



# [AI초급] SQLD 자격증 코스 - 챕터 15



매 주차 강의자료 시작에 PDF파일을 올려두었어요!

▼ PDF 파일

## [수업 목표]

- SQLD 문제 풀이

## [목차]

- [01. 데이터 모델링의 이해](#)
- [02. SQL 기본 및 활용 1](#)
- [03. SQL 기본 및 활용 2](#)
- [04. SQL 기본 및 활용 3](#)
- [05. SQL 기본 및 활용 4](#)



모든 토글을 열고 닫는 단축키

Windows : **Ctrl** + **alt** + **t**

Mac : **⌘** + **~** + **t**

## 01. 데이터 모델링의 이해

▼ 문제 및 정답

### 01번



발생 시점에 따른 엔터티의 종류가 아닌 것은?

1. 기본 엔터티
2. 사건 엔터티
3. 행위 엔터티
4. 중심 엔터티

▼ 정

(2)

발생 시점에 따른 엔터티는 기본/키 엔터티, 중심 엔터티, 행위 엔터티가 있으며

유/무형에 따른 엔터티는 유형 엔터티, 개념 엔터티, 사건 엔터티가 있다

### 02번\*



도메인의 특징으로 알맞은 것은?

1. 속성명과 도메인명은 반드시 동일해야 한다
2. 릴레이션의 속성에 대한 제약사항 및 데이터 타입과 크기이다
3. 각 속성은 도메인 이외의 값을 가질 수 있다
4. 각 엔터티 속성에 도메인을 할당할 필요는 없다

▼ 정답

(2)

*도메인이란?*

- 각 속성이 가질 수 있는 값의 범위를 의미
- 엔터티 내에서 속성에 대한 데이터 타입과 크기, 제약 사항 등을 지정한다
- 예를 들어, 성별이라는 속성의 도메인은 남자와 여자가 있다

*릴레이션이란?*

- 릴레이션과 테이블은 같은 의미로 해석하면 된다

1번 : 속성명과 도메인명은 반드시 동일하지 않아도 된다

3번 : 각 속성은 도메인 이외의 값을 가질 수 없다

4번 : 각 엔터티 속성에 도메인을 할당한다

## 03번\*



아래 ERD데이터 모델링 표기법으로 옳은 것은?

주문

주문번호

고객번호

주문금액

주문일자

상품번호

1. External 표기법
2. Barker 표기법
3. Agile 모델
4. IE 표기법

▼ 정답

(4)

대표적인 데이터 모델 표기법은 IE/Crow's Foot 표기법과 Barker/Case\*Method 표기법이 있다

## 04번



아래 테이블에서 외래키로 볼 수 있는 컬럼은 무엇인가?

| 작가         | 도서         |
|------------|------------|
| 작가 ID (PK) | 도서 번호 (PK) |
| 작가 본명      | 도서명        |
| 작가 활동명     | 작가 ID      |
| 도서 출간 횟수   | 출판사        |

1. 작가 본명
2. 도서명
3. 작가 ID
4. 도서 번호

▼ 정답

(3)

도서 테이블은 작가 ID를 통해 작가 테이블과의 관계를 가지고 있다.

## 05번



아래 설명 중 엔터티라고 할 수 있는 것은?

훈련병 친구의 자대배치를 조회하기 위해서는 훈련병의 정보를 입력해야 한다.

이름/생년월일/입대일자/부대를 입력하면 자대배치를 조회할 수 있다

1. 훈련병
2. 이름
3. 생년월일
4. 입대일자

▼ 정답

(1)

엔터티는 2개 이상의 인스턴스로 구성되어 있으며 집합의 특성을 가지고 있다. 이름,생년월일,입대일자는 속성에 해당한다

## 06번



성능 데이터 모델링에 대한 설명으로 옳지 않은 것은?

1. 분석 및 설계 단계에서 데이터 모델에 성능을 고려한 데이터 모델링을 수행할 경우 성능 저하로 인해 발생하는 재업무(Rework) 비용을 최소화할 수 있다
2. 데이터 모델링 때부터 정규화, 반정규화, 테이블 통합 등 여러 가지 성능과 관련된 사항이 데이터 모델링에 반영될 수 있도록 하는 것으로 정의한다
3. 프로젝트 수행 전, 철저하게 설계된 상태로 도입할수록 비용이 적게 든다
4. 성능 향상을 저해하는 요소는 하나로 좁힐 수 있으며 이를 해결하는 것이 성능 데이터 모델링이다

▼ 정답

(4)

성능 향상을 저해하는 요소는 매우 다양하며 처음부터 성능 향상을 목적으로 진행하는 것이 성능 데이터 모델링이다.

## 07번\*



아래 데이터는 정규직 직원 테이블이다. 테이블에 대한 이상현상을 설명한 것 중 옳지 않은 것은?

| 직원번호     | 직원명 | 부서번호 | 부서명    |
|----------|-----|------|--------|
| 20230830 | 신미미 | 001  | 데이터분석팀 |
| 20230812 | 주우재 | 002  | 개발팀    |
| 20195872 | 정리사 | 002  | 개발팀    |
| 20184720 | 배수지 | 003  | 운영팀    |

1. 삽입 이상 : 정규직 전환이 되지 않은 인턴 직원의 경우 위 테이블에 데이터를 삽입할 수 없다
2. 삭제 이상 : 신미미 직원의 정보가 삭제되면 데이터분석팀 정보도 삭제된다
3. 갱신 이상 : 개발팀 정보가 변경될 경우 주우재 직원의 정보만 변경하면 된다
4. 갱신 이상 : 운영팀 정보가 변경될 경우 배수지 직원의 운영팀 정보만 변경하면 된다

▼ 정답

(3)

개발팀 정보가 변경될 경우 주우재 직원, 정리사 직원의 정보를 변경해야 한다

## 08번



아래 설명은 주식별자의 어떤 특징을 설명한 것인가?

직원 번호만으로도 고유한 구조인데 직원 분류 코드와 직원번호가 조합된 형태로 식별자가 구성될 경우 부적절한 주식별자 구조가 만들어지게 된다

1. 유일성
2. 최소성
3. 불변성
4. 존재성

▼ 정답

(2)

최소성 : 주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수가 되어야합니다

## 09번\*



아래 엔터티는 몇 차 정규화를 위반한 것인가?

#### 직원

|           |
|-----------|
| 직원 번호(PK) |
| 직원명       |
| 부서번호      |
| 부서명       |

1. 1차 정규화
2. 2차 정규화
3. 3차 정규화
4. 위반하지 않고 모든 정규화 유형을 만족하고 있다

▼ 정답

(3)

3차 정규화는 2차 정규화를 만족시키고 일반 속성 간에도 함수 종속 관계가 존재하지 않아야 한다. 하지만 위 직원 엔터티는 부서 번호에 부서명 속성이 종속되는 관계임을 알 수 있다.

## 10번



분산 데이터베이스에 대한 설명으로 옳지 않은 것은?

1. 테이블 통합 요약 분산 방식은 분산되어 있는 다른 내용의 데이터를 이용하여 통합된 데이터를 산출하는 방식이다
2. 중복투명성은 DB 객체가 여러 사이트에 중복되어 있는지 알 필요가 없다
3. 분산 데이터베이스는 DBMS와 물리적 DB 사이의 연결됨(mapping)을 보장한다
4. 하나의 데이터베이스를 여러 가상 시스템으로 사용할 수 있도록 한다

▼ 정답

(4)

분산 데이터베이스는 여러 곳으로 분산되어 있는 데이터베이스를 하나의 가상 시스템으로 사용할 수 있도록 한다

## 02. SQL 기본 및 활용 1

▼ 문제 및 정답

### 01번\*



SQL 데이터 언어 분류로 옳지 않은 것은?

1. DML : SELECT
2. DML : ALTER
3. DCL : REVOKE
4. TCL : ROLLBACK

▼ 정답

(2)

테이블 변경(ALTER, RENAME)은 데이터 정의 언어 DDL로 분류한다

## 02번



데이터를 삽입하기 위해 사용하는 SQL 명령어는?

1. UPDATE
2. CREATE
3. SELECT
4. INSERT

▼ 정답

(4)

데이터 조작 언어에는 데이터 삽입(INSERT), 조회(SELECT), 수정(UPDATE), 삭제(DELETE)가 있다

## 03번\*



다음 중 칼럼명으로 사용할 수 있는 것은?

1. !dept
2. Soldier-unit
3. PARTICIPANT\_2
4. 2024EMP

▼ 정답

(3)

테이블명과 칼럼명은 반드시 문자로 시작해야하며 A-Z,a-z,0-9,\_,,\$,#만 사용이 가능하다

## 04번



제약 조건의 설명으로 옳지 않은 것은?

1. Primary Key는 테이블당 2개 이하의 제약을 정의한다
2. NOT NULL 은 NULL 값을 가지지 않도록 설정하는 제약이다
3. CHECK는 입력 값 범위를 제한하기 위한 제약 조건이다
4. Foreign Key는 NULL 값을 가질 수 있으며 여러 속성이 가능하다

▼ 정답

(1)

기본 키 (Primary Key)는 테이블 당 하나의 제약만 정의한다

## 05번\*



아래 테이블에서 데이터만 삭제하고 테이블 구조는 유지한 상태로 만들기 위한 SQL문은?

```
CREATE TABLE Customers (  
  customer_id NUMBER PRIMARY KEY,  
  first_name VARCHAR2(50),  
  email VARCHAR2(100)  
);
```

1. DELETE Customers FROM \* ;
2. DROP TABLE Customers;
3. TRUNCATE TABLE Customers;
4. CLEAR TABLE Customers;

▼ 정답

(3)

- TRUNCATE TABLE : 테이블의 모든 데이터를 삭제하고 테이블 구조는 유지한다
- DROP TABLE : 테이블 전체를 삭제할 때 사용하며 저장된 데이터도 같이 삭제된다

## 06번



테이블에 대한 설명으로 옳지 않은 것은?

1. 관계형 데이터베이스는 데이터베이스의 기본 단위인 테이블의 형태로 데이터를 저장한다
2. 세로 방향을 칼럼(Column), 가로 방향을 행(Row)이라 부른다
3. 2차원 관계형 테이블을 통한 데이터는 완전한 형태로 저장되었다고 할 수 있다
4. 테이블은 행과 열로 구성된 2차원의 데이터 저장 구조를 가지고 있다

▼ 정답

(3)

2차원 관계형 테이블을 통해서 어떠한 데이터를 저장한다고 하더라도 그 데이터가 완전한 형태로 저장되었다고 확신할 수는 없다

## 07번\*



아래 내용 중 ( ? )에 들어갈 알맞은 것은?

사용자가 데이터베이스에 접근하여 테이블을 만들려면 CREATE 권한이 있어야 한다. 접근 권한이 필요한 사용자에게 권한을 부여하기 위해서는 ( ? )을 사용한다.

1. GRANT
2. REVOKE
3. COMMIT
4. CONTROL

▼ 정답

(1)

데이터베이스 접근하고 객체를 사용하도록 권한을 부여(GRANT)하거나 회수(REVOKE)할 수 있다

## 08번



데이터베이스 접근 권한에 대한 설명으로 옳은 것은?

1. 유저 생성은 누구나 할 수 있다
2. 생성한 유저는 바로 로그인하여 데이터에 접근할 수 있다
3. 자신이 부여받은 권한에 대해서 다른 계정의 사용자에게 권한을 부여할 수 있다.
4. 다른 유저의 테이블에 대한 수정 권한은 없지만 조회는 가능하다

▼ 정답

(3)

유저 생성 및 로그인 후 데이터 접근은 권한이 있어야 가능하며, 다른 유저의 테이블에 대한 권한은 각 테이블의 소유자가 가지고 있기 때문에 권한을 받지 않으면 다른 유저의 테이블을 조회하는 것도 불가능하다

## 09번



사원 테이블에서 사원명과 부서명은 내림차순, 급여는 오름차순으로 정렬한 결과값으로 알맞는 SQL문은?

1. SELECT\*FROM 사원  
ORDER BY 사원명, 부서명 DESC, 급여 ;
2. SELECT\*FROM 사원  
ORDER BY 사원명 DESC, 부서명 DESC, 급여 ;
3. SELECT\*FROM 사원  
ORDER BY 사원명 ASC, 부서명 ASC, 급여 DESC;
4. SELECT\*FROM 사원  
ORDER BY 사원명, 부서명 ASC, 3 DESC;

▼ 정답

(2)

ORDER BY 절에 기재한 칼럼 뒤에 정렬 방식을 설정할 수 있으며 아무것도 적지 않으면 기본 값은 오름차순(ASC)으로 설정된다

## 10번\*



SELECT UPPER('lovesql') FROM DUAL;의 결과를 기술하라

답 : \_\_\_\_\_

▼ 정답

LOVESQLD

UPPER는 문자열을 대문자로 변환하는 연산자이다

## 03. SQL 기본 및 활용 2



▼ 문제 및 정답

01번



대량 데이터에 따른 성능으로 옳지 않은 것은?

1. 대량 데이터로 불필요한 블록의 입/출력이 많아지게 될 경우 수직 분할을 통해 해결이 가능하다
2. 수평 분할은 파티셔닝이라고 하며 특정 기간을 중심으로 분할하는 것을 범위 파티셔닝이라고 한다
3. 범위 분할은 보관 주기에 따른 테이블 관리가 용이하다
4. 해시 분할은 특정 파티션에 데이터가 집중될 가능성이 있는 리스트 분할의 단점을 보완할 수 있다

▼ 정답

(4)

해시 분할은 특정 파티션에 데이터가 집중될 가능성이 있는 ‘범위 분할’의 단점을 보완할 수 있다

02번\*



아래 SQL의 수행 결과를 기술하라

고객 정보

| 고객번호  | 고객명 |
|-------|-----|
| 10482 | 김민지 |
| 10528 | 이제훈 |
| 10937 | 신우리 |
| 10123 | 정수지 |

주문 내역

| 주문번호   | 고객번호  | 결제금액  |
|--------|-------|-------|
| 203491 | 10528 | 18000 |
| 203142 | 10482 | 20000 |
| 205926 | 10482 | 6700  |
| 583610 | 10482 | 16000 |
| 473827 | 10528 | 5000  |
| 202471 | 10937 | 9000  |
| 202516 | 10937 | 13000 |

SQL

```
SELECT SUM (O.결제금액) / COUNT (DISTINCT U.고객번호)
FROM 고객정보 U, 주문내역 O
WHERE O.고객번호 = U.고객번호 AND O.결제금액 > 15000;
```

답 : \_\_\_\_\_

▼ 정답

(27000)

SUM (P.결제금액) 값은 54,000원이고 COUNT (DISTINCT U.고객번호) 값은 2이다

03번\*



아래 SQL의 ( ? )에 들어갈 표현식으로 옳은 것은?

#### P1 테이블

| A1 | A2         | A3 |
|----|------------|----|
| A  | 2040-08-12 | 1  |
| A  | 2040-04-28 | 1  |
| B  | 2040-08-12 | 1  |
| B  | 2040-04-28 | 1  |
| C  | 2040-08-12 | 1  |
| C  | 2040-04-28 | 1  |

#### 결과

| A1 | A2         | A3 |
|----|------------|----|
| A  | 2040-08-12 | 1  |
| B  | 2040-08-12 | 1  |
| C  | 2040-08-12 | 1  |
|    | 2040-08-12 | 3  |
| A  | 2040-04-28 | 1  |
| B  | 2040-04-28 | 1  |
| C  | 2040-04-28 | 1  |
|    | 2040-04-28 | 3  |
|    |            | 6  |

#### SQL

```
SELECT A1, A2, SUM (A3) AS A3
FROM P1
GROUP BY ROLLUP ( ? );
```

1. A2, A1
2. A1, A2
3. A1
4. A2

▼ 정답

(1)

중간중간에 존재하는 NULL은 ROLLUP에 의해 만들어진 값이다.

A1, A2를 기준으로 GROUPING한 결과와 A2를 기준으로 나온 NULL값의 결과 그리고 마지막은 전체로 GROUPING한 결과이다.

ROLLUP은 순서가 중요하기 때문에 3가지 값이 모두 나오려면 A2, A1이 정답으로 들어가야 한다.

## 04번



서브쿼리 사용 시 주의사항으로 옳지 않은 것은?

1. 서브쿼리는 소괄호 ( ) 로 감싸서 사용한다
2. 서브쿼리에 ORDER BY절을 사용하여 정렬할 수 있다
3. 단일 행 비교 연산자는 서브쿼리의 결과가 반드시 1건 이하여야한다
4. 복수 행 비교 연산자는 서브쿼리의 결과 건수와 관련이없다

▼ 정답

(2)

ORDER BY 절은 SELECT 절에서 오직 한 개만 올 수 있기 때문에 메인쿼리의 마지막 문장에 위치해야 한다

## 05번\*



트리거에 대한 설명으로 옳지 않은 것은?

1. 트리거는 특정 데이터의 INSERT, UPDATE 등의 특정 이벤트 발생 시 자동 수행되는 절차형 SQL로 정의한다
2. 트리거의 실행은 외부적인 실행 명령에 의해 이루어진다
3. 트리거는 COMMIT, ROLLBACK 실행이 불가능하다
4. 트리거에 오류 존재 시, 처리 데이터에도 영향을 미치므로 주의가 필요하다

▼ 정답

(2)

트리거의 실행은 트리거링 사건(Triggering Event)에 의해 내부적으로 이루어진다

## 06번\*



아래 SQL의 ( ? )에 들어갈 분석 함수로 옳은 것은?

### SQL

```
SELECT ENAME, SAL, ( ? ) () OVER (ORDER BY SAL) AS Ranking
FROM EMP
WHERE DEPTNO = 10;
```

### 결과

| ENAME | SAL  | Ranking |
|-------|------|---------|
| Jake  | 2700 | 1       |
| Green | 2900 | 2       |
| Lisa  | 2900 | 2       |
| Soo   | 3050 | 4       |
| Jae   | 3100 | 5       |
| Julia | 4000 | 6       |

1. ROW\_NUMBER
2. PERCENT\_RANK
3. DENSE\_RANK
4. RANK

▼ 정답

(4)

- ROW\_NUMBER 결과값 : 1 > 2 > 3 > 4 > 5 > 6
- DENSE\_RANK 결과값 : 1 > 2 > 2 > 3 > 4 > 5

## 07번



PL/SQL에 대한 설명으로 옳지 않은 것은?

1. PL/SQL은 일반 프로그래밍 언어처럼 변수를 할당하고 예외 처리를 할 수 있다
2. PL/SQL은 SQL과 달리 문장의 끝에 반드시 세미콜론(;)을 작성하지 않아도 된다

- 3. PL/SQL로 작성된 코드는 PL/SQL엔진이 처리하고 일반적인 SQL문장은 SQL 실행기가 처리한다
- 4. DML 문장은 직접적으로 쿼리를 작성하여 사용할 수 있지만 DDL, DCL, TCL은 지원하지 않는다

▼ 정답

(2)

문장의 끝에 반드시 세미콜론(;)을 찍어야 하며 없는 경우 오류가 발생하게 되므로 주의해야 한다

08번\*



아래 SQL의 실행 결과로 옳은 것은?

S1 테이블

| A1   | A2   |
|------|------|
| 100  | NULL |
| NULL | 200  |
| 300  | 400  |

SQL

```
SELECT (A1/100) * (A2/100) AS AD
FROM S1
```

1.

|    |
|----|
| AD |
| 1  |
| 2  |
| 12 |
2.

|      |
|------|
| AD   |
| NULL |
3.

|    |
|----|
| AD |
| 12 |
4.

|      |
|------|
| AD   |
| NULL |
| NULL |
| 12   |

▼ 정답

(4)

- 1 \* NULL = NULL
- NULL \* 2 = NULL
- 3 \* 4 = 12

09번



뷰 사용의 장점으로 잘못된 것은?

1. 독립성
2. 편리성
3. 보안성
4. 복잡성

▼ 정답

(4)

복잡한 질의를 뷰로 생성함으로써 관련 질의를 단순하게 작성할 수 있다

## 10번



아래 내용 중 ( ? )에 들어갈 알맞은 것은?

유저를 생성하면 많은 권한을 부여해야 한다. 이에 대한 복잡함과 부담을 줄이기 위해 권한을 그룹으로 묶어 관리할 수 있도록 ( ? )을 제공한다.

1. GRANT
2. REVOKE
3. ROLE
4. RESOURCE

▼ 정답

(3)

ROLE에 포함되어 있는 권한들이 필요한 유저에게는 해당 ROLE만을 부여함으로써 빠르고 정확하게 필요한 권한을 부여할 수 있게 된다

## 04. SQL 기본 및 활용 3

▼ 문제 및 정답

### 01번



윈도우 함수에 대한 설명으로 옳지 않은 것은?

1. 윈도우 함수는 행과 행 간의 관계를 정의하기 위해서 제공되는 함수이다
2. WINDOW 함수의 PARTITION 구문과 GROUP BY 구문의 의미는 유사하다
3. 윈도우 함수를 사용해서 순위나 합계, 평균, 행 위치 등을 조작할 수 있다
4. GROUP BY 구문과 병행하여 중첩 사용할 수 있다

▼ 정답

(4)

윈도우 함수는 다른 함수와 달리 GROUP BY 구문과 병행하여 중첩(NEST) 사용할 수 없다

### 02번



절차형 SQL에 대한 설명으로 옳지 않은 것은?

1. 본질적 목적은 데이터 처리를 보다 효율적으로 하는 것에 있기 때문에 프로그래밍 언어처럼 구성한다
2. 사용자 정의 함수는 연속 실행 또는 구현이 복잡한 트랜잭션을 수행하는 PL/SQL 블록을 DB에 저장하기 위해 생성한다
3. 순차 처리를 지원하기 때문에 IF, LOOP, FOR WHILE 등과 같은 제어문, 반복문의 사용이 가능하다
4. 절차적 데이터 조작어는 사용자가 무슨 데이터를 원하는지만을 명세한다

▼ 정답

(4)

비절차적 데이터 조작어는 사용자가 무슨 (What) 데이터를 원하는지만을 명세하고, 절차적 데이터 조작어는 어떻게 (How) 데이터를 접근해야 하는지 명세하는 차이가 있다

### 03번\*



아래 SQL의 수행 결과를 차례대로 기술하라

EMP 테이블

| NAME  | EMPNO     |
|-------|-----------|
| Alien | 000450267 |
| Jay   | 0390000   |

SQL

```
SELECT LENGTH (SUBSTR (NAME, 2, 4)) + LENGTH (LTRIM (EMPNO, '0')) AS E1
FROM EMP ;
```

답 : \_\_\_\_\_ , \_\_\_\_\_

▼ 정답

(10, 8)

| E1 | 계산식                     |
|----|-------------------------|
| 10 | = lien (4) + 450267 (6) |
| 8  | = ay (2) + 390000 (6)   |

### 04번



파티션에서 가장 나중에 나오는 값을 구하는 WINDOW FUNCTION은?

1. FIRST\_VALUE
2. LAST\_VALUE
3. LEAD
4. LAG

▼ 정답

(2)

| 순서 관련 함수    | 설명   |
|-------------|--|
| FIRST_VALUE | - 파티션에서 가장 처음에 나오는 값을 구할 수 있습니다.<br>- MIN 함수를 사용해서 같은 결과를 구할 수 있습니다. |
| LAST_VALUE  | - 파티션에서 가장 나중에 나오는 값을 구할 수 있습니다.                                     |
| LAG         | - 이전 행을 가지고 올 수 있습니다.  |
| LEAD        | - 윈도우에서 특정 위치의 행을 가지고 올 수 있습니다. - 기본값은 1입니다.                         |

05번



4개의 테이블로부터 필요한 칼럼을 조회할 경우 최소 몇 개의 JOIN이 필요한가?

- 1. 5개
- 2. 4개
- 3. 3개
- 4. 2개

▼ 정답

(3)

여러 개의 테이블에서 원하는 데이터를 조회하기 위해서는 전체 테이블 개수에서 최소 N-1개의 JOIN 조건이 필요하다

06번\*



아래의 SQL 문을 실행하면 총 12개의 행이 조회된다. ( ? )에 들어갈 알맞은 것은?

TAB1

| A1 | A2 |
|----|----|
| 1  | A  |
| 2  |    |
| 3  | B  |
| 4  | C  |

TAB2

| A1 | A2 |
|----|----|
| 1  | A  |
| 2  |    |
| 3  | B  |

SQL

```
SELECT * FROM TAB1 ( ? ) TAB2;
```

- 1. CROSS JOIN
- 2. INENR JOIN
- 3. LEFT OUTER JOIN
- 4. RIGHT OUTER JOIN

▼ 정답

(1)

CROSS JOIN을 넣어 실행할 경우,

(TAB1 테이블 행 개수 4) \* (TAB2 테이블 행 개수 3) = 12 가 된다

## 07번



연산자 우선순위가 바르게 표기된 것은?

1. NOT 연산자 → 비교 연산자 → AND → OR
2. AND → OR → NOT 연산자 → 비교 연산자 → 괄호
3. 괄호 → 비교 연산자 → AND → OR → NOT 연산자
4. 괄호 → NOT 연산자 → 비교 연산자 → AND → OR

▼ 정답

(4)

우선순위를 고려하지 않고 WHERE 문을 작성하면 의도하지 않은 동작이 발생할 수 있으므로 실수하기 쉬운 논리 연산자의 경우 괄호를 사용하여 우선순위를 표시하는 것을 권장한다

## 08번\*



아래 SQL과 동일한 수행 결과를 반환하는 SQL문으로 옳은 것은?

```
SELECT *  
FROM employee  
WHERE NVL (COMM, 0) > 0;
```

1. 

```
SELECT *  
FROM employee  
WHERE COMM IS NULL;
```

2. 

```
SELECT *  
FROM employee  
WHERE COMM IS NOT NULL;
```

3. 

```
SELECT *  
FROM employee  
WHERE COMM > 0;
```

4. 

```
SELECT *  
FROM employee  
WHERE COMM > 0  
OR COMM IS NOT NULL;
```

▼ 정답

(3)

문제에 있는 SQL은 < 'COMM이 NULL일 때 0으로 만들고, COMM이 NULL이 아닐 때는 0보다 크다' = COMM이 0보다 크다 >로 해석할 수 있다.

해당 문제의 SQL은 3번과 같은 형태로 작성하는 것이 더 좋은 SQL이다.



## 09번



순수 관계 연산자로 옳지 않은 것은?

1. UNION 연산
2. SELECT 연산
3. PROJECT 연산
4. DIVIDE 연산

▼ 정답

(1)

UNION 연산은 일반 집합 연산자이며 수학의 합집합을 제공하기 위한 연산자이다

## 10번



NATURAL JOIN에 대한 설명으로 옳지 않은 것은?

1. NATURAL JOIN은 SQL Server에서는 지원하지 않는다
2. NATURAL JOIN사용 시 칼럼에 ALIAS를 붙일 수 없다
3. NATURAL JOIN은 WHERE 절에서 JOIN 조건을 추가할 수 없다
4. 서로 다른 이름을 갖는 칼럼에 대해 자동으로 EQUI JOIN을 수행한다

▼ 정답

(4)

NATURAL JOIN은 INNER JOIN의 하위 개념으로 두 테이블의 JOIN 되는 칼럼을 별도로 지정하지 않아도 동일한 이름을 갖는 모든 칼럼들에 대해 자동으로 EQUI JOIN을 수행한다

## 05. SQL 기본 및 활용 4

▼ 문제 및 정답

### 01번



Join 기법에 대한 설명으로 옳지 않은 것은?

1. Sort Merge Join은 동등 조인에 대한 작업은 불가능하다
2. Sort Merge Join은 넓은 범위의 데이터를 처리할 때 주로 사용되는 기법이다
3. Nested Loops Join은 결과의 행 수가 적은 테이블을 조인의 선행 테이블로 선택하는 것이 유리하다
4. Hash Join은 결과 행의 수가 적은 테이블을 선행 테이블로 사용하는 것이 좋다

▼ 정답

(1)

Sort Merge Join은 동등 조인뿐만 아니라 비동등 조인에 대해서도 조인 작업이 가능하다는 장점이 있다

### 02번



인덱스에 대한 설명으로 옳은 것만 묶은 것은?

- 가) 규칙 기반 옵티마이저는 규칙에 따라 적절한 인덱스가 존재한다면 항상 전체 테이블 스캔을 사용한다
- 나) 비클러스터형 인덱스는 물리적으로 재배열 하지 않고, 별도의 페이지를 구성한다
- 다) 인덱스를 스캔하여 테이블로 데이터를 찾아가는 방식은 작업은 부하가 큰 방식이다
- 라) 인덱스는 데이터를 찾을 때는 쉽게 찾을 수 있으나 데이터를 추가하거나 혹은 삭제할 때는 인덱스를 같이 수정해야 하기에 시스템의 부하를 가중시킨다

1. 다, 라
2. 나, 라
3. 가, 다
4. 나, 다

▼ 정답

(2)

가 : 규칙 기반 옵티마이저는 규칙에 따라 적절한 인덱스가 존재한다면 전체 테이블 스캔보다는 항상 인덱스 스캔을 사용한다

다 : 인덱스를 스캔하여 테이블로 데이터를 찾아가는 방식은 랜덤 액세스 방식을 사용하며 이러한 작업은 부하가 크기 때문에 매우 많은 양의 데이터를 읽을 경우에는 테이블 전체 스캔이 유리할 수 있다

## 03번



셀프 조인에 대한 설명으로 옳은 것은?

1. 두 테이블의 연관 관계를 찾는 작업이다
2. 테이블을 식별하기 위해 반드시 별칭을 붙여줘야 한다
3. 테이블 A와 테이블 B간의 JOIN을 의미한다
4. FROM 절에 서로 다른 테이블이 두 번 이상 나타난다

▼ 정답

(2)

조인(SELF JOIN)은 동일한 테이블 간의 JOIN을 의미하며, 테이블과 칼럼의 이름이 모두 동일하기 때문에 각각의 테이블을 식별하기 위해서 반드시 테이블의 ALIAS(별칭)을 붙여주어야 한다

## 04번



오라클 계층형 질의에 대한 설명으로 옳지 않은 것은?

1. WHERE절은 계층적으로 진행 후에 특정한 조건에 맞는 데이터만 가져온다
2. PRIOR는 계층 구조 전개의 시작 위치를 지정하는 구문이다
3. 루트 노드의 LEVEL 값은 1이다
4. ORDER SIBLINGS BY절은 형제 노드 사이에서 정렬을 수행하는 구문이다

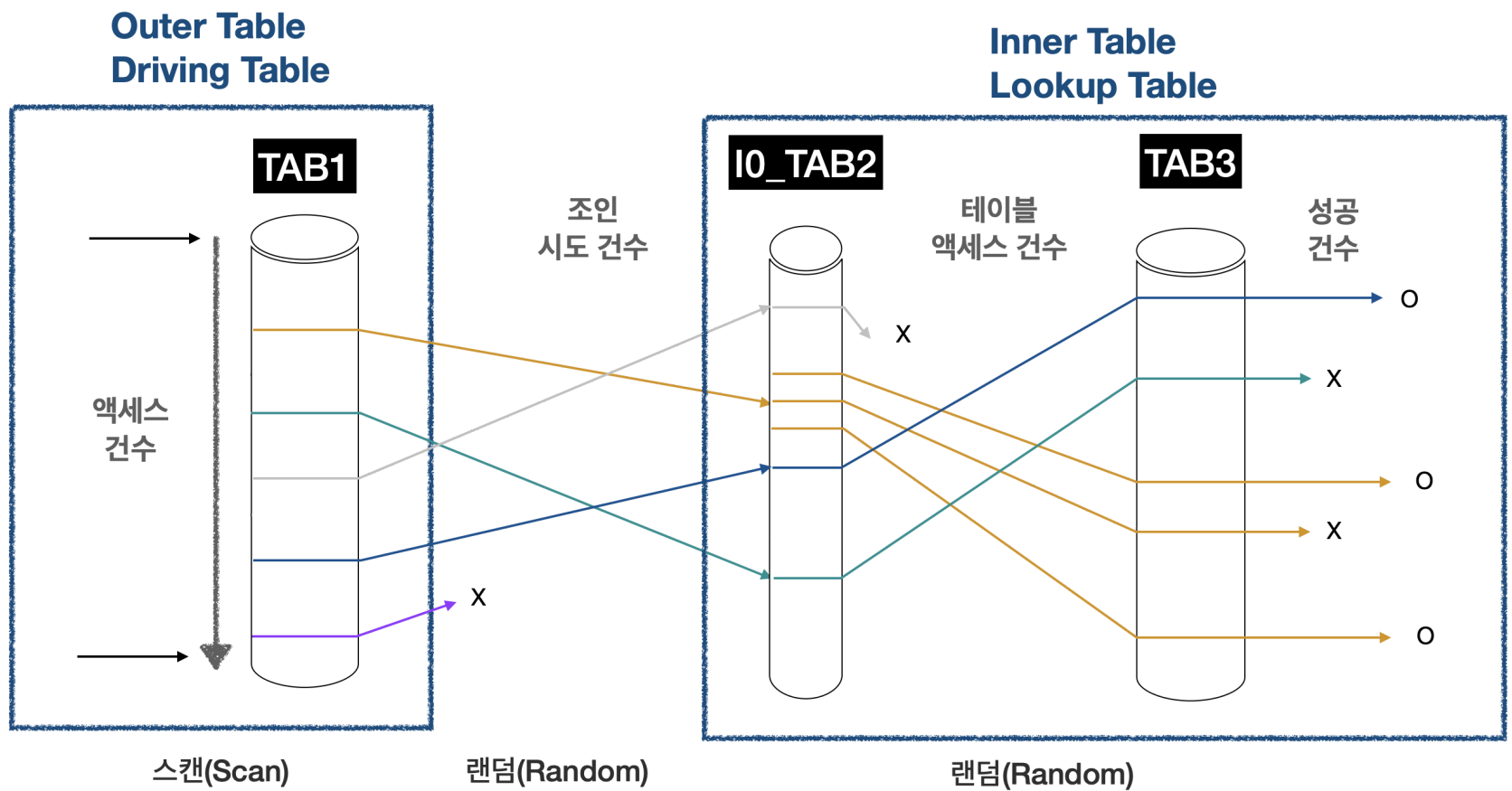
▼ 정답

(2)

계층 구조 전개의 시작 위치를 지정하는 구문은 START WITH이다

## 05번

📖 아래의 그림과 같은 SQL 처리 흐름도에 대한 설명으로 옳지 않은 것은?



1. 실행 계획을 시각적으로 표현한 도표이다
2. 테이블을 읽기 위해 인덱스 스캔을 수행했는지 혹은 전체 테이블을 스캔했는지와 조인 기법 등을 표현할 수 있다
3. SQL 내부적인 처리 절차를 표현하고 있다
4. 실행 시간을 확인할 수 있다

▼ 정답

(4)

실행계획은 예상 정보이므로 실제 실행 시간은 알 수 없다

## 06번

📖스칼라 서브쿼리에 대한 설명으로 옳은 것은?

1. 하나의 행과 하나의 칼럼만을 반환한다
2. 여러 개의 서브쿼리가 겹쳐 있기 때문에 '중첩서브쿼리'라고도 부른다
3. FROM 절에 위치한 서브쿼리를 스칼라 서브쿼리라고 한다
4. 그룹함수와 함께 사용될 때 그룹핑된 결과에 대해 부가적인 조건을 주기 위해서 사용한다

▼ 정답

(1)

- 2번 : WHERE절 안에 들어있는 서브쿼리는 여러 개의 서브쿼리가 겹쳐 있기 때문에 '중첩서브쿼리 (nested subqueries)' 라고도 부른다

- 3번 : SELECT 절에 위치한 서브쿼리를 스칼라 서브쿼리라고 한다
- 4번 : HAVING절에 위치한 서브쿼리는 그룹함수와 함께 사용될 때 그룹핑된 결과에 대해 부가적인 조건을 주기 위해서 사용한다

## 07번\*



아래 SQL문을 'DECODE'를 사용해서 변경하여 작성하라

```
SELECT CASE ranking WHEN 1 THEN 'P' ELSE 'S'
FROM TAB1;
```

### ▼ 정답

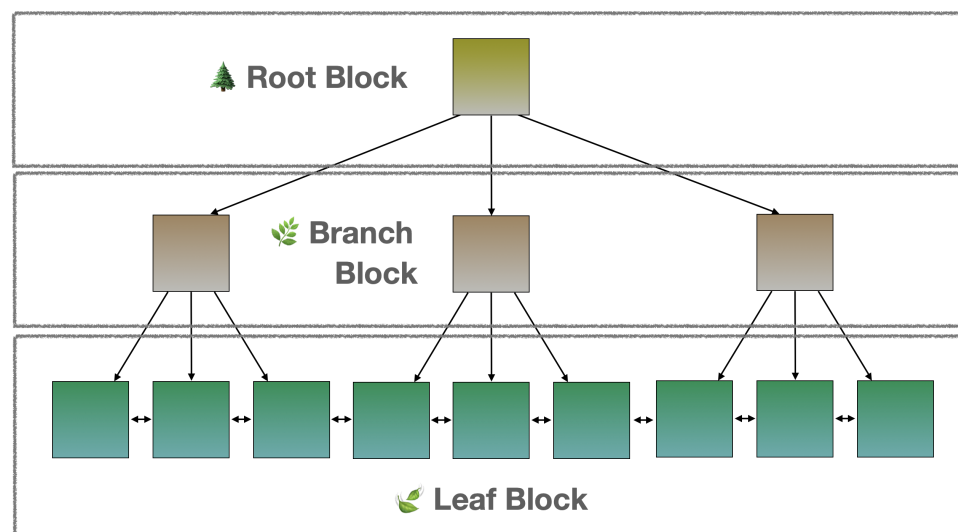
```
SELECT DECODE(ranking, 1, 'P', 'S') FROM TAB1;
```

ranking이 1과 같으면 'P', 그렇지 않으면 'S'를 출력한다

## 08번



아래 그림과 같은 형태의 인덱스에 대한 설명으로 옳지 않은 것은?



1. Balanced-Tree로 부르며 DBMS에서 가장 일반적인 인덱스이다
2. 리프 블록은 인덱스와 ROWID로 구성되어 있다
3. BETWEEN, >, < 등과 같은 검색하는 범위 모두에 적합하지만 = 로 검색하는 일치 검색은 부적합하다
4. 일반적으로 테이블 내의 데이터 중 10% 이하의 데이터를 검색할 때 유리한 인덱스이다

### ▼ 정답

(3)

B-Tree 인덱스는 = 로 검색하는 일치 검색과 BETWEEN, >, < 등과 같은 검색하는 범위 모두에 적합한 구조이다

## 9번



실행계획에 대한 설명으로 옳지 않은 것은?

1. 실행계획을 구성하는 요소에는 최적화 정보, 액세스 기법 등이 있다
2. Cardinality는 주어진 조건을 만족한 결과 집합 혹은 조인 조건을 만족한 결과 집합의 건수를 의미한다
3. 실행 계획이란 SQL의 요구한 사항을 처리하기 위한 절차와 방법이다
4. FROM 절에 존재하는 테이블의 개수가  $n$ 이라면 논리적으로 가능한 조인의 순서는  $n-1$  만큼 존재한다

▼ 정답

(4)

FROM 절에 존재하는 테이블의 개수가  $n$ 이라면 논리적으로 가능한 조인의 순서는  $n!$  만큼 존재한다