##### **1. 다음 JOIN의 종류의 설명 중 틀린 것은 무엇인가?**

① EQUI JOIN 은 반드시 PK, FK 관계에 의해서만 성립된다.

② NON-EUQI JOIN은 등가 조건이 성립되지 않은 테이블에 JOIN을 걸어주는 방법이다.

③ OUTER JOIN 은 JOIN 조건을 만족하지 않는 데이터도 볼 수 있는 방법이다.

④ SELF JOIN 은 하나의 테이블을 논리적으로 분리시켜 EQUI JOIIN 을 이용하는 방법이다.

##### **2. 4개의 테이블로부터 필요한 칼럼을 조회하려고 한다. 최소한 몇 개의 JOIN 이 필요한가 ?**

① 2개 ② 3개

③ 4개 ④ 5개

##### **3. 다음 SET 연산자에 대한 설명 중 틀린 것을 무엇인가 ?**

① UNION 연산자는 조회 결과에 대한 합집합을 나타내며 자동으로 정렬을 해준다.

② UNION ALL 연산자는 조회 결과를 정렬하고 중복되는 데이터를 한 번만 표현한다.

③ INTERSETC 연산자는 조회 결과에 대한 교집합을 의미한다.

④ EXCEPT 연산자는 조회 결과에 대한 차집합을 의미한다.

##### **4. 다음 중 SELF JOIN을 수행해야 할 때는 어떤 경우인가 ?**

① 두 테이블에 대한 공통 칼럼이 존재하고 두 테이블이 연관 관계가 있다.

② 두 테이블에 연관된 칼럼은 없으나 JOIN을 해야 한다.

③ 한 테이블 내에서 두 컬럼이 연관 관계가 있다.

④ 한 테이블 내에서 연관된 칼럼은 없으나 JOIN 을 해야 한다.

##### **5. 일반적으로 FROM 절에 정의된 후 먼저 수행되어 SQL 문장 내에서 절차성을 주는 효과를 볼 수 있는**

것은 어떤 유형의 서브쿼리 문장인가 ?

① SCALAR SUBQUERY

② NESTED SUBQUERY

③ CORRELATED SUBQUERY

④ INLINE VIEW

##### **6. 다음 서브쿼리에 대한 설명 중 틀린 것을 고르시오.**

① 다중행 연산자는 IN, ANY, ALL 이 있으며, 서브쿼리의 결과로 하나 이상의 데이터가 RETURN 되는 서브쿼리이다.

② TOP-N 서브쿼리는 INLINE VIEW 의 정렬된 데이터를 ROWNUM 을 이용해 결과 형수를 제한하거나, TOP(N) 조건을 사용하는 서브쿼리이다.

③ INLINE VIEW 는 FROM 절에 사용되는 서브쿼리로서 실질적인 OBJECT는 아니지만 SQL 문장에서 마치 VIEW 나 테이블처럼 사용되는 서브쿼리이다.

④ 상호연관 서브쿼리는 처리 속도가 가장 빠르기 때문에 최대한 활용하는 것이 좋다.

##### **7. 소계, 중계, 합계처럼 계층적 분류를 포함하고 있는 데이터의 집계에 적합한 GROUP 함수 두 가지는 무엇인가 ?**

① ROLLUP, SUM

② ROLLUP, CUBE

③ GROUPING, SUM

④ CUBE, SUM

##### **8. 그룹 내 순위 관련 WINDOW 함수의 특징이 틀린 것은 무엇인가?**

① RANK 함수는 동일한 값에 대해서는 동일한 순위를 부여한다.

( 같은 등수에 여럿이 존재하는 경우 등수가 SKIP 될 수 있음)

② DENSE\_RANK 함수는 RANK 함수와 흡사하나, 동일한 순위를 하나의 건수로 취합하는 것이 틀린점이다.

( 같은 등수에 여럿이존재하는 경우에도 등수가 SKIP 되지 않음 )

③ CUMM\_RANK 함수는 누적된 순위를 부여할 수 있다.

( 등수를 누적 순위로 표현함 )

④ RANK 함수가 동일한 값에 대해서는 동일한 순위를 부여하는데 반해, ROW\_NUMBER 함수는 고유한 순위를 부여 한다.

( 같은 등수가 존재할 수 없음 )

##### **9. 다음 중 맞는 내용은 무엇인가 ?**

① 유저를 생성하면 생성한 유저로 바로 로그인 할 수 있다.

② 새롭게 생성된 유저라면 조건 없이 새로운 유저를 만들 수 있다.

③ 유저 생성은 누구나 할 수 있지만 권한 설정은 데이터베이스 관리자만 가능하다.

④ 다른 유저의 테이블은 그 테이블에 대한 권한 없이는 조회할 수 없다.

##### **10. 다음 중 절차형 SQL을 이용하여 주로 만드는 것이 아닌 것은 무엇인가?**

① PROCEDURE

② TRIGGER

③ USER DEFINED FUNCTION

④ BUILD-IN FUNCTION

|  |  |  |
| --- | --- | --- |
| **문제** | **정답** | **해설** |
| 1 | ① | EQUI JOIN은 반드시 PK, FK 관계에 의해서만 성립되는 것은 아니다. 조인 칼럼이 1:1로 맵핑이 가능하면 사용할 수 있다. |
| 2 | ② | 여러 테이블로부터 원하는 데이터를 조회하기 위해서는 전체 테이블 개수에서 최소 N-1개 만큼의 JOIN 조건이 필요하다. |
| 3 | ② | UNION ALL 연산자는 조회 결과에 대해 별도의 정렬 작업을 하지 않는다. 또한 중복 데이터에 대해서도 삭제하지 않고 여러 번 중복 표현한다. |
| 4 | ③ | SELF JOIN은 하나의 테이블에서 두 개의 컬럼이 연관 관계를 가지고 있는 경우에 사용한다. |
| 5 | ④ | FROM 절에 정의된 서브쿼리는 INLINE VIEW이다. INLINE VIEW는 일반적으로 메인쿼리보다 먼저 수행되므로 SQL 문장 내에서 절차성을 주는 효과를 얻을 수 있다. |
| 6 | ④ | 상호 연관 서브쿼리는 서브쿼리가 메인쿼리의 행 수 만큼 실행되는 쿼리로서 실행 속도가 상대적으로 떨어지는 SQL문장이다. 그러나, 복잡한 일반 배치 프로그램을 대체할 수 있기 때문에 조건에 맞는다면 적극적인 검토가 필요하다. |
| 7 | ② | ROLLUP, CUBE는 GROUP BY의 확장된 형태로 병렬로 수행이 가능하고 사용하기가 쉽기 때문에 효과적이다. 다차원적인 집계가 필요한 경우는 CUBE를 사용한다. |
| 8 | ③ | 그룹 내 순위 관련 WINDOW FUNCTION으로는 RANK, DENSE\_RANK, ROW\_NUMBER 함수가 있고, ③의 지문은 DENSE\_RANK 함수에 대한 설명이며, CUMM\_RANK 함수는 존재하지 않는다. |
| 9 | ④ | 테이블에 대한 권한은 각 테이블의 소유자가 가지고 있기 때문에 소유자로부터 권한을 받지 않으면 다른 유저의 테이블에 접근할 수 없다. |
| 10 | ④ | 절차형 SQL을 이용하여 PROCEDURE, TRIGGER, USER DEFINED FUNCTION을 만들 수 있다. |