GUSTAVO CORONEL
DESARROLLA SOFTWARE

TALLER DE ORACLE SQL

CLASE 5

GUSTAVO CORONEL
desarrollasoftware.com

CONTENIDO

OPERADORES DE COMPARACIÓN ALL, ANY Y SOME

Operador ALL

El operador ALL se utiliza para comparar un valor con una lista o una subconsulta. Debe ir precedido de =, !=, >, <, <=, >= y seguido de una lista o subconsulta.

Cuando después del operador ALL se tiene una lista, el optimizador expande la condición inicial a todos los elementos de la lista y los une con el operador AND, como se muestra a continuación.

```
SELECT employee_id, salary
FROM   employees
WHERE  salary > ALL (9500, 10500, 11500);
```

Su transformación equivalente es la siguiente sentencia:

```
SELECT employee_id, salary
FROM   employees
WHERE  salary>9500 AND salary>10500 AND salary>11500;
```

Para ambos casos el resultado es el mismo:

```
EMPLOYEE_ID     SALARY
----------- ----------
        100      24000
        101      17000
        102      17000
        108      12008
        145      14000
        146      13500
        147      12000
        201      13000
        205      12008

9 filas seleccionadas.
```

Cuando después de operador ALL se tiene una subconsulta, como se ilustra a continuación:

```
SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE  e1.salary > ALL (SELECT distinct e2.salary
                        FROM   employees e2
                       WHERE  e2.department_id = 20);

FIRST_NAME               SALARY
-------------------- ----------
Karen                     13500
John                      14000
Lex                       17000
Neena                     17000
Steven                    24000
```

Aquí témenos su transformación utilizando subconsulta correlacionada:

```
SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE  NOT EXISTS (SELECT 1
                   FROM   employees e2
                   WHERE  e2.department_id = 20
                   AND    e1.salary <= e2.salary );

FIRST_NAME               SALARY
-------------------- ----------
Karen                     13500
John                      14000
Lex                       17000
Neena                     17000
Steven                    24000
```

Asumiendo que las subconsultas no devuelvan cero filas, se pueden hacer las siguientes declaraciones para las versiones de lista y subconsulta:

x = ALL (...) : El valor debe coincidir con todos los valores de la lista para evaluar como TRUE.

x != ALL (...) : El valor no debe coincidir con ningún valor en la lista para evaluar como TRUE.

x > ALL (...) : El valor debe ser mayor que el valor más grande de la lista para evaluar como TRUE.

x < ALL (...) : El valor debe ser menor que el valor más pequeño de la lista para evaluar como TRUE.

x >= ALL (...) : El valor debe ser mayor o igual que el valor más grande de la lista para evaluar como TRUE.

x <= ALL (...) : El valor debe ser menor o igual que el valor más pequeño de la lista para evaluar como TRUE.

Si una subconsulta devuelve cero filas, la condición se evalúa como TRUE. En el siguiente ejemplo, la subconsulta devuelve cero filas, lo que significa que la expresión completa "salary > ALL (cero filas)" se evalúa como VERDADERO, por lo que se muestran todas las filas.

```
SELECT salary FROM employees e2 WHERE e2.department_id = 120;

no se ha seleccionado ninguna fila

SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE e1.salary > ALL (SELECT salary
          FROM employees e2
          WHERE e2.department_id = 120);
```

Operador ANY

El operador ANY se utiliza para comparar un valor con una lista o una subconsulta. Debe ir precedido de =, !=, >, <, <=, >= y seguido de una lista o subconsulta.

Cuando el operador ANY va seguida de una lista, el optimizador expande la condición inicial a todos los elementos de la lista y los une con el operador OR, como se muestra a continuación:

```
SELECT employee_id, salary
FROM   employees
WHERE  salary > ANY (13000, 14000, 15000);

EMPLOYEE_ID     SALARY
----------- ----------
        100      24000
        101      17000
        102      17000
        145      14000
        146      13500
```

La transformación sin ANY seria de esta manera:

```
SELECT employee_id, salary
FROM   employees
WHERE  salary > 13000 OR salary > 14000 OR salary > 15000;

EMPLOYEE_ID     SALARY
- ---------- ----------
        100      24000
        101      17000
        102      17000
        145      14000
        146      13500
```

Cuando el operador ANY va seguida de una subconsulta, como se muestra a continuación:

```
SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE  e1.salary > ANY (SELECT distinct e2.salary
                        FROM   employees e2
                        WHERE  e2.department_id = 90);
```

A continuación, tenemos su transformación utilizando subconsulta correlacionada:

```
SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE  EXISTS (SELECT 1
               FROM   employees e2
               WHERE  e2.department_id = 90
               AND    e1.salary > e2.salary );
```

Asumiendo que las subconsultas no devuelvan cero filas, se pueden hacer las siguientes declaraciones para las versiones de lista y subconsulta:

x = ALL (...) : El valor debe coincidir con uno o más valores en la lista para evaluar como TRUE.

x != ALL (...) : El valor no debe coincidir con ningún valor en la lista para evaluar como TRUE.

x > ALL (...) : El valor debe ser mayor que el valor más pequeño de la lista para evaluar como TRUE.

x < ALL (...) : El valor debe ser menor que el valor más grande en la lista para evaluar como TRUE.

x >= ALL (...) : El valor debe ser mayor o igual que el valor más pequeño de la lista para evaluar como TRUE.

x <= ALL (...) : El valor debe ser menor o igual que el valor más grande en la lista para evaluar como TRUE.

Si una subconsulta devuelve cero filas, la condición se evalúa como FALSE. En el siguiente ejemplo, la subconsulta devuelve cero filas, lo que significa que la expresión completa "salary > ANY (cero filas)" se evalúa como FALSE, por lo que se muestran todas las filas.

```
SELECT salary FROM employees e2 WHERE e2.department_id = 120;

no se ha seleccionado ninguna fila

SELECT e1.first_name, e1.salary
FROM   employees e1
WHERE e1.salary > ANY (SELECT salary
FROM employees e2
WHERE e2.department_id = 120);

no se ha seleccionado ninguna fila
```

Operador SOME

Los operadores de comparación SOME y ANY hacen exactamente lo mismo y son completamente intercambiables.

COMMON TABLE EXPRESSIONS (CTE)

CTE es una consulta simple para simplificar las diferentes clases de consultas SQL, ya que el concepto de tabla derivada simplemente no es adecuado. Se puede definir como un conjunto de resultados temporal con nombre que solo puede existir dentro del alcance de una sola declaración (En este caso, la declaración aquí significa SELECT y también declaraciones DML como INSERT y UPDATE) y se puede hacer referencia a ella dentro de esa declaración en particular varias veces según lo requiera el desarrollador.

```
WITH
CTE_NAME1 (column1, column2,…)] AS
(
    CTE QUERY
)[,
CTE_NAME1 (column1, column2,…)] AS
(
    CTE QUERY
)] [, … ]
```

Ejemplo 1

```
WITH
CTE AS (
     SELECT employee_id, first_name
     FROM employees
     WHERE department_id = 60
)
select * from CTE;

EMPLOYEE_ID FIRST_NAME
----------- --------------------
        103 Alexander
        104 Bruce
        105 David
        106 Valli
        107 Diana
```

Ejemplo 2

```
WITH
CTE1 AS (
     SELECT employee_id, first_name, department_id
     FROM employees
),
CTE2 AS (
     SELECT department_id, department_name
     FROM departments
)
SELECT
     CTE1.employee_id,
     CTE1.first_name,
     CTE2.department_name
FROM CTE1 JOIN CTE2
ON CTE1.department_id = CTE2.department_id;
```

EJERCICIOS PROPUESTOS

Ejercicio 1

Se necesita una consulta para conocer el importe de la planilla por departamento.

Esquema: RECURSOS, HR

El modelo del reporte es el siguiente:

NOMBRE DEPARTAMENTO	CANTIDAD DE EMPLEADOS	IMPORTE	COMISION	TOTAL

IMPORTE: El cálculo del IMPORTE de la planilla es en función al sueldo del empleado.

COMISION: El cálculo de la COMISIÓN de la planilla es en función a la comisión que recibe el empleado

TOTAL: Para el cálculo del TOTAL de la planilla debes considerar la suma del IMPORTE más las COMISION.

Ejercicio 2

Se necesita una consulta para saber la cantidad de empleados por ciudad.

Esquema: RECURSOS, HR

El modelo del resultado es el siguiente:

CIUDAD	CANTIDAD DE EMPLEADOS	PORCENTAJE DEL TOTAL

Ejercicio 3

Desarrolle una consulta para encontrar el empleado que tiene el menor salario por departamento.

Esquema: `RECURSOS, HR`

Ejercicio 4

Se necesita una consulta para encontrar el importe proyectado y el importe real recaudado por curso.

Esquema: `EDUCA`

El modelo del resultado es el siguiente:

NOMBRE DEL CURSO	MATRICULADOS	IMPORTE PROYECTADO	IMPORTE RECAUDADO

Ejercicio 5

Se necesita una consulta para saber el importe en SOLES y DOLARES en cada sucursal, y cuanto representa del total en %.

Esquema: EUREKA