🥷

[취업폭격기 Zeromini 위클리 개념 폭격 #24]

📕 과목 : 데이터베이스론

🔥 참고문제 : 2021년 공무원 7급

🌝 문제 수정 버전 : V 1.0

1. SQL 언어
    - 문제: SQL 언어의 주요 특징과 관계형 데이터베이스 시스템에서의 역할에 대해 설명하세요.
    - 해설: SQL은 관계형 데이터베이스 관리 시스템(RDBMS)의 표준 언어로, 데이터 정의, 질의, 갱신 등을 위한 다양한 문법을 제공합니다. 데이터 조작어(DML), 데이터 정의어(DDL), 데이터 제어어(DCL) 등으로 구성되며, 데이터베이스의 구조를 정의하고, 데이터를 조작하며, 데이터베이스의 보안과 무결성을 관리하는 데 사용됩니다.

2. 트랜잭션
    - 문제: 데이터베이스에서 트랜잭션의 중요성과 ACID 속성에 대해 설명하세요.
    - 해설: 트랜잭션은 데이터베이스에서 하나의 논리적 작업 단위로, 여러 작업들이 모두 성공하거나 실패해야 하는 일련의 연산들을 의미합니다. ACID 속성은 원자성(Atomicity), 일관성(Consistency), 격리성(Isolation), 지속성(Durability)을 나타내며, 데이터베이스의 신뢰성과 정확성을 보장하는 데 필수적입니다.
3. 뷰(View)
    - 문제: 데이터베이스에서 뷰의 개념과 사용 목적에 대해 설명하세요.
    - 해설: 뷰는 데이터베이스의 하나 이상의 테이블에서 파생된 가상 테이블로, 실제 데이터를 저장하지 않고 특정 데이터 집합에 대한 논리적인 창을 제공합니다. 사용자에게 필요한 데이터만을 보여주거나, 복잡한 쿼리를 단순화하는 데 사용되며, 데이터 보안과 관리의 편의성을 높이는 데 기여합니다.
4. 관계 모델
    - 문제: 관계 모델의 기본 구조와 특징에 대해 설명하세요.
    - 해설: 관계 모델은 데이터를 테이블 형태로 표현하는 데이터 모델로, 테이블은 행(튜플)과 열(속성)로 구성됩니다. 각 행은 고유한 데이터 레코드를 나타내며, 열은 해당 레코드의 특정 속성을 나타냅니다. 데이터는 원자값으로 구성되어야 하며, 테이블 간의 관계는 키를 통해 정의됩니다.
5. 정규화
    - 문제: 데이터베이스 정규화의 목적과 주요 정규형에 대해 설명하세요.
    - 해설: 정규화는 데이터베이스 설계 과정에서 중복을 최소화하고 데이터 무결성을 유지하기 위해 데이터를 구조화하는 과정입니다. 제1정규형부터 시작하여, 각 단계는 특정 조건을 만족시키며, 데이터의 중복을 줄이고, 삽입, 삭제, 수정 시 발생할 수 있는 이상 현상을 방지합니다.
6. 인덱스(Index)
    - 문제: 데이터베이스에서 인덱스의 역할과 효과에 대해 설명하세요.
    - 해설: 인덱스는 데이터베이스에서 데이터 검색 속도를 향상시키기 위해 사용되는 데이터 구조입니다. 테이블의 특정 열에 대해 생성되며, 데이터의 빠른 검색을 위해 키-값 쌍으로 구성됩니다. 인덱스는 검색 성능을 향상시키지만, 삽입, 삭제, 갱신 작업 시 추가적인 비용이 발생할 수 있습니다.
7. 조인(Join)
    - 문제: SQL에서 사용되는 다양한 조인 유형과 그 특징에 대해 설명하세요.

- 해설: SQL에서 조인은 두 개 이상의 테이블을 연결하여 데이터를 검색하는 방법입니다. 내부 조인(Inner Join), 외부 조인(Left, Right, Full Outer Join), 교차 조인(Cross Join), 자연 조인(Natural Join) 등이 있으며, 각각의 조인 유형은 테이블 간의 관계와 필요한 데이터에 따라 선택됩니다.

8. 데이터베이스 쿼리
    - 문제: '학생' 테이블에서 '학과'가 '컴퓨터공학'인 학생들의 이름을 검색하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 '학생' 테이블에서 특정 조건을 만족하는 데이터를 검색하는 기본적인 SQL 쿼리입니다. 예를 들어, `SELECT 이름 FROM 학생 WHERE 학과 = '컴퓨터공학';`과 같은 쿼리를 사용하여 '컴퓨터공학' 학과의 학생들의 이름을 검색할 수 있습니다.
9. 데이터베이스 보안
    - 문제: 데이터베이스 보안의 중요성과 기본적인 보안 조치에 대해 설명하세요.
    - 해설: 데이터베이스 보안은 무단 접근, 데이터 유출, 손상으로부터 데이터를 보호하는 것을 목적으로 합니다. 이를 위해 사용자 인증, 접근 제어, 암호화, 백업 및 복구 전략 등 다양한 보안 조치가 필요합니다. 이러한 조치는 데이터의 무결성과 기밀성을 유지하는 데 중요합니다.
10. 데이터 모델링
    - 문제: 데이터 모델링의 목적과 과정에 대해 설명하세요.
    - 해설: 데이터 모델링은 현실 세계의 데이터를 데이터베이스 시스템 내에서 효과적으로 표현하기 위한 과정입니다. 이 과정은 요구 사항 분석, 개념적 모델링, 논리적 모델링, 물리적 모델링 단계를 포함하며, 데이터의 구조, 관계, 제약 조건을 정의하는 데 중점을 둡니다.
11. 데이터베이스 쿼리
    - 문제: '도서' 테이블과 '저자' 테이블을 조인하여, 모든 도서의 제목과 해당 저자의 이름을 검색하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 두 테이블을 조인하여 관련 데이터를 검색하는 예입니다. 예를 들어, `SELECT 도서.제목, 저자.이름 FROM 도서 INNER JOIN 저자 ON 도서.저자ID = 저자.ID;` 쿼리를 사용하여 각 도서의 제목과 해당하는 저자의 이름을 검색할 수 있습니다.
12. 데이터베이스 쿼리 최적화
    - 문제: 데이터베이스 쿼리 최적화의 중요성과 기본적인 최적화 기법에 대해 설명하세요.

- 해설: 쿼리 최적화는 데이터베이스의 성능을 향상시키고, 자원 사용을 최소화하는 데 중요합니다. 인덱스 활용, 쿼리 재작성, 실행 계획 분석, 적절한 데이터 타입 사용 등이 기본적인 최적화 기법입니다. 이러한 기법들은 쿼리의 실행 시간을 단축하고, 시스템의 부하를 줄이는 데 도움이 됩니다.

13. 데이터베이스 무결성
    - 문제: 데이터베이스의 무결성이란 무엇이며, 어떻게 유지할 수 있는지 설명하세요.
    - 해설: 데이터베이스 무결성은 데이터의 정확성, 일관성, 신뢰성을 의미합니다. 이를 유지하기 위해 제약 조건(기본 키, 외래 키, 유니크, 체크 제약 조건 등), 트랜잭션 관리, 백업 및 복구 전략 등이 사용됩니다. 이러한 조치들은 데이터의 오류와 손상을 방지하고, 데이터의 품질을 유지하는 데 필수적입니다.
14. 데이터 웨어하우스
    - 문제: 데이터 웨어하우스의 개념과 기업에서의 활용 방법에 대해 설명하세요.
    - 해설: 데이터 웨어하우스는 다양한 소스에서 수집된 대량의 데이터를 통합, 저장, 관리하는 시스템입니다. 기업에서는 의사결정 지원, 비즈니스 인텔리전스, 데이터 마이닝 등에 활용됩니다. 데이터 웨어하우스는 데이터의 일관성을 유지하고, 복잡한 쿼리와 분석을 위한 효율적인 환경을 제공합니다.
15. NoSQL 데이터베이스
    - 문제: NoSQL 데이터베이스의 특징과 관계형 데이터베이스와의 주요 차이점에 대해 설명하세요.
    - 해설: NoSQL 데이터베이스는 스키마가 없거나 유연한 스키마를 가지며, 대규모 분산 데이터 처리에 적합합니다. 관계형 데이터베이스와 달리, 엄격한 테이블 구조를 요구하지 않고, 수평적 확장이 용이합니다. 주로 빅 데이터, 실시간 웹 애플리케이션 등에서 사용됩니다.
16. 데이터 마이닝
    - 문제: 데이터 마이닝의 개념과 기업에서의 활용 예를 설명하세요.
    - 해설: 데이터 마이닝은 대규모 데이터 집합에서 유용한 패턴, 관계, 추세를 발견하는 과정입니다. 기업에서는 고객 세분화, 시장 분석, 리스크 관리, 제품 추천 등에 활용됩니다. 데이터 마이닝 기법에는 분류, 군집화, 연관 규칙 학습 등이 있으며, 이를 통해 의사결정 과정을 지원하고 비즈니스 통찰력을 제공합니다.
17. 데이터베이스 쿼리
    - 문제: '주문' 테이블에서 최근 30일 이내에 이루어진 모든 주문의 수를 계산하는 SQL 쿼리를 작성하세요.

- 해설: 이 쿼리는 특정 기간 내의 데이터를 집계하는 방법을 보여줍니다. 예를 들어, `SELECT COUNT(*) FROM 주문 WHERE 주문일자 >= CURDATE() - INTERVAL 30 DAY;` 쿼리를 사용하여 최근 30일 이내의 주문 수를 계산할 수 있습니다.

18. 데이터베이스 샤딩
    - 문제: 데이터베이스 샤딩의 개념과 이점에 대해 설명하세요.
    - 해설: 샤딩은 데이터베이스를 수평으로 분할하여 여러 서버에 분산 저장하는 방법입니다. 이는 데이터베이스의 부하를 분산시키고, 읽기 및 쓰기 성능을 향상시키며, 대규모 데이터 처리에 효과적입니다. 샤딩은 데이터 관리를 최적화하고, 시스템의 확장성을 높이는 데 도움이 됩니다.
19. 데이터베이스 복제
    - 문제: 데이터베이스 복제의 목적과 방법에 대해 설명하세요.
    - 해설: 데이터베이스 복제는 데이터를 다른 위치에 복사하여 데이터의 가용성과 안정성을 높이는 과정입니다. 이는 재해 복구, 데이터 분산, 부하 분산, 백업 등의 목적으로 사용됩니다. 복제 방법에는 마스터-슬레이브 복제, 피어 투 피어 복제 등이 있으며, 각각의 방법은 특정 요구 사항과 환경에 따라 선택됩니다.
20. 데이터베이스 트리거
    - 문제: 데이터베이스 트리거의 개념과 사용 예를 설명하세요.
    - 해설: 트리거는 데이터베이스 테이블에 특정 조건이나 이벤트가 발생했을 때 자동으로 실행되는 프로시저입니다. 데이터 무결성 유지, 자동 갱신, 감사 로그 작성 등에 사용됩니다. 예를 들어, 특정 테이블에 데이터가 삽입되면 관련 테이블을 자동으로 업데이트하는 트리거를 설정할 수 있습니다.
21. 데이터베이스 뷰(View) 업데이트
    - 문제: 데이터베이스 뷰를 업데이트할 수 있는 조건과 뷰를 업데이트하는 과정에 대해 설명하세요.
    - 해설: 뷰는 기본적으로 읽기 전용이지만, 특정 조건 하에서 업데이트가 가능합니다. 업데이트 가능한 뷰는 단일 테이블에 기반해야 하며, 모든 필드가 기본 테이블의 키를 포함해야 합니다. 뷰를 업데이트하는 과정은 실제로 기본 테이블의 데이터를 변경하는 것을 의미합니다.
22. 데이터베이스 백업 전략
    - 문제: 데이터베이스의 백업 전략과 중요성에 대해 설명하세요.
    - 해설: 데이터베이스 백업은 데이터 손실을 방지하고 재해 복구를 위해 필수적입니다. 전략에는 전체 백업, 증분 백업, 차등 백업 등이 있으며, 각각은 데이터의 중요도

와 변경 빈도에 따라 선택됩니다. 정기적인 백업은 데이터의 안전성을 보장하고, 비상 상황에서 데이터 복구를 가능하게 합니다.

23. 데이터베이스 쿼리
    - 문제: '고객' 테이블에서 '도시'별로 고객 수를 계산하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 특정 열에 따른 데이터 집계를 보여줍니다. 예를 들어, `SELECT 도시, COUNT(*) FROM 고객 GROUP BY 도시;` 쿼리를 사용하여 각 도시별 고객 수를 계산할 수 있습니다.
24. 데이터베이스 이상(Anomaly)
    - 문제: 데이터베이스에서 발생할 수 있는 삽입, 삭제, 수정 이상에 대해 설명하세요.
    - 해설: 삽입 이상은 적절한 데이터를 삽입할 수 없는 상황을, 삭제 이상은 한 데이터를 삭제함으로써 관련 데이터가 손실되는 상황을, 수정 이상은 일부 데이터만 변경하여 데이터 불일치가 발생하는 상황을 의미합니다. 이러한 이상은 적절한 데이터베이스 설계와 정규화를 통해 방지할 수 있습니다.
25. 데이터베이스 스키마
    - 문제: 데이터베이스 스키마의 개념과 중요성에 대해 설명하세요.
    - 해설: 데이터베이스 스키마는 데이터베이스의 구조적인 설계를 나타내며, 테이블, 열, 관계, 제약 조건 등을 정의합니다. 스키마는 데이터의 조직화, 무결성, 효율적인 데이터 관리를 위해 중요하며, 데이터베이스 설계의 기본이 됩니다.
26. 데이터베이스 쿼리
    - 문제: '직원' 테이블에서 '부서'별 평균 급여를 계산하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 특정 그룹에 대한 평균 값을 계산하는 방법을 보여줍니다. 예를 들어, `SELECT 부서, AVG(급여) FROM 직원 GROUP BY 부서;` 쿼리를 사용하여 각 부서별 평균 급여를 계산할 수 있습니다.
27. 트랜잭션 관리
    - 문제: 데이터베이스에서 트랜잭션의 개념과 ACID 속성에 대해 설명하세요.
    - 해설: 트랜잭션은 데이터베이스에서 하나의 논리적인 작업 단위로, 여러 쿼리를 포함할 수 있습니다. ACID 속성은 원자성(Atomicity), 일관성(Consistency), 격리성(Isolation), 지속성(Durability)을 의미하며, 트랜잭션이 안전하고 신뢰성 있게 처리되도록 보장합니다.
28. 데이터베이스 정규화
    - 문제: 데이터베이스 정규화의 목적과 주요 정규형에 대해 설명하세요.

- 해설: 정규화는 데이터베이스의 설계를 최적화하여 중복을 줄이고, 무결성을 향상시키는 과정입니다. 주요 정규형에는 제1정규형(1NF), 제2정규형(2NF), 제3정규형(3NF), BCNF 등이 있으며, 각 단계는 데이터의 구조를 점점 더 개선합니다.

29. 데이터베이스 쿼리
    - 문제: '주문' 테이블에서 각 '고객ID'별로 최대 주문 금액을 찾는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 각 그룹별로 최대 값을 찾는 방법을 보여줍니다. 예를 들어, `SELECT 고객ID, MAX(주문금액) FROM 주문 GROUP BY 고객ID;` 쿼리를 사용하여 각 고객별 최대 주문 금액을 찾을 수 있습니다.
30. 데이터베이스 캐싱
    - 문제: 데이터베이스 캐싱의 개념과 이점에 대해 설명하세요.
    - 해설: 데이터베이스 캐싱은 자주 접근하는 데이터를 메모리에 임시 저장하여 빠른 데이터 검색을 가능하게 하는 기술입니다. 이는 데이터베이스의 응답 시간을 단축하고, 시스템의 전반적인 성능을 향상시킵니다.
31. 분산 데이터베이스
    - 문제: 분산 데이터베이스의 개념과 장단점에 대해 설명하세요.
    - 해설: 분산 데이터베이스는 네트워크를 통해 여러 위치에 분산된 데이터베이스 시스템입니다. 이는 데이터의 가용성과 신뢰성을 향상시키지만, 복잡한 관리와 일관성 유지의 어려움이 있습니다.
32. 데이터베이스 쿼리
    - 문제: '직원' 테이블에서 '부서'별로 총 급여 합계를 계산하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 특정 그룹별로 합계를 계산하는 방법을 보여줍니다. 예를 들어, `SELECT 부서, SUM(급여) FROM 직원 GROUP BY 부서;` 쿼리를 사용하여 각 부서별 총 급여 합계를 계산할 수 있습니다.
33. 데이터베이스 인덱싱
    - 문제: 데이터베이스 인덱싱의 개념과 이점에 대해 설명하세요.
    - 해설: 인덱싱은 데이터베이스에서 데이터 검색 속도를 향상시키기 위해 사용하는 기술입니다. 인덱스는 데이터의 빠른 검색을 가능하게 하며, 특히 대규모 데이터베이스에서 성능 향상에 크게 기여합니다. 그러나 인덱스는 추가적인 저장 공간을 필요로 하며, 데이터 삽입이나 수정 시 성능 저하를 일으킬 수 있습니다.
34. 데이터베이스 보안

- 문제: 데이터베이스 보안의 중요성과 기본적인 보안 조치에 대해 설명하세요.
- 해설: 데이터베이스 보안은 무단 접근, 데이터 손실, 손상으로부터 데이터를 보호하는 것을 의미합니다. 기본적인 보안 조치에는 사용자 인증, 역할 기반 접근 제어, 암호화, 백업 및 복구 계획 등이 포함됩니다. 이러한 조치들은 데이터의 무결성과 기밀성을 유지하는 데 필수적입니다.

35. 데이터베이스 쿼리
    - 문제: '고객' 테이블에서 '이름'이 '김%'로 시작하는 모든 고객의 정보를 검색하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 LIKE 연산자를 사용하여 특정 패턴을 만족하는 데이터를 검색하는 방법을 보여줍니다. 예를 들어, `SELECT * FROM 고객 WHERE 이름 LIKE '김%';` 쿼리를 사용하여 '김'으로 시작하는 이름을 가진 모든 고객의 정보를 검색할 수 있습니다.
36. 데이터베이스 트랜잭션 격리 수준
    - 문제: 데이터베이스 트랜잭션의 격리 수준과 각 수준에서 발생할 수 있는 문제점에 대해 설명하세요.
    - 해설: 트랜잭션 격리 수준은 다른 트랜잭션으로부터 데이터를 얼마나 격리시킬지 결정합니다. 격리 수준에는 READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE 등이 있습니다. 낮은 격리 수준은 성능은 높지만, 더티 리드, 논리적 오류 등의 문제를 일으킬 수 있습니다.
37. 데이터베이스 쿼리
    - 문제: '주문' 테이블에서 '주문일자'가 지난 달인 모든 주문을 검색하는 SQL 쿼리를 작성하세요.
    - 해설: 이 쿼리는 날짜 계산을 사용하여 특정 기간의 데이터를 검색하는 방법을 보여줍니다. 예를 들어, `SELECT * FROM 주문 WHERE 주문일자 BETWEEN DATE_FORMAT(NOW() - INTERVAL 1 MONTH, '%Y-%m-01') AND LAST_DAY(NOW() - INTERVAL 1 MONTH);` 쿼리를 사용하여 지난 달의 모든 주문을 검색할 수 있습니다.
38. 데이터 모델링
    - 문제: 데이터 모델링의 목적과 과정에 대해 설명하세요.
    - 해설: 데이터 모델링은 데이터베이스의 구조를 계획하고 설계하는 과정입니다. 이 과정은 요구 사항 분석, 개념적 모델링, 논리적 모델링, 물리적 모델링 단계를 포함합니다. 목적은 데이터의 구조를 명확하게 하고, 효율적인 데이터 관리 및 사용을 지원하는 것입니다.
39. 데이터베이스 쿼리

- 문제: '직원' 테이블에서 '부서'별로 최소, 최대, 평균 급여를 계산하는 SQL 쿼리를 작성하세요.
- 해설: 이 쿼리는 집계 함수를 사용하여 특정 그룹별로 다양한 통계 값을 계산하는 방법을 보여줍니다. 예를 들어, `SELECT 부서, MIN(급여), MAX(급여), AVG(급여) FROM 직원 GROUP BY 부서;` 쿼리를 사용하여 각 부서별 최소, 최대, 평균 급여를 계산할 수 있습니다.

40. 데이터베이스 성능 튜닝
    - 문제: 데이터베이스 성능 튜닝의 중요성과 기본적인 접근 방법에 대해 설명하세요.
    - 해설: 데이터베이스 성능 튜닝은 시스템의 응답 시간을 개선하고, 자원 사용을 최적화하는 과정입니다. 이를 위해 쿼리 최적화, 인덱싱 전략, 하드웨어 및 서버 설정 조정 등이 필요합니다. 성능 튜닝은 데이터베이스의 효율적인 운영을 위해 필수적입니다.