

## 4. SQL 활용 - 2

1. 다음 중 뷰(VIEW)에 대한 설명으로 가장 부적절한 것은?

- ① 뷰는 단지 정의만을 가지고 있으며, 실행 시점에 질의를 재작성하여 수행
- ② 뷰는 복잡한 SQL 문장을 단순화 시켜주는 장점이 있는 반면, 테이블 구조가 변경되면 응용 프로그램을 변경해 주어야 한다.
- ③ 뷰는 보안을 강화하기 위한 목적으로도 활용
- ④ 실제 데이터를 저장하고 있는 뷰를 생성하는 기능을 지원하는 DBMS도 존재

2. 아래 테이블에 대한 [뷰 생성 스크립트]를 실행한 후, 조회 SQL의 실행결과로 맞는 것은?

[TBL]

C1	C2
A	100
B	200
B	100
B	
	200

[뷰 생성 스크립트]

```
CREATE VIEW V_TBL
```

```
AS
```

```
SELECT *
```

```
FROM TBL
```

```
WHERE C1='B' OR C1 IS NULL;
```

[조회 SQL]

```
SELECT SUM(C2) C2
```

```
FROM V_TBL
```

```
WHERE C2>=200 AND C1='B';
```

3. 아래 결과를 얻기 위한 SQL문에서 괄호에 들어갈 함수를 작성하시오.

[구매이력]

구매고객	구매월	총 구매건	총 구매액
AAA	201001	1	1000
AAA	201002	2	3000
AAA	201003	1	1000
AAA		4	5000
BBB	201001	3	2000
BBB	201002	5	3000
BBB	201003	1	2000
BBB		9	7000
CCC	201101	1	2000
CCC	201102	1	5000
CCC	201103	1	1000
CCC		3	8000
		16	20000

[SQL문]

```
SELECT 구매고객, 구매월, COUNT(*) "총 구매건", SUM(구매금액) "총 구매액"
```

```
FROM 구매이력
```

```
GROUP BY ( ) (구매고객, 구매월);
```

4. 아래와 같이 설비와 에너지사용 테이블을 이용하여 결과를 나타낼 수 있는 SQL문을 작성하시오.

[설비]

설비ID	설비명
1	설비1
2	설비2
3	설비3

[에너지사용]

설비ID	에너지코드	사용량
1	전기	100
1	용수	200
1	바람	300
2	전기	200
2	용수	300
3	전기	300

[결과]

설비ID	에너지코드	사용량합계
1	바람	300
1	용수	200
1	전기	100
1	NULL	600
2	용수	300
2	전기	200
2	NULL	500
3	전기	300
3	NULL	300
NULL	바람	300
NULL	용수	500
NULL	전기	600
NULL	NULL	1400

5. 월별매출 테이블을 대상으로 수행한 결과는?

[월별매출]

상품ID	월	매출액
P001	2014.10	1500
P001	2014.11	1500
P001	2014.12	2500
P002	2014.10	1000
P002	2014.11	2000
P002	2014.12	1500
P003	2014.10	2000
P003	2014.11	1000
P003	2014.12	1000

[SQL]

```
SELECT 상품ID, 월, SUM(매출액) AS 매출액
FROM 월별매출
WHERE 월 BETWEEN '2014.10' AND '2014.12'
GROUP BY GROUPING SETS(상품ID, 월);
```

6. 다음 중 윈도우 함수에 대한 설명으로 가장 부적절한 것은?

- ① PARTITION과 GROUP BY 구문은 의미적으로 유사하다.
- ② PARTITION 구문이 없으면 전체 집합을 하나의 PARTITION으로 정의한 것과 동일하다.
- ③ 윈도우 함수 처리로 인해 결과 건수가 줄어든다.
- ④ 윈도우 함수 적용 범위는 PARTITION을 넘을 수 없다.

7. 추천내역 테이블에서 아래와 같이 수행한 결과는?

[추천내역]

추천경로	추천인	피추천인	추천점수
SNS	나한일	강감찬	75
SNS	이순신	강감찬	80
이벤트응모	홍길동	강감찬	88
이벤트응모	저절로	이순신	78
홈페이지	저절로	이대로	93
홈페이지	홍두깨	심청이	98

[SQL]

```
SELECT 추천경로, 추천인, 피추천인, 추천점수
FROM (SELECT 추천경로, 추천인, 피추천인, 추천점수, ROW_NUMBER( )
      OVER (PARTITION BY 추천경로 ORDER BY 추천점수 DESC) AS RNUM
      FROM 추천내역)
WHERE RNUM=1;
```

9. 사원 테이블에서 아래와 같이 수행한 결과는?

[사원]

사원ID	부서ID	사원명	연봉
001	100	홍길동	2500
002	100	강감찬	3000
003	200	김유신	4500
004	200	김선달	3000
005	200	유학생	2500
006	300	변사포	4500
007	300	박문수	3000

[SQL]

```
SELECT Y.사원ID, Y.부서ID, Y.사원명, Y.연봉
FROM (SELECT 사원ID, MAX(연봉) OVER(PARTITION BY 부서ID) AS 최고연봉
      FROM 사원) X, 사원 Y
WHERE X.사원ID = Y.사원ID
AND X.최고연봉=Y.연봉;
```

10. 테이블 및 인덱스 등의 통계 정보를 활용하여 SQL문을 실행하는데 소요 될 처리시간 및 CPU, I/O 자원량 등을 계산하여 가장 효율적일 것으로 예상되는 실행계획을 선택하는 옵티마이저를 ( )라 한다.

11. 아래 실행계획의 실행순서에 맞게 빈칸을 채우시오.

1. NESTED LOOPS
2. HASH JOIN
3. TABLE ACCESS (FULL) TAB1
4. TABLE ACCESS (FULL) TAB2
5. TABLE ACCESS (BY ROWID) TAB3
6. INDEX (UNIQUE SCAN) PK\_TAB3

( ) -> ( ) -> ( ) -> 6 -> 5 -> 1

12. 다음 중 실행계획에 대한 설명으로 가장 부적절한 것은?

- ① 실행계획은 SQL 처리를 위한 실행절차와 방법을 표현한 것이다.
- ② 실행계획은 조인방법, 조인순서, 액세스 기법 등이 표현된다.
- ③ 동일 SQL문에 대해 실행계획이 다르면 실행 결과도 달라질 수 있다.
- ④ 비용기반 옵티마이저의 실행계획에는 단계별 예상 비용 및 건수 등이 표시 된다.

13. 관계형 데이터베이스의 인덱스에 대한 설명으로 가장 적절한 것은?

- ① 기본 인덱스(Primary Key Index)에 중복된 키 값들이 나타날 수 있다.
- ② 기본 인덱스에 널 값들이 나타날 수 있다.
- ③ 보조 인덱스에는 고유한 키 값들만 나타날 수 있다.
- ④ 자주 변경되는 속성은 인덱스를 정의할 좋은 후보이다.

14. 관계형 데이터베이스 인덱스에 대한 설명으로 가장 부적절한 것을 2개 고르시오.

- ① 테이블의 전체 데이터를 읽는 경우는 인덱스가 거의 불필요하다.
- ② 인덱스는 조회, 삽입, 삭제, 갱신 연산의 속도를 향상시킨다.
- ③ B 트리는 관계형 데이터베이스의 주요 인덱스 구조이다.
- ④ 대량의 데이터를 삽입할 때는 모든 인덱스를 생성하고 데이터를 입력하는 것이 좋다.

15. 옵티마이저와 실행계획에 대한 설명 중 옳은 것을 모두 고르시오.

- ① ORACLE의 규칙기반 옵티마이저에서 가장 우선순위가 높은 규칙은 SINGLE ROW BY ROWID 액세스 기법이다.
- ② 비용기반 옵티마이저는 테이블, 인덱스, 칼럼 등 객체의 통계정보를 사용하여 실행계획을 수립하므로 통계정보가 변경되면 SQL의 실행계획이 달라질 수 있다.
- ③ ORACLE의 실행계획에 나타나는 기본적인 JOIN기법으로는 NL JOIN, HASH JOIN, SORT MERGE JOIN 등이 있다.
- ④ 다양한 JOIN 기법 중 NL JOIN은 DW(DATA WAREHOUSE, 분석용 데이터베이스) 등에서 데이터를 집계하는 업무에 많이 사용된다.

16. NESTED LOOP JOIN에 대한 설명으로 가장 부적절한 것은?

- ① 조인 칼럼에 적당한 인덱스가 있어서 자연조인(NATURAL JOIN)이 효율적일 때 유용하다.
- ② DRIVING TABLE의 조인 데이터양이 큰 영향을 주는 조인 방식이다.
- ③ SORT MERGE JOIN 하기에 두 테이블이 너무 커서 SORT 부하가 심할 때 유용하다.
- ④ 유니크 인덱스를 활용하여 수행시간이 적게 걸리는 소량 테이블을 온라인 조회하는 경우 유용하다.

17. SORT MERGE JOIN에 대한 설명으로 가장 부적절한 것은?

- ① 조인 칼럼에 적당한 인덱스가 없어서 NL 조인이 비효율적일 때 사용
- ② DRIVING TABLE의 개념이 중요하지 않은 조인 방식
- ③ 조인 조건의 인덱스의 유무에 영향 받지 않는다.
- ④ EQUI(=) 조인 조건에서만 동작

18. JOIN 기법에 대한 설명으로 가장 적절한 것은?

- ① NL JOIN은 선택도가 낮은(결과 행의 수가 적은) 테이블이 선행 테이블로 선택되는 것이 일반적으로 유리
- ② SORT MERGE JOIN은 EQUI JOIN에서만 사용할 수 있으므로 제약이 존재한다.
- ③ HASH JOIN은 결과 행의 수가 큰 테이블을 선행 테이블로 사용하는 것이 성능에 유리하다.
- ④ HASH JOIN은 SORT MERGE JOIN보다 항상 우수한 성능을 보장한다.