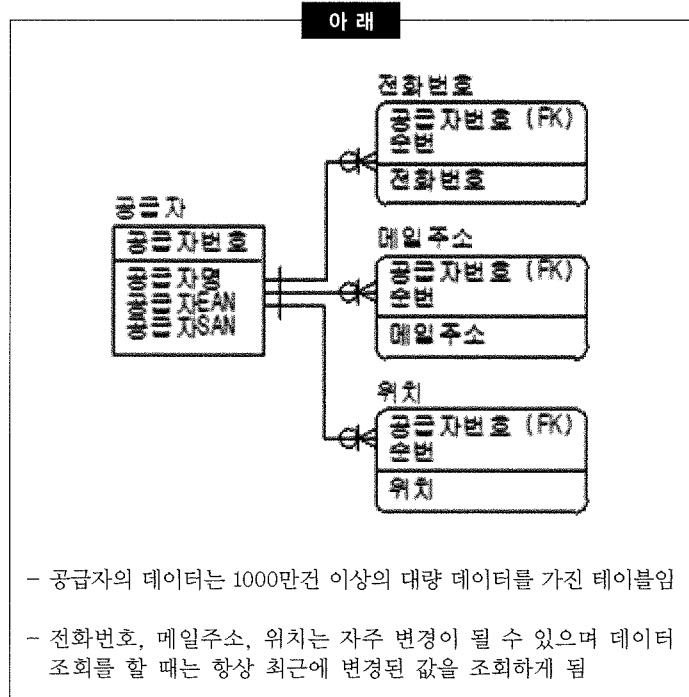
아 래



- ① 제품 엔터티에 단가를 합한 계산된 칼럼을 추가하도록 한다.
- ② 주문목록 엔터티에 단가를 합한 계산된 칼럼을 추가하도록 한다.
- ③ 주문 엔터티에 단가를 합한 계산된 칼럼을 추가하도록 한다.
- ④ 제품 엔터티에 최근값 여부에 대한 칼럼을 추가하도록 한다.

44

다음 중 아래 데이터모델에 대한 설명으로 가장 부적절한 것은?



- ① 공급자별로 최근에 변경된 전화번호, 메일주소, 위치와 공급자 이름을 같이 조회할 때 이 값들을 공급자 테이블에 반정규화로 갖고 있는 경우에 비해 조회 성능이 저하되지 않는다.
- ② 데이터를 조회할 때 과도한 조인으로 인해 조회성능이 저하될 수 있으므로 공급자 테이블에 가장 빈번하게 조회되는 값인 최근 변경값에 해당하는 전화번호, 메일주소, 위치를 반정규화하여 조회 성능을 향상시킬 수 있다.
- ③ 전화번호, 메일주소, 위치에 대한 가장 최근에 변경된 값을 알 수 있도록 최신여부 라는 속성을 추가함으로써 최근 값을 찾기 위한 조회 성능 저하를 예방할 수 있다. **과도한 조인**
- ④ 조회 성능을 위해서는 하나의 테이블로 통합하여 전화번호, 메일주소, 위치 등이 변경될 경우 전체 속성이 계속 발생되는 이력의 형태로 설계될 수 있다. 이럴 경우 조회에 대한 성능은 향상이 되나, 과도한 데이터가 한 테이블에 발생하게 되어 용량이 너무 커지는 단점이 있다.



## 핵심정리

### 반정규화의 대상에 대해 다른 방법으로 처리

- 지나치게 많은  
조인(JOIN)이 걸려  
데이터를 조회하는  
작업이 기술적으로  
어려울 경우 뷰  
(VIEW)를 사용하면  
이를 해결할 수도 있다.
- 대량의 데이터처리나  
부분처리에 의해  
성능이 저하되는  
경우에 클러스터링을  
적용하거나 인덱스를  
조정함으로써 성능을  
향상시킬 수 있다.
- 대량의 데이터는  
Primary Key의 성격에  
따라 부분적인  
테이블로 분리할 수  
있다. 즉 파티셔닝 기법  
(Partitioning)이  
적용되어 성능저하를  
방지할 수 있다.
- 응용 애플리케이션에서  
로직을 구사하는  
방법을 변경함으로써  
성능을 향상시킬 수  
있다.

45

다음 중 칼럼수가 많은 테이블에 대한 설명으로 가장 적절한 것은?

- ☒ 한 테이블에 많은 칼럼을 가지고 있으면 조인이 발생되지 않아 여러  
개 테이블일 때에 비해 성능이 항상 우수하다고 할 수 있다.
- ☒ 로우চে이닝이 발생할 정도로 한 테이블에 많은 칼럼들이 존재할 경우  
조회성능저하가 발생할 수 있다. 한 테이블내에서 칼럼의 위치를 조정  
하면 디스크 I/O가 줄어들어 조회 성능을 향상 시킬 수 있다.
- ☒ 로우চে이닝이 발생할 정도로 한 테이블에 많은 칼럼들이 존재할 경우  
조회성능저하가 발생할 수 있다. 트랜잭션이 접근하는 칼럼유형을 분  
석하여 1:1로 테이블을 분리하면 디스크 I/O가 줄어들어 조회 성능을  
향상 시킬 수 있다.
- ☒ 로우চে이닝이 발생할 정도로 한 테이블에 많은 칼럼들이 존재할 경우  
조회성능저하가 발생할 수 있다. 그러나 이를 분리할 경우 조인으로  
인한 성능 저하가 더 심하게 나타날 수 있으므로 감수하는 것이 좋다.

46

아래 설명에서 데이터 액세스 성능을 향상시키기 위해 적용하는 방법에 대해  
서 ㉠ 을 채우시오.

파티셔닝

아 래

하나의 테이블에 많은 양의 데이터가 저장되면 인덱스를 추가하고  
테이블을 몇 개로 쪼개도 성능이 저하되는 경우가 있다. 이때 논리적  
으로는 하나의 테이블이지만 물리적으로는 여러 개의 테이블로 분리  
하여 데이터 액세스 성능도 향상시키고, 데이터 관리방법도 개선할  
수 있도록 테이블에 적용하는 기법을 ㉠ 이라고 한다.

## 핵심정리

슈퍼/서브 타입 데이터  
모델의 변환기술

- 개별로 발생하는 트랜잭션에 대해서는 개별 테이블로 구성
- 슈퍼타입+서브타입에 대해 발생하는 트랜잭션에 대해서는 슈퍼타입+서브타입 테이블로 구성
- 전체를 하나로 묶어 트랜잭션이 발생할 때는 하나의 테이블로 구성

47

다음 중 아래 데이터 모델과 SQL문에 대해 개선해야 할 사항에 대한 설명으로 가장 적절한 것은?

아래

[긴급사건]

긴급사건 번호

사건명  
발생일시

[특수사건]

특수사건 번호

사건명  
발생일시

[일반사건]

일반사건 번호

사건명  
발생일시

SELECT 긴급사건번호, 사건명 FROM 긴급사건 WHERE 발생일시 = '20150905'  
UNION ALL

SELECT 특수사건번호, 사건명 FROM 특수사건 WHERE 발생일시 = '20150905'  
UNION ALL

SELECT 일반사건번호, 사건명 FROM 일반사건 WHERE 발생일시 = '20150905'

※ 전제조건 : 위 세 테이블은 함께 조회하는 경우가 대부분이고 아직 시스템을 오픈하지 않았다.

- ① UNION ALL로 조회하면 정렬로 인한 성능이 저하되므로 UNION으로 조합하여 조회한다.
- ② 긴급사건, 특수사건, 일반사건을 하나의 테이블로 통합하고 PK를 사건 분류코드 + 사건번호로 조합하여 구성하도록 한다.
- ③ 긴급사건, 특수사건, 일반사건 테이블을 개별로 유지하되 PK에 사건 분류코드를 포함하도록 한다.
- ④ 세 개의 테이블을 그대로 유지하되 반정규화된 형태의 통합테이블을 하나 더 생성하여 조회의 성능을 향상하도록 한다.

48

다음 중 논리데이터모델의 슈퍼타입과 서브타입 데이터모델을 물리적인 테이블 형식으로 변환할 때 설명으로 가장 부적절한 것은?

- ① 트랜잭션은 항상 전체를 대상으로 일괄 처리하는데 테이블은 서브타입 별로 개별 유지하는 것으로 변환하면 Union 연산에 의해 성능이 저하될 수 있다.
- ② 트랜잭션은 항상 서브타입 개별로 처리하는데 테이블은 하나로 통합하여 변환하면 불필요하게 많은 양의 데이터가 집적되어 있어 성능이 저하될 수 있다.
- ③ 트랜잭션은 항상 슈퍼+서브 타입을 함께 처리하는데 개별로 유지하면 조인에 의해 성능이 저하될 수 있다.
- ④ 트랜잭션은 항상 전체를 통합하여 분석 처리하는데 하나로 통합되어 있으면 데이터 집적으로 인해 성능이 저하될 수 있다.

다음 중 아래와 같은 '현금출금기실적' 테이블과 이 테이블에서 데이터를 조회할 때 사용되는 아래의 SQL패턴에 대한 설명으로 가장 적절한 것은?

## 아 래

## 현금출금기실적

거래일자
사무소코드
출금기번호
명세표번호
건수
금액

```
SELECT 건수, 금액
FROM 현금출금기실적
WHERE 거래일자 BETWEEN
'20140701' AND '20140702'
AND 사무소코드 = '000368'
```

(단, PK인덱스는 설계된 순서 그대로 생성함)

- ㉠ 사무소코드가 '='로 상수값이 들어 왔고, 거래일자가 범위 'BETWEEN'으로 들어왔기 때문에 거래일자+사무소코드 + 출금기번호 + 명세표번호로 구성된 PK인덱스는 최적의 효율성을 가지고 있음.
- ㉡ 사무소코드가 '='로 상수값이 들어 왔고, 거래일자가 범위 'BETWEEN'으로 들어왔기 때문에 PK의 순서를 사무소코드 + 출금기번호 + 명세표번호 + 거래일자로 바꾸고 인덱스를 생성하는 것이 성능에 유리함.
- ㉢ 사무소코드가 '='로 상수값이 들어 왔고, 거래일자가 범위 'BETWEEN'으로 들어왔기 때문에 PK의 순서를 거래일자 + 출금기번호 + 명세표번호 + 사무소코드로 바꾸고 인덱스를 생성하는 것이 성능에 유리함.
- ㉣ 사무소코드가 '='로 상수값이 들어 왔고, 거래일자가 범위 'BETWEEN'으로 들어왔기 때문에 PK의 순서를 사무소코드 + 거래일자 + 출금기번호 + 명세표번호로 바꾸고 인덱스를 생성하는 것이 성능에 유리함.

## 핵심정리

PK순서를 결정하는 기준은 인덱스 정렬구조를 이해한 상태에서 인덱스를 효율적으로 이용할 수 있도록 PK순서를 지정해야 한다. 즉 인덱스의 특징은 여러 개의 속성이 하나의 인덱스로 구성되어 있을 때 앞쪽에 위치한 속성의 값이 비교자로 있어야 인덱스가 좋은 효율을 나타낼 수 있다. 앞쪽에 위치한 속성 값이 가급적 '=' 아니면 최소한 범위 'BETWEEN' < >가 들어와야 인덱스를 이용할 수 있는 것이다.

50

다음 중 아래의 엔터티와 이 엔터티를 매우 빈번하게 참조하는 SQL의 성능에 대한 설명으로 가장 적절한 것은?

아 래

실적

일자	지사코드	명세번호
금액	건수	

```
SELECT 건수, 금액
FROM 실적
WHERE 일자 BETWEEN '20110101' AND '20110102'
AND 지사코드 = '1001'
```

(단, PK 인덱스가 존재하며 칼럼순서대로 인덱스가 생성되어 있고, 이 SQL이 트랜잭션의 대부분을 차지한다고 가정함)

- ① 명세번호를 실적 테이블에 맨 처음 칼럼으로 옮기면 PK 인덱스의 이용 효율성이 향상된다.
- ② 지사코드에 대해 입력 값이 EQUAL 조건으로 사용되므로 지사코드를 실적 테이블의 맨 처음 위치로 옮겨서 일자 칼럼의 앞에 두면 인덱스 이용 효율성이 높아진다.
- ③ 일자가 SQL문장의 Where절에 첫 번째 조건으로 나왔으므로 칼럼 순서에서도 첫번째에 위치하는 것이 효율성이 높다.
- ④ 일자, 명세번호, 지사코드로 칼럼 순서를 바꾸어야 일자 범위에 대한 내용을 먼저 식별하고, 그 다음으로 지사코드를 찾기에 용이할 수 있으므로 이때 인덱스의 효율성이 가장 높다.

## 핵심정리

### 분산 데이터베이스 장단점

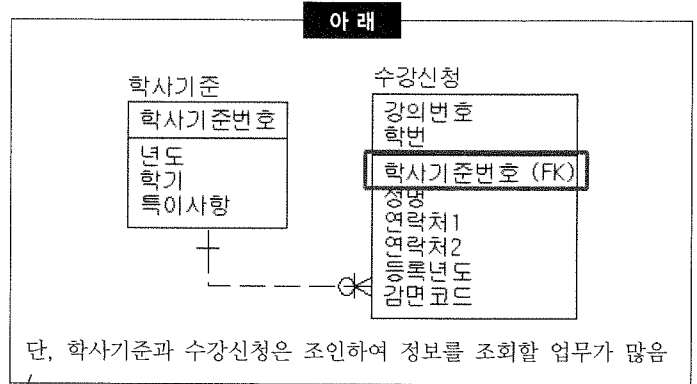
- 지역 자치성, 점증적 시스템 용량 확장
- 신뢰성과 가용성
- 효율성과 융통성
- 빠른 응답 속도와 통신비용 절감
- 데이터의 가용성과 신뢰성 증가
- 시스템 규모의 적절한 조절
- 각 지역 사용자의 요구 수용 증대

### 단점

- 소프트웨어 개발 비용
- 오류의 잠재성 증대
- 처리 비용의 증대
- 설계, 관리의 복잡성과 비용
- 불규칙한 응답 속도
- 통제의 어려움
- 데이터 무결성에 대한 위협

51

다음 중 아래 데이터모델에 표현된 FK(Foreign Key)에 대한 설명으로 가장 적절한 것을 2개 고르시오.



- ① 학사기준번호는 부모 테이블에 이미 인덱스가 존재하기 때문에 상속받아 생긴 수강신청에는 학사기준번호 칼럼에 대한 별도의 인덱스가 필요하지 않다.
- ② 학사기준번호는 부모 테이블에 이미 인덱스가 존재하나 수강신청과 조인에 의한 성능저하 예방을 위해 상속받아 생긴 수강신청에도 학사기준번호 칼럼에 대한 별도의 인덱스가 필요하다.
- ③ 데이터모델에서는 관계를 연결하고 데이터베이스에 FK제약조건 생성을 생략하는 경우에 학사기준번호에 대한 인덱스를 생성할 필요가 없다.
- ④ 데이터모델에서는 관계를 연결하고 데이터베이스에 FK제약조건 생성을 생략하는 경우에도 데이터의 조인관계가 필요하므로 학사기준번호에 대한 인덱스를 생성할 필요가 있다.

52

다음 중 데이터가 여러 지역에 분산되어 있지만 하나의 데이터베이스처럼 사용하기를 원하는 분산데이터베이스 환경에서 데이터베이스 분산설계를 적용하여 효율성을 증대시킬 수 없는 것은?

- ① 공통코드, 기준정보 등 마스터 데이터는 분산데이터베이스에 복제분산을 적용한다.
- ② 거의 실시간(Near Real Time) 업무적인 특성을 가지고 있을 때 분산 데이터베이스를 사용하여 구성할 수 있다.
- ③ 백업 사이트를 구성할 때 간단하게 분산기능을 적용하여 구성할 수 있다.
- ④ Global Single Instance(GSI)를 구성할 때 분산데이터베이스를 활용하여 구성하는 것이 효율적이다.