2018 NOVEMBER 18

SQLD 기출문제 모음

21, 24, 25, 26 30 회차

KIM SEOKRAE

SQLD STUDY

내용

I 데이터 모델의 이해 1

1 데이터 모델링의 이해 1

1) 데이터 모델의 이해 1

2) 엔터티(Entity) 1

3) 속성(Attribute) 2

4) 관계(Relationship) 2

5) 식별자(Identifiers) 3

2 데이터 모델과 성능 3

1) 성능 데이터 모델링의 개요 3

2) 정규화와 성능 3

3) 반정규화와 성능 4

4) 대량 데이터에 따른 성능 5

5) 분산 데이터베이스와 성능 5

II SQL 기본 및 활용 5

1 SQL 기본 5

1) 관계형 데이터베이스 개요 5

2) DDL(Data Definition Language) 6

3) DML(Data Manipulation Language) 6

4) TCL(Transaction Control Language) 6

5) Where 절 7

6) 함수(Function) 10

7) Group By, Having 절 14

8) Order by 절 14

9) 조인(Join) 16

2 SQL 활용 17

1) 표준 조인(Standard Join) 17
2) 집합 연산자(Set Operator) 20
3) 계층형 질의와 셀프 조인 21
4) 서브쿼리 23
5) 그룹 함수(Group Function) 28
6) 윈도우 함수(Window Function) 32
7) DCL(Data Control Language) 35
8) 절차형 SQL 36
3 SQL 최적화 기본 원리 36
1) 옵티마이저와 실행계획 36
2) 인덱스 기본 37
3) 조인 수행 원리 38

Ⅰ 데이터 모델의 이해

1 데이터 모델링의 이해

1) 데이터 모델의 이해

4. 데이터 모델링에 대한 아래 보기 설명 중 알맞은 것은? [21회 기출] 1) 논리 모델링의 외래키는 물리 모델에서 반드시 구현되지는 않는다 2) 실제로 데이터베이스를 구축할 때 참고되는 모델은 개념적 데이터 모델링이다 3) 물리 모델링 -> 논리 모델링 -> 개념 모델링 단계로 갈수록 구체적이다 4) 데이터 모델링의 3가지 요소는 Process, Attributes, Relationship 이다
해답 논리 모델링의 외래키는 물리 모델에서 반드시 구현되지 않음. (선택 사항임) 실제 데이터베이스 구축 시 참고되는 모델은 물리적 데이터 모델링이다 개념 모델링에서 물리 모델링으로 가면서 더 구체적이며 개념 모델링이 가장 추상적이다 데이터 모델링의 3요소는 Thing, Attributes, Relationship 이다

5. 데이터 모델링에 대한 단계 중 아래에서 설명하는 단계는 어떤 단계의 모델링인가? [21회 기출] 추상화 수준이 높고 업무중심적이고 포괄적인 수준의 모델링 진행. 전사적 데이터 모델링, EA 수립 시 많이 이용됨 1) 개념적 데이터 모델링 2) 논리적 데이터 모델링 3) 물리적 데이터 모델링 4) 추상적 데이터 모델링
해설

6. 데이터 모델링이 최종적으로 완료된 상태라고 정의할 수 있는, 즉 물리적인 스카마 설계를 하기 전 단계를 가리키는 말은? [기출 30회차 - 데이터 모델링] 보기 + 개괄, 개념적, 논리적, 물리적 데이터 모델링
사전 지식 "논리 데이터 모델링의 결과로 얻어지는 논리 데이터 모델은 데이터 모델링이 최종적으로 완료된 상태라고 정의할 수 있다. 즉 물리적인 스키마 설계를 하기 전 단계의 '데이터 모델' 상태를 일컫는 말이다."
해답: 논리적 모델링

2) 엔터티(Entity)

6. 엔터티 - 인스턴스 - 속성 - 속성값에 대한 관계 설명중 틀린 것을 고르시오 1) 한 개의 엔터티는 두 개 이상의 인스턴스의 집합이어야 한다 2) 한 개의 엔터티는 두 개 이상의 속성을 갖는다 3) 하나의 속성은 하나 이상의 속성값을 가진다 4) 하나의 엔터티의 인스터스는 다른 엔터티의 인스턴스간의 관계인 Paring 을 가진다
해설 하나의 속성은 하나의 속성값을 가지며 하나 이상의 속성값을 가지는 경우 정규화가 필요함

22. 다음주 엔터티의 종류가 아닌 것은? 1) 교수 2) 학생 **3) 청약자** 4) 수강
사전 지식 * 엔터티 기준 - 엔터티는 사람, 장소, 물건, 사건, 개념 등의 명사에 해당한다. - 엔터티는 업무상 관리가 필요한 관심사에 해당한다. - 엔터티는 저장이 되기 위한 어떤 것(Thing)이다.
해답 교수,학생은 논란의 여지가 없으나, 수강과 청약자에서 논란이 일어남 청약자 라는 개념은 관심사라기 보다는 청약의 주체가 되는 속성에 가까운 곳으로 보임. 속성값을 가지기에 애매함 * 참고 : 대다수 수강이라고 했으나 청약자로 체킹한 분이 만점으로 답으로 인정

4. 발생시점에 따른 엔터티 분류에 의한 중심 엔터티가 아닌 것은? [기출 30회차 - 엔터티] 보기 + 매출, 계약, 사원, 주문
사전 지식 "예를 들어 사원 , 부서 , 고객 , 상품 , 자재 등이 기본엔터티가 될 수 있다"
해답: 사원 (기본 엔터티임)

5. 발생시점에 따른 엔터티 분류에 대해서 아래의 내용 중 잘못 짝 지어진 것은? [기출 30회차 - 엔터티] 보기 + 기본/키 엔터티 : 조직 사원 + 기본/키 엔터티 : ??? + 중심 엔터티 : 상품, 주문상품 + …
해답 : 중심 엔터티 : 상품, 주문상품 -> 상품은 기본 엔터티임

3) 속성(Attribute)

1. 도메인의 특징에 대해서 설명한 후 보기에서 고르는 문제 [기출 30회차 - 도메인]
- 답: 도메인

4) 관계(Relationship)

7. 테이블 3개(옵션널 관계) 식별자 관계 3개 테이블에 대한 설명중 부적절한 것은? (답이 두개로 보임) [기출 30회차 - 관계]** 테이블A ----O<- 테이블B ----O<- 테이블C

식별자관계 식별자관계 보기 + B 를 제외하고 A와 C 는 카르테시안 조인이 된다. + 조인 최소 조건은 3개이다 + B, C 에서 C의 데이터는 모두 B에 존재하므로 Outer join 을 안해도 된다 + B 는 A 의 데이터를 모두 포함하지 않는다.
사전 지식 * A, C 는 식별자 관계이므로 카르테시안 조인이 되지 않을 수 있다.
해답: 조인 최소 조건은 N개 테이블일 경우 n-1 개이다 (2개)

5) 식별자(Identifiers)

8. 식별자에 대한 설명으로 잘못된 것은? [기출 30회차 – 식별자] * 주식별자 / 보조 식별자 설명에 대해서 주식별자가 대표성을 띌수 없다는 내용
해답: 주식별자 / 보조 식별자

2 데이터 모델과 성능

1) 성능 데이터 모델링의 개요

2) 정규화와 성능

9. 아래의 ERD 에서 3차 정규형을 만족하게 할 때 엔터티의 개수는 몇개가 되는가?
[21회 기출 – 정규화]

ㄱ. 평가코드,평가내역은 학번에 종속적
ㄴ. 코스명,기간은 코스코드에 종속적
ㄷ. 평가코드 평가내역은 속성간 종속적 관계

-〉 1차 정규형 : 모든 속성은 반드시 하나의 값. 속성값의 중복 제거
-〉 2차 정규형 : 식별자에 종속되지 않는 속성의 중복 제거
-〉 3차 정규형 : 2차 정규형 만족 + 식별자 외 일반 컬럼간의 종속 존재 제거
1) 1개
2) 2개
3) 3개
4) 4개

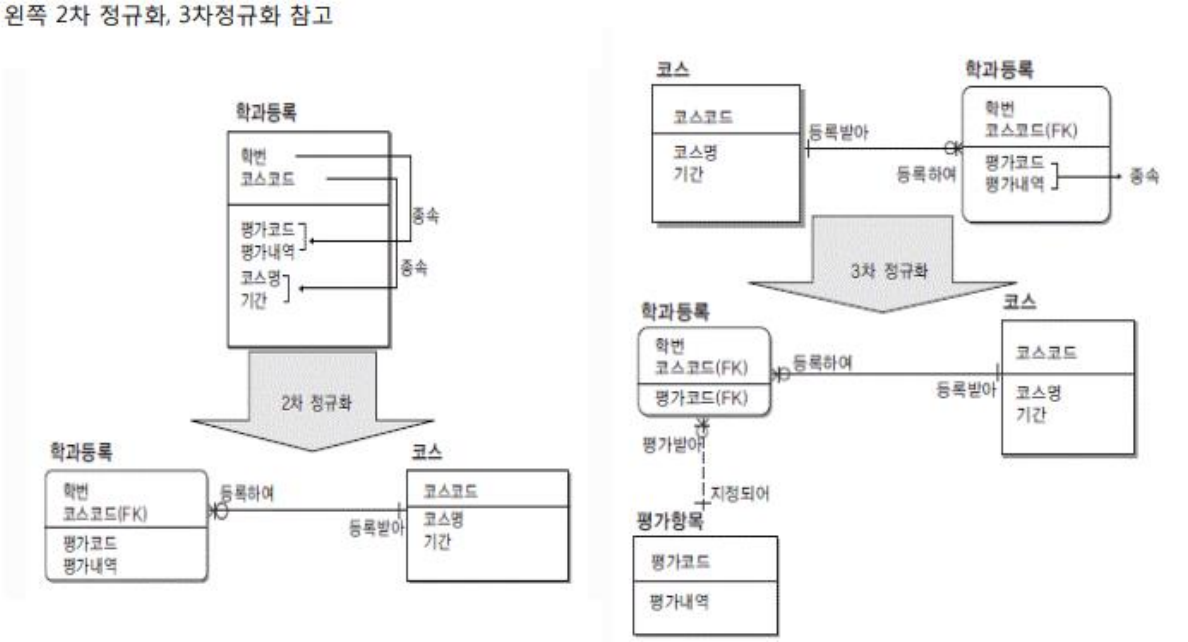
사전 지식
* 1차 정규형
- 모든 속성은 반드시 하나의 값을 가져야 한다. 즉, 반복 형태가 있어서는 안된다.

- 각 속성의 모든 값은 동일한 형식이어야 한다.
- 각 속성들은 유일한 이름을 가져야 한다.
- 레코드들은 서로 간에 식별 가능해야 한다.

* 2차 정규형
- 식별자가 아닌 모든 속성들은 식별자 전체 속성에 완전 종속되어야 한다.
- 이것을 물리 데이터 모델의 테이블로 말하면 기본키가 아닌 모든 칼럼들이 기본키에 종속적이어야 2차 정규형을 만족할 수 있다는 것이다

* 3차 정규형
- 2차 정규형을 만족하고 식별자를 제외한 나머지 속성들 간의 종속이 존재하면 안된다.
- 이것이 3차 정규형을 만족하는 것이다.

왼쪽 2차 정규화, 3차정규화 참고



3) 반정규화와 성능

2. 반정규화에 대한 설명중 부적절한 것은 [기출 30회차 - 반정규화] * 해당 내용은 반정규화 기법임.
사전 지식 "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."
해답 반정규화 전에 테이블 추가(통계 테이블 추가, 중복 테이블 추가, 이력 테이블 추가, 부분 테이블 추가) 를 통해 반정규화를 하지 않음

3. 반정규화 대상이 아닌것은? [기출 30회차 - 반정규화]
사전 지식 "- 자주 사용되는 테이블에 접근(Access) 하는 프로세스의 수가 많고 항상 일정한 범위만을 조회하는 경우

에 반정규화를 검토한다 - 테이블에 대량의 데이터가 있고 대량의 데이터 범위를 자주 처리하는 경우에 처리범위를 일정하게 줄이지 않으면 성능을 보장할 수 없을 경우에 반정규화를 검토한다. - 통계성 프로세스에 의해 통계 정보를 폄요로 할 때 별도의 통계태이블(반정규화 테이블)을 생성한다. - 테이블에 지나치게 많은 조인 (JOIN) 이 걸려 데이터를 조회하는 작업이 기술적으로 야려울 경우 반정규회를 검토한다"
해답 Sorting, Order by 하는 프로세스

4) 대량 데이터에 따른 성능

10. 아래의 설명중 바른 것은? (Row Migration 과 Row Chaining 에 대한 설명) **[기출 30회차 - 대량 데이터에 따른 테이블 분할]**
사전 지식 "많은 칼럼은 로우 체이닝과 로우 마이그레이션이 많아지게 되어 성능이 저하된다. 로우 길이가 너무 길어서 데이터 블록 하나에 데이터가 모두 저장되지 않고 두 개 이상의 블록에 걸쳐 하나의 로우가 저장되어 있는 형태가 로우체이닝 (Row Chaining) 현상이다. 또한 로우 마이그레이션 (Row Migration)은 데이터 블록에서 수정이 발생하면 수정된 데이터를 해당 데이터 블록에서 저장하지 못하고 다른 블록의 빈 공간을 찾아 저장하는 방식이다. 로우 체이닝과 로우 마이그레이션이 발생하여 많은 블록에 데이터가 저장되면 데이터베이스 메모리에서 디스크와 I/O(입력/출력)가 발생할 때 불필요하게 I/O가 많이 발생하여 성능이 저하된다"
해답: ROW Chaining 에 대한 설명이 맞음

5) 분산 데이터베이스와 성능

9. 분산 데이터베이스의 특징 중 저장 장소 명시가 불필요 하다는 특성은 무엇인가? 보기 + 분할 투명성, 위치 투명성, 지역사상 투명성, 병행 투명성
사전 지식 사용하려는 데이터의 저장 장소 명시 불필요. 위치정보가 System Catalog 에 유지되어야 함 지역 사상 투명성: 지역 DBMS와 물리적 DB사이의 Mapping 보장 각 지역시스템 이름과 무관한 이름 사용가능
해답: 위치 투명성

II SQL 기본 및 활용

1 SQL 기본

1) 관계형 데이터베이스 개요

13. DML, DCL, DDL 이 잘못 짝지워진 것은? [기출 30회차 - SQL]
사전 지식 * SELECT 는 DML 임
해답: DCL - SELECT

2) DDL(Data Definition Language)

4. 아래의 SQL 구문을 완성하시오. (SQL SERVER 임) [기출 30회차 - DDL] [SQL] ( ) DEPT ( ) VARCHAR(30) NOT NULL;
사전 지식 * SQL SERVER 에서는 ALTER COLUMN, ORACLE 에서는 MODIFY 임
해답 : ALTER TABLE , ALTER COLUMN

5. 아래와 같은 테이블이 있을 때 모든 SQL 이 수행된 이후의 결과는? [기출 30회차 – DDL] [테이블] TABLE : SQLD_D5 COL1 VARCHAR2(30) COL2 NUMBER [SQL] INSERT INTO SQLD_D5(COL1, COL2) VALUES('ABCD',NULL); INSERT INTO SQLD_D5(COL1, COL2) VALUES('BC',NULL); ALTER TABLE SQLD_D5 MODIFY COL2 DEFAULT 10; INSERT INTO SQLD_D5(COL1, COL2) VALUES('XY',NULL); INSERT INTO SQLD_D5(COL1) VALUES('EXD'); SELECT SUM(COL2) FROM SQLD_D5;
해답 : 10

3) DML(Data Manipulation Language)

4) TCL(Transaction Control Language)

1. 아래와 같은 트랜잭션이 일어났을 때 트랜잭션 종료 후 값이 트랜잭션의 어느 속성을 위배하는가? [기출 24회차 – TCL (트랜잭션)] ( ) Time Session 1 Start 10,000 T1 Update T2 Error T3 Update T4 Commit End 9,000
사전 지식 *** 문제의 예제가 적절하지는 않음. 하나의 트랜잭션에서 중간 Erorr 발생 후 commit 이 되었을 때 그 값이 변했다를 의미함 이러한 현상이 발생하면 어떤 트랜잭션 속성을 위배했는지를 묻는 문제임
1번 해답: 원자성 원자성: 트랜잭션에서 정의된 연산들은 모두 성공적으로 실행되던지 아니면 전혀 실행되지 않은 상태로 남아

있어야 한다
일관성: 트랜잭션이 실행되기 전의 데이터베이스 내용이 잘못 되어 있지 않다면 트랜잭션이 실행된 이후에도 데이터베이스의 내용에 잘못이 있으면 안 된다
고립성: 트랜잭션이 실행되는 도중에 다른 트랜잭션의 영향을 받아 잘못된 결과를 만들어서는 안 된다.
지속성: 트랜잭션이 성공적으로 수행되면 그 트랜잭션이 갱신한 데이터베이스의 내용은 영구적으로 저장된다

1. 아래와 같은 명령어가 순서대로 수행된 이후의 결과를 작성하시오
[기출 25회차 – TCL (Commit)]
〈단계별 SQL 수행〉
- 01 : CREATE TABLE TAB1 (N1 NUMBER, N2 NUMBER) ;
- 02 : INSERT INTO TAB1 VALUES (1,1);
- 03 : INSERT INTO TAB1 VALUES (2,2);
- 04 : CREATE TABLE TAB2 (V1 VARCHAR2(10), V2 VARCHAR2(10));
- 05 : ROLLBACK;
- 06 : SELECT COUNT(*) FROM TAB1;

사전 지식
Oracle 의 경우 DML 후 자동 COMMIT 아니나,
DDL 발생하면 DML이 은 암묵적인 COMMIT이 자동으로 발생되어 전체 트랜잭션이 COMMIT 됨.
DDL이후 ROLLBACK이 진행되었으나 이미 COMMIT 이 되었으므로 COUNT 는 2 가 됨
SQL Server 의 경우 자동 COMMIT 이므로 역시 COUNT 는 2가 됨.(문제는 Oracle 이라고 나왔던 것으로 추정됨

1번 해답: 2

9. 아래와 같은 SQL 이 순서대로 수행되고 난 후 결과값으로 알맞은 것은? [기출 30회차 – TCL(Rollback)]
[SQL]
(1) INSERT INTO TAB1 VALUES (1);
(2) INSERT INTO TAB1 VALUES (2);
(3) SAVEPOINT SV1;
(4) UPDATE TAB1 SET COL1=4 WHERE COL1=2;
(5) SAVEPOINT SV1;
(6) DELETE TAB1 WHERE COL1 〉=2;
(7) ROLLBACK TO SV1;
(8) INSERT INTO TAB1 VALUES (3);
(9) SELECT MAX(COL1) FROM TAB1;

사전 지식
* SV1 이 같을 경우 가장 아래에 있는 쪽으로 ROLLBACK 됨

해답: 4

5) Where 절

13. 보기의 테이블 TAB_A, TAB_B 에 INSERT 를 한 결과로 알맞은 것은?
[기출 21회 – 제약조건의 종류 Primary key, Unique key, Not null, Check, Foreign key]

TAB_A	A IDENTITY (1,1)	TAB_B	A CHECK (A 〈 5)

(Sql Server) B VARCHAR2(10)
INSERT INTO TAB_A(A,B) VALUES(1,'A');
INSERT INTO TAB_A(B) VALUES('B');
INSERT INTO TAB_A(B) VALUES('D');

(Oracle) B VARCHAR2(10)
INSERT INTO TAB_B VALUES(1,'A');
INSERT INTO TAB_B VALUES(2,'B');
INSERT INTO TAB_B VALUES(6,'D');
INSERT INTO TAB_B VALUES(NULL,'X');

1)

TAB_A	TAB_B
1,B	1,A
2,D	2,B
	NULL,X

2)

TAB_A	TAB_B
1,A	1,A
2,B	2,B
3,D	6,D

3)

TAB_A	TAB_B
1,A	1,A
	2,B

4)

TAB_A	TAB_B
1,B	1,A
2,D	2,B

사전 지식

* SQL SERVER : IDENTITY [(seed , increment)]
  - SEED : 첫번째 행이 입력될 때의 시작값
  - 증가되는 값
 - 해당 컬럼에 값을 넣을 경우 Error 발생 (Error 발생을 막기 위해서는 IDENTITY_INSERT 를 OFF 로 하면 되나 이런 경우 IDENTITY 를 쓴 의미가 없어짐)

해설

SQL SERVER 의 IDENTITY 로 지정된 컬럼에는 값을 넣으면 에러가 남
ORACLE 의 CHECK 조건을 만족하지 못할 경우 에러가 나나 NULL 은 무시됨

18. 비교연산자의 어느 한쪽이 VARCHAR 유형 타입인 경우 문자 유형 비교에 대한 설명 중 가장 알맞지 않은 것은? [기출 21회차 - 비교연산자 (문자유형비교)]

1) 서로 다른 문자가 나올 때까지 비교한다
2) 길이가 다르다면 짧은 것이 끝날 때까지만 비교한 후에 길이가 긴 것이 크다고 판단한다
3) 길이가 같고 다른 것이 없다면 같다고 판단한다
4) 길이가 다르다면 작은 쪽에 SPACE 를 추가하여 길이를 같게 한 후에 비교한다

[표 Ⅱ-1-18] 문자 유형 비교 방법

구분	비교 방법
비교 연산자의 양쪽이 모두 CHAR 유형 타입인 경우	길이가 서로 다른 CHAR형 타입이면 작은 쪽에 SPACE를 추가하여 길이를 같게 한 후에 비교한다.
	서로 다른 문자가 나올 때까지 비교한다.
	달라진 첫 번째 문자의 값에 따라 크기를 결정한다.
	BLANK의 수만 다르다면 서로 같은 값으로 결정한다.
비교 연산자의 어느 한 쪽이 VARCHAR 유형 타입인 경우	서로 다른 문자가 나올 때까지 비교한다.
	길이가 다르다면 짧은 것이 끝날 때까지만 비교한 후에 길이가 긴 것이 크다고 판단한다.
	길이가 같고 다른 것이 없다면 같다고 판단한다.
	VARCHAR는 NOT NULL까지 길이를 말한다.
상수값과 비교할 경우	상수 쪽을 변수 타입과 동일하게 바꾸고 비교한다.
	변수 쪽이 CHAR 유형 타입이면 위의 CAHAR 유형 타입의 경우를 적용한다.
	변수 쪽이 VARCHAR 유형 타입이면 위의 VARCHAR 유형 타입의 경우를 적용한다.

해설

SAPCE 를 추가하여 길이를 맞춰 비교하는 방법은 CHAR 타입인 경우임

5. 아래와 같은 SQL이 있을 때 조건절을 넣기 위한 키워드는 무엇인지 작성하시오.

[기출 21회차 – where 절]
SELECT *
FROM EMP
( ) EMPID = 10;

5번 해답: WHERE

2. 아래와 같은 테이블에 대한 수행 결과가 다음과 같을 때 빈칸을 완성하시오
[기출 25회차 – Where 절]
〈테이블 구조〉

TAB1	COL1	COL2	
	A01	10	
	A02	20	
`	A03	30	
	NULL	40	** 문자열이 아닌 NULL 임
	NULL	50	** 문자열이 아닌 NULL 임

〈실행 결과〉

CO1	CO2
A02	20
A03	30
NULL	40
NULL	50

〈수행 SQL〉
SELECT COL1, COL2
FROM TAB1
WHERE ( );

사전 지식
WHERE 절에 조건절이 쓰이게 되면 암묵적으로 해당 컬럼에 대한 IS NOT NULL 조건이 생성됨
수행 결과에 NULL 이 포함되어 있으므로 NVL 함수로 NULL 에 대한 처리를 명시적으로 해줘야 함

2번 해답 : NVL(COL1,'X') 〈〉 'A01'

1. ERD 를 참고하여 아래의 SQL 중 오류가 날 수 있는 SQL 을 고르시오.
[기출 30회차 – WHERE 제약조건]
[ERD]
TABLE A ----〈- TABLE B / 컬럼 정보 및 데이터

사전 지식
* TABLE B 의 컬럼 중 문자열 컬럼에 대해서 숫자 비교를 하는 SQL 이 답임
* 문자와 숫자가 만나면 문자가 숫자로 묵시적 형변환이 일어남. 해당 문자열에는 숫자 뿐만 아니라 문자가 존재함

해답: .... WHERE 문자열컬럼 〉 0

4. 데이터가 주어지고 WHERE 절에서 WHERE (COL1,COL2) in ((1000,'ABC')) 결과로 알맞은 것은?
[기출 30회차 – WHERE]

해답: 위의 조건은 COL1, COL2 가 동시에 1000, 'ABC' 인 것을 찾는 것으로 데이터상 한건이 도출됨

5. 데이터가 주어지고 WHERE 절에서 WHERE COL1 in (1,2, NULL) 결과로 알맞은 것은? [기출 30회차 – WHERE]
사전 지식 * NULL 은 비교에서 애초에 제외 되며 IN (...) 안에 NULL 이 있어도 NULL 비교는 되지 않음
해답 : IN (....) 의 NULL 이 있을 경우는 COL1 IN (1,2) 와 동일한 결과임

6) 함수(Function)

8. 아래의 SQL 에서 FUNCTION 자리에 쓰인 함수에 의한 결과값이 다른 하나는? [기출 21회 – 숫자형 함수 ABS, SIGN, MOD, CEIL, FLOOR, ROUND, TRUNC, SIN, COS, TAN..., EXP(), POWER(), SQRT(), LOG(), LN()] SELECT function(3.46) FROM DUAL; 1) TRUNC 2) CEIL 3) FLOOR 4) ROUND
사전 지식 * FLOOR: 숫자보다 작거나 같은 최대 정수를 리턴 * CEIL/CEILING: 숫자보다 크거나 같은 최소 정수를 리턴 * TRUNC: 숫자를 소수 m 자리에서 잘라서 리턴 (m default: 0) * ROUND: 숫자를 소수 m 자리에서 반올림하여 리턴 (m default: 0)
해답 TRUNC(3.46) = 3, FLOOR(3.46) = 3 ROUND(3.46)=3 CEIL(3.46) = 4

19. 아래의 SQL 에 대해서 결과값이 다른 것은? [기출 21회차 – 문자형 함수 lower(), upper(), ascii(), char(), concat(), substr(), length(), ltrim(), rtrim(), trim()] 1) SELECT CONCAT ('RDBMS', ' SQL') FROM DUAL; 2) SELECT 'RDMBS' || ' SQL' FROM DUAL; 3) SELECT 'RDBMS' + ' SQL' FROM DUAL; 4) SELECT 'RDBMS' & ' SQL' FROM DUAL;
해답 Oracle 의 결합 함수 : CONCAT / || SQL Server : +

1. 아래와 같은 테이블이 있을때 아래와 같은 결과가 나오기 위한 주어진 SQL 구문을 완성하시오. [기출 21회차 – 문자형 함수] TABLE SQLD_21_01 TABLE SQLD_21_02

```
N1      V1                          N1      V1
------- --------                    ------- --------
1       a                           1       A
2                                   2
3       b                           3       B
4       c
SELECT A.*
FROM SQLD_21_01 A,
        SQLD_21_02 B
WHERE (                     ) (A.V1) LIKE B.V1||'%'

RESULT>         N1      V1
                1       a
                3       b
```

해답: UPPER
소문자 / 대문자로 되어 있는 경우 두개의 문자열이 같기 위해서 적용해야 하는 함수 UPPER 를 사용

유사 문제
유사 문제 만약 주어진 테이블의 데이터가 아래와 같을 경우는 빈칸 제거가 답임

```
        TABLE SQLD_21_01                TABLE SQLD_21_02
        N1        V1                    N1        V1
        --------- ----------            --------- ----------
        1         A (스페이스 뒤에 있음)    1       A
        2                                       2
        3         B (스페이스 뒤에 있음)    3       B
        4         C (스페이스 뒤에 있음)

SELECT A.*
FROM SQLD_21_01 A,
     SQLD_21_02 B
WHERE (       )(A.V1) LIKE B.V1
-> TRIM
```

1. 아래와 같은 테이블이 있을때 아래의 SQL 결과에 대해서 Oracle, SQL Server 순서로 적으시오.
[기출 26회차 - length(), len()]

```
TAB1    COL1 CHAR(10)
        COL2 CHAR(10)
INSERT INTO TAB1 VALUES ('1','');
INSERT INTO TAB1 VALUES ('2','');
INSERT INTO TAB1 VALUES ('3','');
 -- Oracle                                    -- SQL Server
SELECT SUM(LENGTH(COL1) ) , SUM(LENGTH(COL2) )      SELECT    SUM(LEN(COL1)    )    ,
SUM(LEN(COL2) )
FROM TAB1;                                     FROM TAB1;
```

사전 지식

〈SQL Server LEN 함수〉
SELECT LEN('TechOnTheNet.com'); SELECT LEN('TechOnTheNet.com ');
Result: 16 Result: 16 (trailing spaces are not included in the calculation)

SELECT LEN(' TechOnTheNet.com'); SELECT LEN(' TechOnTheNet.com ');
Result: 19 Result: 19 (trailing spaces are not included in the calculation)

SELECT LEN(' ');
Result: 0 (trailing spaces are not included in the calculation)

SELECT LEN(''); SELECT LEN(NULL);
Result: 0 Result: NULL

1번 해답 : Oracle (30, Null) / SQL Server (3, 0)
CHAR형의 경우 고정형으로 사이즈에 비해 작은 값이 들어올 경우 나머지 사이즈를 ' '(스페이스) 로 채운다
Oracle 의 경우 length 함수는 ' ' (스페이스) 를 1로 인식하고
SQL Server 의 len 함수는 문자 뒤의 ' ' (스페이스)를 0으로 인식한다.
즉 SQL Server 의 Len 은 문자열 뒤의 빈칸은 계산하지 않는다.

2. 아래와 같이 문자와 문자를 연결하는 함수에 대해서 작성하시오.
[기출 26회차 – 문자형 함수 lower(), upper(), ascii(), char(), concat(), substr(), length(), ltrim(), rtrim(), trim()]
SELECT ( ) (COL1, COL2)
FROM EMP
WHERE EMPID = 10;

2번 해답 : CONCAT

3. 아래와 같은 테이블이 있을때 아래와 같은 SQL 의 수행결과에 대해 작성하시오
[기출 26회차 – where 절]

TAB1	COL1	COL2
	Z	10
	Y	20
`	X	30

SELECT COUNT(*)
FROM TAB1
WHERE COL1 = 'X'
AND COL2 〈 20;

3번 해답: 0

2. EMP 테이블은 사원과 매니저의 정보를 담은 계층형 데이터를 포함한 테이블이다.
매니저부터 사원까지 결제 단계가 가장 많은 레벨을 구하려고 할 때 빈칸을 완성하시오
[기출 21회차 – 함수]
SELECT ( )
FROM SCOTT.EMP

START WITH MGR IS NULL
CONNECT BY PRIOR EMPNO = MGR;

2번 해답: MAX(LEVEL)
계층형 쿼리에서 최대 계층의 수를 구하기 위한 문제. MAX (LEVEL) 을 사용하여 최대 계층 수를 구함

7. SELECT nvl(count(*), 9999) from table where 1=2 의 결과값은? [기출 30회차 - NULL관련 함수]

사전 지식
* COUNT(*) 인 경우 공집합일때 0 을 반환하고 다른 집계함수의 경우는 NULL 을 반환함

해답: 0

11.SELECT 결과가 NULL 이 아닌 경우는? [기출 30회차 – Null 처리 함수]

[보기]
+ COALESCE ('AB','BC','CD')
+ CASE WHEN 'AB' THEN 'BC' ELSE 'CD' END...
+ DECODE ('AB','CD','DE')
+ NULLIF ('AB','CD')

사전 지식
- COALESCE 함수는 인수의 숫자가 한정되어 있지 않으며, 임의의 개수 EXPR에서 NULL이 아닌 최초의 EXPR을 나타낸다. 만일 모든 EXPR이 NULL이라면 NULL을 리턴 한다.
- NVL/ISNULL
- NULLIF 함수는 EXPR1이 EXPR2와 같으면 NULL을, 같지 않으면 EXPR1을 리턴 한다.
  특정 값을 NULL로 대체 하는 경우에 유용하게 사용할 수 있다. NULLIF (EXPR1, EXPR2)

해답: COALESCE ('AB','BC','CD')

12. 아래의 데이터가 있을 때 SQL 이 수행된 결과로 옳은 것은? [기출 30회차 – COALESCE함수]

[데이터]

```
COL1        COL2
--------------
   100       100
 NULL         60
 NULL       NULL
```

[SQL]

```
SELECT COALESCE(COL1, COL2 *50 , 50) FROM TAB1;
```

해답 : 100 / 3000 / 50

15. 문자열이 들어갈 때 빈 공간을 채우는 형태의 데이터 타입은? [기출 30회차 – 문자형 함수]

[보기]
+ VARCHAR2, CHAR, DATE, NUMBER

해답 : CHAR

1. SELECT ABS(-3.8), FLOOR(3.8), TRUNC(3.8), ROUND(3.8) FROM DUAL; 에 대한 값을 구하시오. [기출 30회차 - 함수]
해답 : 3.8 , 3, 3, 4

8. 아래의 SQL 결과를 작성하시오. [기출 30회차 – Upper()] [SQL] SELECT Upper(Sqldeveloper) FROM DUAL;
해답 : SQLDEVEOPER

7) Group By, Having 절

4. 아래와 같은 테이블이 있을 때 하루에 2번 이상 주문하는 대상을 조회하는 SQL 을 완성하시오 [기출 26회차 - Having Count(*)] TAB1 주문번호 주문일자 고객명 ` …. SELECT 주문일자, 고객명, COUNT(*) FROM TAB1 GROUP BY 주문일자,고객명 ( )
4번 해답: HAVING COUNT(*) >= 2

8. 아래의 데이터가 있을때 아래의 SQL 에 대한 결과값은? [기출 30회차 – Group by 절] [데이터] col1 col2 ---------- 1 100 1 null 2 100 2 200 [SQL] SELECT A, SUM(X) FROM TAB GROUP BY A;
해답 : 1 100 / 2 300

8) Order by 절

16. 아래의 SQL 에 대해서 실행 순서를 올바르게 나열한 것은? [기출 21회차 - Select 문장 실행 순서] SELECT DEPTNO, COUNT(EMPNO) FROM SCOTT.EMP WHERE SAL >= 500 GROUP BY DEPTNO

HAVING COUNT(EMPNO) 〉 2
ORDER BY DEPTNO;

1) FROM -〉 WHERE -〉 GROUP BY -〉 HAVING -〉 SELECT -〉 ORDER BY
2) FROM -〉 WHERE -〉 GROUP BY -〉 HAVING -〉 ORDER BY -〉 SELECT
3) FROM -〉 WHERE -〉 GROUP BY -〉 HAVING -〉 SELECT -〉 ORDER BY
4) FROM -〉 WHERE -〉 GROUP BY -〉 SELECT -〉 HAVING -〉 ORDER BY

사전 정보

5. SELECT 칼럼명 [ALIAS명]
1. FROM 테이블명
2. WHERE 조건식
3. GROUP BY 컬럼 (Column) 이나 표현식
4. HAVING 그룹조건식
6. ORDER BY 컬럼 (Column) 이나 표현식

해설

1. 발췌 대상 테이블을 참조한다 (FROM)
2 발췌 대상 데이터가 아닌 것은 제거한다 (WHERE)
3 행들을 소그룹화 한다 (GROUP BY)
4 그룹핑된 값의 조건에 맞는 것만을 출력 한다 (HAVING)
5 데이터 값을 출력/계산한다 (SELECT)
6 데이터를 정렬한다 (ORDER BY)

20. 아래의 ORACLE SQL 을 SQL SERVER SQL 로 전환한 것중 가장 알맞은 것은?
[기출 21회차 - Top N 쿼리, Rownum]

```
SELECT ENAME, SAL
FROM (SELECT ENAME, SAL
          FROM SCOTT.EMP
          ORDER BY SAL DESC)
WHERE ROWNUM 〈 4 ;
```

1)
```
SELECT TOP(4) ENAME,SAL
FROM SCOTT.EMP
ORDER BY SAL DESC
```

2)
```
SELECT TOP(3) ENAME,SAL
FROM SCOTT.EMP
ORDER BY SAL DESC
```

3)
```
SELECT TOP(4) WITH TIES ENAME,SAL
FROM SCOTT.EMP
ORDER BY SAL DESC
```

4)
```
SELECT TOP(3) WITH TIES ENAME,SAL
FROM SCOTT.EMP
ORDER BY SAL DESC
```

사전 지식

* TOP (Expression) [PERCENT] [WITH TIES]
  - WITH TIES: ORDER BY 조건 기준으로 TOP N 의 마지막 행으로 표시되는 추가 행의 데이터가 같을 경우 N+ 동일 정렬 순서 데이터를 추가 반환하도록 지정하는 옵션 (마지막 기준 공통일 경우 모두 출력)

해답

ROWNUM < 4 의 의미는 3건까지 출력이므로 TOP(3)와 같다.
WITH TIES 를 사용할 경우 동일 데이터가 있을 때 추가 건수가 출력되는 현상이 가능함

16. 아래의 SQL 과 같은 SQL 은? [기출 30회차 – Top()]

[SQL]

```
SELECT TOP(10) ENAME, SAL
FROM EMP
ORDER BY SAL;
```

사전 지식

(TOP() 과 ORACLE 의 ROWNUM, ORDER BY SQL 에 관련된 문제)

```
* SELECT ..
  FROM (SELECT ... )
  -> 같은 SELECT 내에 있는 것을 같은 레벨이라 함
* 답 예시
  SELECT ...
  FROM (SELECT .. FROM .. ORDER BY )
  WHERE ROWNUM <= 10
```

해답: 보기에서 ORDER BY 구문과 ROWNUM 이 같은 레벨에 없는 SQL

7. 아래와 같은 결과가 나오도록 SQL 을 작성하시오. [기출 30회차 – Order by]

[결과]

회원ID	RANK	주문금액
B	1	450
G	2	255
F	2	255
H	3	100

[SQL]

```
SELECT 회원ID,
       DENSE_RANK() OVER(ORDER BY (     ) ),
       주문금액
FROM SQLD_D7;
```

사전 지식

* 주문금액으로 ORDER BY 시 동일 주문금액에 대해서 동일 등수가 매겨질 수 있음

해답 : 주문금액 DESC

9) 조인(Join)

26. SQL 구문에서 FROM 절에 대한 설명 중 가장 올바르지 않은 것은?
[기출 21회차 – FROM에 대한 내용이 어디에 딱 정해져 있지 않다.]

1) FROM 절에 ALIAS 를 쓰기 위해서 AS 키워드 사용이 가능하다
2) FROM 은 가장 먼저 수행된다

3) FROM 절에 사용되는 subquery 를 보통 inline view 라고 한다
4) FROM 절은 SELECT 와 항상 짝을 이룬다

해답
1) FROM 절에 ALIAS 를 쓰기 위해서 AS 키워드 사용이 가능하다

2 SQL 활용

1) 표준 조인(Standard Join)

11. 다음의 SQL 을 표준 ANSI SQL 로 알맞게 바꾼것은? 단, 조인 조건과 조회 조건은 분리한다.
[기출 21회 JOIN INNER JOIN, NATURAL JOIN, USING 조건절, ON 조건절, CROSS JOIN, OUTER JOIN]

```
SELECT *
FROM SCOTT.EMP A,
SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
AND B.DNAME = 'SALES'
```

1)
```
SELECT *
FROM SCOTT.EMP A
LEFT OUTER JOIN SCOTT.DEPT B
ON ( A.DEPTNO = B.DEPTNO
      AND B.DNAME ='SALES')
WHERE 1=1;
```

2)
```
SELECT *
FROM SCOTT.EMP A
RIGHT OUTER JOIN SCOTT.DEPT B
ON ( A.DEPTNO = B.DEPTNO
AND B.DNAME ='SALES')
WHERE 1=1;
```

3)
```
SELECT *
FROM SCOTT.EMP A
INNER JOIN SCOTT.DEPT B
ON ( A.DEPTNO = B.DEPTNO
      AND B.DNAME ='SALES')
WHERE 1=1;
```

4)
```
SELECT *
FROM SCOTT.EMP A
INNER JOIN SCOTT.DEPT B
ON A.DEPTNO = B.DEPTNO
WHERE 1=1
AND B.DNAME ='SALES';
```

사전 지식
ANSI SQL 에서 조인 조건절 (ON 절) 에 사용된 조건절은 조인 전 조건으로 작용한다
ON 절 이후 WHERE 절에서 쓰인 조건절은 조인후 조건절로 사용된다.

해답
조인 조건과 조회 조건이 분리되어야 하므로 조회 조건은 WHERE 절로 분리되어야 함
OUTER JOIN 이 아닌 INNER JOIN 이므로 INNER JOIN 에 조회 조건 분리가 된 4번이 맞음

21. 아래의 ANSI JOIN SQL 에서 가장 올바르지 않은 것은? [기출 21회차 – Natural Join]

1)
```
SELECT EMP.DEPTNO, EMPNO, ENAME, DNAME
FROM EMP INNER JOIN DEPT
          ON EMP.DEPTNO = DEPT.DEPTNO;
```

2)

```
SELECT EMP.DEPTNO, EMPNO, ENAME, DNAME
FROM EMP NATURAL JOIN DEPT;
```

3)

```
SELECT *
FROM DEPT JOIN DEPT_TEMP
USING (DEPTNO);
```

4)

```
SELECT E.EMPNO, E.ENAME, E.DEPTNO, D.DNAME
FROM EMP E INNER JOIN DEPT D
        ON (E.DEPTNO = D.DEPTNO);
```

해답

NATURAL JOIN 에서 사용된 열은 식별자를 가질 수 없음.
즉 EMP.DEPTNO 와 같이 OWNER 명을 사용하면 에러가 남

24. 아래와 같은 테이블 TAB1, TAB2 가 있을 때 아래의 SQL의 결과 건수를 알맞게 나열한 것은?
[기출 21회차 – JOIN]

TAB1	COL1	COL2	KEY1
	BBB	123	B
	DDD	222	C
	EEE	233	D
	FFF	143	E

TAB2	KEY2	COL1	COL2
	A	10	BC
	B	10	CD
	C	10	DE

```
SELECT * FROM TAB1 A INNER JOIN TAB2 B ON (A.KEY1 = B.KEY2)
SELECT * FROM TAB1 A LEFT OUTER JOIN TAB2 B ON (A.KEY1 = B.KEY2)
SELECT * FROM TAB1 A RIGHT OUTER JOIN TAB2 B ON (A.KEY1 = B.KEY2)
SELECT * FROM TAB1 A FULL OUTER JOIN TAB2 B ON (A.KEY1 = B.KEY2)
SELECT * FROM TAB1 A CROSS JOIN TAB2 B
```

1) 2, 4, 3, 5, 12
2) 2, 4, 5, 3, 12
3) 2, 3, 4, 5, 12
4) 2, 4, 3, 7, 12

사전 지식

FULL OUTER JOIN = LEFT OUTER JOIN UNION RIGHT OUTER JOIN
Cross join은 곱집합 (TAB1 4 * TAB2 3) = 12

해답: 1번 2, 4, 3, 5, 12

27. 아래의 SQL 의 결과로 알맞은 것은? [기출 21회차 – Non Equal Join]

TABLE SQLD_21_01		TABLE SQLD_21_02	
N1	V1	N1	V1
------	----------	------	----------

1	A	1	A
2		2	
3	B	3	B
4	C		

```
SELECT SUM(A.N1)
FROM SQLD_21_01 A,
          SQLD_21_02 B
WHERE A.V1 <> B.V1;
```

1) 10
2) 30
3) 12
4) 8

사전 지식

Non Equal Join 의 경우는 조인 조건을 제외한 Cross 조인 후
조인 조건을 필터 조건으로 처리하는 것이 좋음

해답

Cross Join 결과 12건에 대한 조건 체크(조건 체크시 NULL 제외) 후 SUM

A	A	1
A		1
A	B	1
	A	2
		2
	B	2
B	A	3
B		3
B	B	3
C	A	4
C		4
C	B	4

SUM = 12

3. 아래와 같은 테이블에 대한 수행 SQL 결과에 대해서 작성하시오 [기출 25회차 – Join]

<테이블 구조>

TAB1	COL1	COL2	TAB2	COL1	COL3
	A01	10		A01	가
	A02	20		A02	나
`	A03	30	`	A03	다
	A10	40		A04	라
	A11	50		A05	마

<수행 SQL>

```
SELECT COUNT(*)
FROM (
 SELECT A.COL1, B.COL3
```

```
FROM TAB1 A INNER JOIN TAB2 B
                    ON (A.COL1 = B.COL1)
UNION
SELECT A.COL1, B.COL3
FROM TAB1 A LEFT OUTER JOIN TAB2 B
                     ON (A.COL1 = B.COL1)
UNION
SELECT COL1, COL3
FROM TAB1 NATURAL INNER JOIN TAB2
);
```

3번 해답: 5

20. 아래 ERD 를 참고하여 모든 회원의 일자별 총 주문금액을 구하는 SQL 중 잘못된 SQL 은?
[기출 30회차 – JOIN]

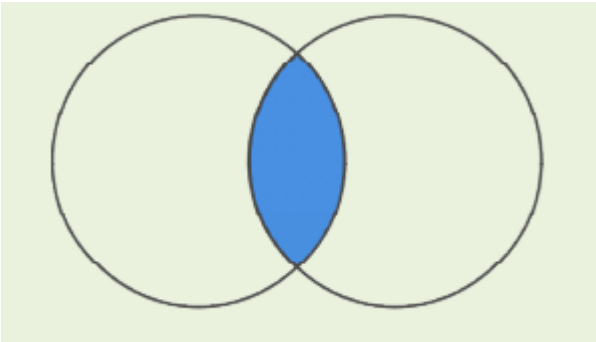
[ERD]
테이블A ----O<- 테이블B ->O---- 테이블C
* optional 관계로 연결되어 있음
[보기]
+ 스칼라 서브쿼리로 합 구하는 SQL
+ OUTER JOIN 으로 합 구하는 SQL
+ INNER JOIN 으로 합 구하는 SQL
+ UNION ALL 로 위는 EXIST, 아래는 JOIN 형태

사전 지식
* 모든 회원을 구하라고 했으나 일부 INNER JOIN 인 경우 일부 회원이 누락됨

해답: INNER JOIN 형태 SQL

2) 집합 연산자(Set Operator)

4. 아래와 같은 SQL Set Operation (색칠부분) 이 무엇인지 작성하시오. ( )
[기출 24회차 – Set Operator]



4번 해답: INTERSECT(교집합)

18. SQL SET OPERATOR 에서 중복 제거를 위해 정렬 작업을 하지 않는 집합 연산자는?
[기출 30회차 – Set Operator]

해답: UNION ALL

21. 아래의 SQL 의 로우의 건수로 올바른 것은? [기출 30회차 – Set Operator(Union All)]
[데이터] TABLE A COL1 COL2 ---------- 1 2 1 2 1 3 TABLE B COL1 COL2 ---------- 1 2 1 4 1 5 [SQL] SELECT DISTINCT COL1, COL2 FROM TAB_A UNION ALL SELECT COL1, COL2 FROM TAB_B;
해답:

3) 계층형 질의와 셀프 조인

5. 아래와 같은 계층구조로 되어있는 데이터에 대해서 아래 SQL 수행 결과를 작성하시오. [기출 24회차 – 계층형 질의]
(,) 〈테이블 구조〉 TAB1 EMP_ID MGR_ID DEPT_NM A HEADROOM B A HEADROOM ` C A HEADROOM D C TECH E C TECH 〈수행 SQL〉 SELECT COUNT(EMP_ID), COUNT(MGR_ID) FROM TEST_24_10 WHERE 1=1 START WITH MGR_ID IS NULL CONNECT BY PRIOR EMP_ID = MGR_ID;
5번 해답 : 5, 4

6. 아래와 같은 계층구조로 되어있는 데이터에 대해서 계층형 결과를 보여주는 SQL 을 작성하시오.
[기출 24회차 - 계층형 질의]
〈테이블 구조〉

TAB1	EMP_ID	MGR_ID	DEPT_NM
	A		HEADROOM
	B	A	HEADROOM
`	C	A	HEADROOM
	D	C	TECH
	E	C	TECH

〈수행 결과〉

LEVEL	사원	관리자	ISLEAF	DEPT_NM
1	A		0	HEADROOM
2	B	A	1	HEADROOM
2	C	A	0	HEADROOM
3	D	C	1	TECH
3	E	C	1	TECH

〈수행 SQL〉

```
SELECT LEVEL, LPAD(' ', 4 * (LEVEL-1)) || EMP_ID 사원 ,
  MGR_ID, CONNECT_BY_ISLEAF ISLEAF,
  DEPT_NM
FROM TEST_24_10
WHERE 1=1
START WITH MGR_ID IS NULL
CONNECT BY PRIOR (                    );
```

6번 해답 : EMP_ID = MGR_ID

23. 아래의 계층형 SQL 에서 리프 데이터이면 1, 그렇지 않으면 0을 출력하고 싶을 때 사용하는 키워드로 알맞은 것은? [기출 21회차 - 계층형 질의]

```
SELECT LEVEL, LPAD(' ',4 * (LEVEL -1) ) || EMPNO,
            MGR, (     ) AS ISLEAF
FROM SCOTT.EMP
START WITH MGR IS NULL
CONNECT BY PRIOR EMPNO = MGR;
```

1) CONNECT_BY_ISLEAF
2) CONNECT_BY_ISCYCLE
3) SYS_CONNECT_BY_PATH
4) CONNECT_BY_ROOT

사전 지식

CONNECT_BY_ISLEAF: 전개 과정에서 해당 데이터가 리프 데이터면 1, 아니면 0
CONNECT_BY_ISCYCLE: 전개 과정에서 자식을 갖는데, 해당 데이터가 조상으로 존재하면 1, 그렇지 않으면 0, 여기서 조상이란 자신으로부터 루트까지의 경록에 존재하는 데이터를 말함
SYS_CONNECT_BY_PATH: 하위 레벨의 컬럼까지 모두 표시해줌 (구분자 지정 가능)

CONNECT_BY_ROOT: Root 노드의 정보를 표시
해답 CONNECT_BY_ISLEAF : 전개 과정에서 해당 데이터가 리프 데이터면 1,아니면 0

17. 아래의 계층형 SQL 수행했을 때 아래와 같은 결과가 나올때 ( ) 을 채우시오. **[기출 30회차 - 계층형 질의]** [SQL] SELECT LEVEL, LPAD('** ', (LEVEL -1) * 2, ' ') || 사원 AS 사원트리, 관리자, 사원 FROM SQLD_17 START WITH ( ) CONNECT BY PRIOR ( )
해답: 관리자 IS NULL, 사원 = 관리자

3. 아래와 같은 데이터가 들어있는 테이블에서 아래 SQL이 수행되었을 때의 결과 건수는? **[기출 30회차 - 계층형 질의]** [데이터] 계층번호 상위계층번호 ---------- 1 null 2 null 4 1 5 1 6 2 7 2 8 4 9 5 10 6 11 7 [SQL] (SQL 은 조건절 외에는 가상으로..) SELECT LEVEL, LPAD('** ', (LEVEL -1)*2 ,' ') || 계층번호 AS 계층트리, 계층번호, 상위계층번호 FROM SQLD_D3 START WITH 상위계층번호 IS NULL CONNECT BY 계층번호 = PRIOR 상위계층번호;
해답 : 2

4) 서브쿼리

1. 아래의 쿼리를 만족하는 결과를 가장 잘 설명한 것은?
[기출 21회 - 다중 행 비교 연산자 IN, ALL, ANY, SOME, EXISTS]

```
SELECT A.*
FROM HR.EMPLOYEES A,
        HR.EMPLOYEES B
WHERE 1=1
AND A.MANAGER_ID = B.EMPLOYEE_ID
AND B.SALARY >= ANY A.SALARY;
```

1) 어떤 부하 직원보다도 연봉이 높은 상사
2) 어떤 부하 직원보다도 연봉이 낮은 상사
3) 어떤 상사 보다도 연봉이 높은 부하 직원
4) 어떤 상사 보다도 연봉이 낮은 부하 직원

사전 지식
* 다중 행 비교 연산자 ANY (서브쿼리)
 - 서브쿼리의 결과에 존재하는 어느 하나의 값이라도 만족하는 조건을 의미.
 비교연산자로 " > " 를 사용했다면 메인쿼리는 서브쿼리의 값들 중 어떤 값이라도
 만족하면 되므로 서브쿼리의 결과의 최소값보다 큰 모든 건이 조건을 만족
 - SOME 과 동일

해답
A.MANAGER_ID = B.EMPLOYEE_ID 이므로
A 테이블의 메니저 아이디가 B 테이블의 직원 아이디이므로
A 가 직원, B 가 상사라고 볼수 있음

```
SELECT A.EMPLOYEE_ID AS "부하직원", A.LAST_NAME, A.MANAGER_ID "상사",
A.SALARY AS "부하연봉",
B.SALARY AS "상사연봉"
FROM HR.EMPLOYEES A, -- EMPLOYEE SALARY
 HR.EMPLOYEES B -- MANAGER SALARY
WHERE 1=1
AND A.MANAGER_ID = B.EMPLOYEE_ID
AND B.SALARY >= ANY A.SALARY;
```

이므로 상사의 연봉이 어떤 부하직원보다도 큰 결과를 뽑는 sql임

2. 아래의 쿼리의 결과를 만족하는 결과로 가장 알맞은 것은?
[기출 21회 - 다중 행 비교 연산자 IN, ALL, ANY, SOME, EXISTS]
DEPARTMENT_ID :
        NULL, 10,20,30,40,50,~90,100,110

```
SELECT
        DISTINCT DEPARTMENT_ID
FROM HR.EMPLOYEES A
WHERE A.DEPARTMENT_ID <= ALL (30,50);
```

1) 10,20
2) 10,20,30
3) 10,20,30,40
4) 10,20,30,40,50

사전 지식

* 비교연산자 ALL (다중 행 서브쿼리)
 - 서브쿼리의 결과에 존재하는 모든값을 만족하는 조건을 의미.
 - 비교연산자로 " 〉" 를 사용했다면 메인쿼리는 서브쿼리의 모든 값을
 - 만족해야 하므로 서브쿼리의 결과의 최대값보다 큰 모든 건이 조건을 만족

해답

A.DEPARTMENT_ID <= ALL(30,50) 이므로
30 보다 작고, 50 보다 작아야 함
즉, 30보다 작거나 같은 값이 모두 해당됨

3. 아래와 같은 테이블에 데이터가 있다. 각 SQL 에 대한 결과값이 잘못된 것은?
[기출 21회 - 다중 행 비교 연산자 IN, ALL, ANY, SOME, EXISTS]

```
TABLE SQLD_21_01          TABLE SQLD_21_02
N1        V1              N1        V1
--------- ----------      --------- ----------
1         A               1         A
2         2
3         B               3         B
4         C

1) SELECT * FROM SQLD_21_01
WHERE V1 IN (SELECT V1 FROM SQLD_21_02);

N1        V1
--------- ----------
1         A
3         B
2) SELECT * FROM SQLD_21_01
WHERE V1 NOT IN (SELECT V1 FROM SQLD_21_02);
N1        V1
--------- ----------
4         C

3) SELECT * FROM SQLD_21_01 A
WHERE EXISTS (SELECT 'X' FROM SQLD_21_02 B
WHERE A.V1 = B.V1);
N1        V1
--------- ----------
1         A
```

3 B 4) SELECT * FROM SQLD_21_01 A WHERE NOT EXISTS (SELECT 'X' FROM SQLD_21_02 B WHERE A.V1 = B.V1); N1 V1 --------- ---------- 2 4 C
사전 지식 NOT IN 의 경우 조건절이 아래와 같이 변형되어 수행됨 LNNVL(V1 <> :B1) -----> NULL 인 경우 Unkown으로 True Return. --> 조건절 만족, Not Exists 를 만족하지 않음 HASH AN 으로 풀릴 경우 A.V1 = B.V1 으로 ACCESS 됨 즉, NOT 조건이므로 NULL 연산이 Unknown으로 처리되어 True 로 반환되는 현상이 나타남
해답 SQLD_21_02 의 V1 에 NULL 이 존재하므로 NOT IN 경우 NULL 로 인해 모든 조건이 참이 되어 NOT INT 결과 0건이 RETURN 됨

3. 아래의 NOT EXISTS 구문을 동일한 결과를 출력하는 SQL 로 변경할 때 빈칸을 완성하시오 **[기출 21회차 - 다중 행 비교 연산자 IN, ALL, ANY, SOME, EXISTS]** SELECT ... FROM 급여이력 S WHERE NOT EXISTS (SELECT 'X' FROM 사원 P WHERE P.사원번호 = S.사원번호) SELECT .... FROM 급여이력 S LEFT OUTER JOIN 사원 P ON (S.사원번호 = P.사원번호) WHERE ( )
3번 해답 : P.사원번호 IS NULL NOT EXISTS 의 OUTER JOIN 으로의 변형을 묻는 문제로 NOT EXISTS 는 OUTER JOIN으로 변경 시 NOT NULL COLUMN 에 대한 IS NULL 체크로 NOT EXISTS를 구현 가능하다
유사 문제 SELECT ... FROM 급여이력 S WHERE NOT EXISTS (SELECT 'X' FROM 사원 P WHERE P.사원번호 = S.사원번호) SELECT .... FROM 급여이력 S (LEFT OUTER) JOIN 사원 P ON (S.사원번호 = P.사원번호)

WHERE P.사원번호 IS NULL

7. 아래의 SQL 의 출력 결과를 작성하시오. **[기출 21회차 - 다중 행 비교 연산자 IN, ALL, ANY, SOME, EXISTS]** TAB1 COL1 COL2 TAB2 COL1 COL2 Z 10 Y 1 Y 20 Y 2 X 30 Y 3 SELECT COUNT(*) FROM TAB1 WHERE EXISTS (SELECT 1 FROM TAB2 WHERE TAB2.COL1 = 'X'); **해답:** 0

12. 아래 VIEW 에 대한 설명 중 가장 올바르지 않은 것은? [기출 21회 View] 1) 독립성: 테이블 구조가 변경되어도 뷰를 사용하는 응용 프로그램은 변경하지 않아도 된다. 2) 편리성: 복잡한 질의를 뷰로 생성함으로써 관련 질의를 단순하게 작성할 수 있다. 또한, 해당 형태의 SQL문을 자주 사용할 때 뷰를 이용하면 편리하게 사용할 수 있다. 3) 물리성: 실제 데이터를 가지고 있어서 물리적인 관리가 가능하다 4) 보안성: 직원의 급여정보와 같이 숨기고 싶은 정보가 존재한다면 , 뷰를 생성할 때 해당 칼럼을 빼고 생성함으로써 사용자에게 정보를 감출 수 있다
해답 View 는 논리적인 SELECT 를 포함하는 OBEJCT 로 실제 데이터를 저장하지 않음

28. 서브쿼리에 대한 설명 중 가장 올바르지 않은 것은? [기출 21회차 - 서브쿼리] 1) 서브쿼리는 괄호로 감싸서 사용한다 2) 서브쿼리는 비교 연산자와 함께 사용 가능하다 3) 메인쿼리는 서브쿼리의 컬럼을 쓸 수 없다 4) 서브쿼리는 SELECT 절, FROM 절, WHERE 절 등에서 사용 가능하다
해답 서브쿼리, 특히 INLINE VIEW의 컬럼을 메인 쿼리에서도 사용 가능함

2. ERD 를 참고하여 아래의 SQL 중 오류가 날 수 있는 SQL 을 고르시오. [기출 30회차 - 서브쿼리] [ERD] TABLE A ----<- TABLE B -<------ TABLE C PK A A,B C
사전 지식 * 단일 행을 요구하는 SQL 에서 서브쿼리의 결과가 PK 중 일부 컬럼만 들어와서 다중 행이 나오는 서브쿼리는 오류가 남

해답: 서브쿼리에서 B 테이블의 A 컬럼만 조인조건으로 나오는 SQL 이 답임

2. 아래에서 설명하는 것은 무엇인가? [기출 30회차 – View]

[설명]

+ 복잡한 질의를 단순하게 작성할 수 있다.또한 해당 형태의 SQL문을 자주 사용할 때 이용하면 편리하게 사용할 수 있다.

+ 사용자에게 정보를 감출 수 있다

+ 실제 데이터를 가지고 있지 않다

해답 : VIEW

5) 그룹 함수(Group Function)

5. 아래와 같은 수행 결과가 나오도록 SQL 을 완성하시오
[기출 26회차 – 집계 함수 Rollup, Cube, Grouping sets]

```
SELECT CASE GROUPING (DNAME) WHEN 1 THEN 'All Departments' ELSE DNAME END AS DNAME,
 CASE GROUPING (JOB) WHEN 1 THEN 'All Jobs' ELSE JOB END AS JOB,
 COUNT(*) "Total Empl",
 SUM(SAL) "Total Sal"
FROM EMP, DEPT
WHERE DEPT.DEPTNO = EMP.DEPTNO
GROUP BY ( );
```

〈실행 결과〉

DNAME	JOB	Total Empl	Total Sal
All Departments	All Jobs	14	29025
All Departments	CLERK	4	4150
All Departments	ANALYST	2	6000
All Departments	MANAGER	3	8275
All Departments	SALESMAN	4	5600
All Departments	PRESIDENT	1	5000
SALES	All Jobs	6	9400
SALES	CLERK	1	950
SALES	MANAGER	1	2850
SALES	SALESMAN	4	5600
RESEARCH	All Jobs	5	10875
RESEARCH	CLERK	2	1900
RESEARCH	ANALYST	2	6000
RESEARCH	MANAGER	1	2975
ACCOUNTING	All Jobs	3	8750
ACCOUNTING	CLERK	1	1300
ACCOUNTING	MANAGER	1	2450
ACCOUNTING	PRESIDENT	1	5000

5번 해답 : CUBE (DNAME, JOB)

10. 아래 쿼리중 결과값이 다른 하나는?

[기출 21회 – GROUP 함수 ROLLUP, CUBE, GROUPING SETS
집합 연산자 UNION, UNION ALL, INTERSECT, EXCEPT]

1)
```
SELECT DNAME,JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
        SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
GROUP BY ROLLUP(DNAME,JOB)
ORDER BY DNAME,JOB;
```

2)
```
SELECT DNAME,JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
WHERE A.DEPTNO = B.DEPTNO
GROUP BY GROUPING SETS( (DNAME,JOB),DNAME,NULL)
ORDER BY DNAME,JOB;
```

3)
```
SELECT DNAME,JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
        SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
GROUP BY DNAME,JOB
UNION ALL
SELECT DNAME,'' AS JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
        SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
GROUP BY DNAME
UNION ALL
SELECT '' AS DNAME,'' AS JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
        SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
ORDER BY 1,2;
```

4)
```
SELECT DNAME,JOB,
        COUNT(*) "Total Empl",
        SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
        SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
GROUP BY CUBE(DNAME,JOB)
ORDER BY DNAME,JOB;
```

사전 지식

```
GROUP BY ROLLUP(DNAME,JOB)
 = GROUP BY DNAME,JOB
    UNION ALL
    GROUP BY DNAME
    UNION ALL
    모든 집합 그룹 결과
GROUP BY CUBE(DNAME,JOB)
 = GROUP BY DNAME,JOB
    UNION ALL
```

```
GROUP BY GROUPING SET(DNAME,JOB)
= GROUP BY DNAME
 UNION ALL
GROUP BY JOB
```

```
GROUP BY DNAME
 UNION ALL
 GROUP BY JOB
 UNION ALL
 모든 집합 그룹 결과
```

해답

```
GROUPING SETS(A,B,C) = GROUP BY A
                       UNION ALL
                       GROUP BY B
                       UNION ALL
                       GROUP BY C 이므로
```

4번만 다른 결과를 도출함

4. 아래 SQL 의 출력되는 ROWS 의 개수를 구하시오.
[기출 21회차 - 그룹함수 Rollup, Cube, Grouping sets]

EMP TABLE	DEPTNO	JOB	SAL
	20	CLERK	800
	30	SALESMAN	1600
	30	SALESMAN	1250
	20	MANAGER	2975
	30	SALESMAN	1250
	30	MANAGER	2850
	10	MANAGER	2450
	20	ANALYST	3000
	10	PRESIDENT	5000
	30	SALESMAN	1500
	20	CLERK	1100
	30	CLERK	950
	20	ANALYST	3000
	10	CLERK	1300

DEPT TABLE	DEPTNO	DNAME
	10	ACCOUNTING
	20	RESEARCH
	30	SALES
	40	OPERATIONS

```
SELECT DNAME,JOB,
                COUNT(*) "Total Emp",
                SUM(SAL) "Total Sal"
FROM SCOTT.EMP A,
                SCOTT.DEPT B
WHERE A.DEPTNO = B.DEPTNO
GROUP BY CUBE(DNAME,JOB)
```

사전 지식

** DNAME 의 그룹은 3, JOB 의 그룹은 5, DNAME,JOB 의 그룹은 9개가 형성됨

```
** GROUP BY CUBE(DNAME,JOB)
 = GROUP BY DNAME,JOB
   UNION ALL
   GROUP BY DNAME
   UNION ALL
   GROUP BY JOB
   UNION ALL
   모든 집합
```

4번 해답 : 18건

DNAME	JOB	Total Emp	Total Sal
		14	29025
	CLERK	4	4150
	ANALYST	2	6000
	MANAGER	3	8275
	SALESMAN	4	5600
	PRESIDENT	1	5000
SALES		6	9400
SALES	CLERK	1	950
SALES	MANAGER	1	2850
SALES	SALESMAN	4	5600
RESEARCH		5	10875
RESEARCH	CLERK	2	1900
RESEARCH	ANALYST	2	6000
RESEARCH	MANAGER	1	2975
ACCOUNTING		3	8750
ACCOUNTING	CLERK	1	1300
ACCOUNTING	MANAGER	1	2450
ACCOUNTING	PRESIDENT	1	5000

5. 아래와 같은 테이블이 있을 때 SQL 수행 결과와 동일한 결과를 얻기 위한 SQL 을 작성하시오.
[기출 25회차 – GROUP 함수 (집계함수) ROLLUP, CUBE, GROUPING SETS]

〈테이블 구조〉

TAB1	COL1	COL2	COL3
	한국	A01	1000
	한국	A01	2000
`	한국	A02	5000
	한국	A02	3000
	중국	B01	100
	중국	B01	200
	중국	B03	500
	미국	C01	10000
	미국	C01	20000
	미국	C03	5000

〈실행 결과〉

COL1	COL2	CNT	T_SUM
미국	C01	2	30000
미국	C03	1	5000
미국		3	35000
중국	B01	2	300
중국	B03	1	500
중국		3	800
한국	A01	2	3000
한국	A02	2	8000
한국		4	11000
		10	46800

〈수행 SQL〉

```
SELECT COL1, COL2,
  COUNT(*) AS CNT,
  SUM(연봉) AS T_SUM
FROM TAB1
GROUP BY ( )
ORDER BY 1,2;
```

5번 해답: ROLLUP (COL1, COL2)

3. 아래의 결과를 보고 빈칸에 들어가는 GROUP BY 뒤의 내용을 고르시오.
[기출 30회차 – 그룹 함수 (집계함수)]

[결과]

COL1	COL2	집계결과
…		
??		???
…		
NULL	NULL	???

사전 지식

* COL1 과 COL2 가 모두 NULL 인 전체 집계와 COL1 의 소계만 존재하므로 ROLLUP 임

해답: ROLLUP (COL1, COL2)

6) 윈도우 함수(Window Function)

25. 아래의 WINDOW FUNCTION 을 사용한 SQL 중 가장 올바르지 않은 것은?
[기출 21회차 – Window Function]

1) SUM(SAL) OVER()
2) SUM(SAL) OVER(PARTITION BY JOB ORDER BY EMPNO
   RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) SAL1
3) SUM(SAL) OVER(PARTITION BY JOB ORDER BY JOB
   RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) SAL2
4) SUM(SAL) OVER(PARTITION BY JOB ORDER BY EMPNO

RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED PRECEDING) SAL3

사전 지식

RANGE BETWEEN start_point AND end_point
- start_point는 end_point와 같거나 작은 값이 들어감
- Default값은 RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW
- UNBOUNDED PRECEDING : start_point만 들어갈 수 있으며, 파티션의 first row
- UNBOUNDED FOLLOWING : end_point만 들어갈 수 있으며, 파티션의 last row
- CURRENT ROW : start, end_point 둘다 가능. 윈도우는 CUREENT ROW에서 start하거나 end 함

해답

UNBOUNDED PRECEDING 은 end point 에 사용될 수 없음

3. 아래와 같은 테이블이 있을 때 수행 결과와 같은 SQL 을 작성하시오
[기출 24회차 - 윈도우 함수 Rank(), Dense_rank(), Row_number()]

〈테이블 구조〉

TAB1	COL1	COL2	COL3
	0001	001	1500
	0002	001	1500
`	0003	001	1000
	0004	001	500
	0005	002	1000
	0006	002	1000
	0007	003	500
	0007	004	1000

〈수행 SQL〉

```
SELECT *
FROM (
 SELECT COL2, ( ) OVER (PARTITION BY COL2 ORDER BY COL3 ) RNUM
 FROM TAB1
 )
WHERE MOD(RNUM,2)=1;
```

〈수행 결과〉

COL2	COL3	RNUM
001	500	1
001	1500	3
002	1000	1
003	500	1
004	1000	1

*** 동일 등수를 허용하지 않으며 등수에서 짝수등수는 제외함

3번 해답 : ROW_NUMBER

6. 테이블 구조에서 아래와 같은 누적 SUM 을 구하기 위한 SQL 을 완성하시오.
(COL3 기준으로 정렬된 누적평균임) [기출 25회차 - Window 함수]

TAB1	COL1	COL2	COL3

	한국	A01	1000
	한국	A01	2000
`	한국	A02	5000
	한국	A02	3000
	중국	B01	100
	중국	B01	200
	중국	B03	500
	미국	C01	10000
	미국	C01	20000
	미국	C03	5000

〈실행 결과〉

COL1	COL2	COL3	누적평균
미국	C03	5000	5000
미국	C01	10000	7500
미국	C01	20000	11667
중국	B01	100	100
중국	B01	200	150
중국	B03	500	267
한국	A01	1000	1000
한국	A01	2000	1500
한국	A02	3000	2000
한국	A02	5000	2750

〈수행 SQL〉

```
SELECT COL1, COL2,
        ROUND ( AVG (COL3) OVER(        ) , 0) AS 누적평균
FROM TAB1
ORDER BY COL1, COL3;
```

6번 해답:

PARTITION BY COL1 ORDER BY COL3
RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW

7. 아래와 같은 결과처럼 파티션별 윈도우에서 제일 먼저 나오는 것을 0으로, 제일 늦게 나오는 것을 1로 하여 값이 아닌 행의 순서별 백분율을 구하는 Window Funciton 은 무엇인가?
[기출 25회차 - Window 함수]
〈실행 결과〉

DEPTNO	ENAME	SAL	P_R
10	KING	5000	0
10	CLARK	2450	0.5
10	MILLER	1300	1
20	SCOTT	3000	0
20	FORD	3000	0
20	JONES	2975	0.5
20	ADAMS	1100	0.75
20	SMITH	800	1
30	BLAKE	2850	0
30	ALLEN	1600	0.2
30	TURNER	1500	0.4
30	MARTIN	1250	0.6
30	WARD	1250	0.6
30	JAMES	950	1

7번 해답 : PERCENT_RANK

6. 아래와 같은 수행 결과가 나오도록 SQL 을 완성하시오
[기출 26회차 - Window 함수]

```
SELECT 매출월,
 매출액,
 (  ) AS 누적매출
FROM 매출집계
WHERE 매출월 >= '201701';
```

<실행 결과>

매출월	매출액	누적매출
201701	1000	1000
201702	2000	3000
201703	3000	6000
201704	1000	7000
201705	400	7400
201706	5000	12400

6번 해답:
SUM(매출액) OVER (PARTITION BY 매출월 ORDER BY 매출월 RANGE BETWEEN UNOBOUNDED PRECED

7) DCL(Data Control Language)

29. 유저와 권한 중 권한에 대한 설명 중 가장 올바르지 않은 것은? [기출 21회차 - DCL]
1) 사용자가 실행하는 모든 DDL 문장은 그에 해당하는 적절한 권한이 있어야만 문장을 실행 할 수 있다.
2) DBA 권한을 가진 유저만이 권한을 부여 할 수 있다
3) 테이블의 소유자는 해당 테이블의 DML 권한을 다른 유저에게 부여 할 수 있다.
4) 권한 부여를 편리하게 관리하기 위해 만들어진 권한의 집합인 ROLE 이 있다

해답 DBA 권한은 SYSTEM,SYS 등의 상위 유저와 그에 해당하는 권한을 가진 경우 부여 가능

7. Object 의 권한이나 Role 을 부여할 때 사용하는 명령어는 GRANT 이다. 부여된 권한을 회수하는 명령어는? [기출 26회차 - DCL]
7번 해답 : REVOKE

14.SQL 명령어중 DCL 인 것은? [기출 30회차 – DCL] [보기] + UPDATE, GRANT, SELECT, COMMIT
해답 : GRANT

6. 테이블 생성을 위한 권한을 주기위해 아래의 SQL 을 완성하시오. [기출 30회차 – DCL] [SQL] ( ) CREATE TABLE TO USER01;
해답: GRANT

8) 절차형 SQL

10.TRIGGER 에 대한 설명중 부적절한 것은? (****) [기출 30회차 – 절차형 SQL(Trigger)]
사전 지식 *:NEW 는 NULL 을 가지므로 해당 보기가 답이 아닐 수 있음
해답: DELETE ON TRIGGER 의 경우 :OLD 는 삭제 전 데이터를, :NEW 는 삭제 후 데이터를 나타낸다

3 SQL 최적화 기본 원리

1) 옵티마이저와 실행계획

7. 학생관련 정보를 조회하는 SQL 을 작성할려고 한다. 조회하는 사람은 주로 학생 본인이 학번으로 조회를 주로 한다. 이런 SQL 일 때 성능을 개선하는 방법으로 가장 알맞은 것은? [기출 21회 - 규칙기반 옵티마이저] 1) 학교명을 선두컬럼으로 하는 INDEX 를 생성한다 2) 학번을 선두컬럼으로 하는 INDEX 를 생성한다 3) 학교명 + 학번순으로 구성된 INDEX 를 생성한다 4) 학교명 + 이름 + 학번으로 구성된 INDEX 를 생성한다
해답 학생 본인이 학번으로 주로 조회를 하기 때문에 학번이 선두 컬럼으로 있는 인덱스가 가장 적절함 학교명이 앞으로 올 경우 반드시 학교명을 적어야 하는 부담이 있음 해당 문제는 SQLD 보다는 SQLP 에 가까운 튜닝 문제임

15. 아래의 실행 계획을 올바르게 설명한 것은? [기출 21회 - 실행계획 (조인순서, 조인기업, 액세스 기법, 최적화정보, 연산)]

```
-----------------------------------------------------------
0         SELECT STATEMENT Optimizer=ALL_ROWS (Cost=7 Card=9 Bytes=1K)
1         0         HASH JOIN (Cost=7 Card=9 Bytes=1K)
2         1         TABLE ACCESS (FULL) OF 'SCOTT.DEPT' (TABLE) (Cost=3 Card=1 Bytes=30)
3         1         VIEW (Cost=3 Card=9 Bytes=783)
4         3         COUNT (STOPKEY)
5         4         TABLE ACCESS (FULL) OF 'SCOTT.EMP' (TABLE) (Cost=3 Card=14 Bytes=1K)
-----------------------------------------------------------
Predicate information (identified by operation id):
-----------------------------------------------------------
1 - access("A"."DEPTNO"="B"."DEPTNO")
2 - filter("B"."DNAME"='SALES')
4 - filter(ROWNUM<10)
-----------------------------------------------------------
```

1) EMP TABLE 에 대한 행제한 구문이 있다
2) EMP TABLE 과 DEPT TABLE 은 OUTER JOIN 으로 수행되고 있다
3) EMP TABLE 과 DEPT TABLE 에서 선행 테이블은 EMP TABLE 이다
4) DEPT TABLE 은 별도의 조건이 없어 FULL SCAN 을 하고 있다

해답
Outer Join 이라고 표시된 SQL Operation 이 존재하지 않음
Hash Join 의 선행 테이블은 SCOTT.DEPT 테이블임
Predicate 정보를 보면 DEPT 테이블에 대한 Full Scan 후 DNAME ='SALES' 로 필터됨

4. 쿼리를 수행하는데 소요되는 일량 또는 시간을 기반으로 최적화를 수행하는 옵티마이저는 무엇인지 작성하시오. [기출 25회차 - 옵티마이저]
( ) 옵티마이저

4번 해답 : 비용 기반

6. 아래의 실행계획의 순서를 바로 적은 것은? [기출 30회차 - 실행계획]

[보기]

```
0- SELECT ..
1-   NESTED LOOP JOIN
2-     NESTED LOOP JOIN
3-       TABLE ACCESS (FULL)
4-       TABLE ACCESS (BY INDEX ROWID)
5-         INDEX (RANGE SCAN)
6-     TABLE ACCESS (BY INDEX ROWID)
7-       INDEX (RANGE SCAN)
```

해답: 3-5-4-2-7-6-1-0

2) 인덱스 기본

17. 아래와 같은 컬럼으로 구성된 테이블에 COL1을 구성 컬럼으로 가지는 인덱스가 있다.

가장 효율적으로 해당 인덱스를 사용할 수 있는 조건절은? [기출 21회차] TAB_A COL1 NUMBER INDEX COL1 COL2 VARCHAR2(10) 1) WHERE COL1 LIKE '2%' 2) WHERE COL1 = 10 3) WHERE COL1 IS NOT NULL 4) WHERE COL1 <> 10
사전 지식 * 문자를 숫자형으로 형변환함 * 예외적으로 LIKE 의 경우 COL 을 무조건 형변환함. 숫자를 문자로 변환함
해답 LIKE 의 경우 컬럼을 무조건 문자로 형변환을 함. 즉, 해당 조건절은 WHERE TO_CHAR(COL1) LIKE '2%' 로 변형되어 인덱스를 사용하지 못함 IS NOT NULL 은 해당 인덱스를 FULL SCAN 할 수 있으나 효율이 떨어짐 부정형 비교는 인덱스 사용이 불가함

3) 조인 수행 원리

14. 조인에 대한 설명 중 Hash Join 에 대한 특성으로 부적절한 것 2개를 고르시오 [기출 21회 – 조인 수행 원리 NL Join, Hash Join, Sort Merge Join] 1) 각 테이블에 INDEX 가 반드시 필요한 것은 아니다 2) 일반적으로 작은 테이블을 MEMORY 에 올리는 선행 테이블로 사용한다 3) Non Equal Join 이 가능하다 (비동등) 4) 조인을 위해 사전 소트 작업이 필요하다
해답 Hash Join 은 Non Equal Join 은 불가능함. Equal Join 만 가능함 사전 Sorting 작업이 필요한 Join 알고리즘은 Sort Merge Join 임

8. 프로그래밍에서 사용되는 중첩된 반복문과 유사한 방식으로 조인을 수행하는 방식을 무엇이라 하는가? [기출 25회차 – 조인 기본원리 Nested Loop Join, NL Join, Sot Merge Join, Hash Join, Scalar SubQuery] ( )
8번 해답 : NESTED LOOP JOIN

8. 아래와 같은 상황에서 사용되는 SQL Join Mechanism 을 작성하시오 [기출 26회차 – 조인 수행원리 NL Join, Sort Merge Join, Hash Join] 조건 1: TAB1 과 TAB2 를 조인하여 결과를 가져와야 한다. 조건 2: 두개 테이블에는 인덱스가 없다. 조건 3: Equal Join 에서만 해당 조인 방식을 사용할 수 있다.
8번 해답 : HASH JOIN Non Equal Join 이 되지 못하는 Join 은 Hash Join 이며 대량의 데이터를 처리할 때 주로 사용됨

19.SQL 조인 방식 아래의 설명을 만족하는 조인은? [기출 30회차 - 조인수행원리] [설명] + 대용량 데이터를 정렬하여 조인한다 + 동등 조인, 비동등 조인에서 다 사용 가능하다 + 기타 등등
해답 : Sort Merge JOIN