



 **ACADEMY**

# Programación de Bases de Datos con SQL

3-1

Comparaciones Lógicas y Reglas de Prioridad



**ORACLE** ACADEMY

Copyright © 2017, Oracle y sus filiales. Todos los derechos reservados.

# Objetivos

En esta lección se abordan los siguientes objetivos:

- Evaluar las comparaciones lógicas para restringir las filas devueltas en función de dos o más condiciones
- Aplicar las reglas de prioridad para determinar el orden en el que se evalúan y calculan las expresiones

# Objetivo

- No hay muchas cosas en la vida que dependan de una sola condición.
- Por ejemplo, si desea ir a la universidad, probablemente necesite buenas notas y dinero para pagarla.
- Si tiene dinero adicional, puede guardarlo o gastarlo.
- Si desea ir al cine, es posible que no desee ir este fin de semana y puede que no desee sentarse en las primeras 10 filas de la sala.

# Objetivo

- En SQL, por lo general, es aconsejable poder restringir las filas devueltas en una consulta en función de dos o más condiciones.
- Como jefe de un negocio de comida rápida, puede que necesite conocer los nombres de su los empleados que sean cocineros o los que toman los pedidos.
- No necesita ni desea una lista de todo el personal, solo quiere un subconjunto de ella.
- Los operadores condicionales como AND, NOT y OR facilitan el uso de este tipo de solicitudes.

# Condiciones Lógicas

- Las condiciones lógicas combinan el resultado de dos condiciones de componentes para producir un único resultado según dichas condiciones.
- Por ejemplo, para asistir a un concierto de rock, debe adquirir una entrada y el transporte para llegar allí.
- Si se cumplen las dos condiciones, puede ir al concierto.
- ¿Qué hacer si no puede obtener transporte, puede ir?

# Condiciones Lógicas

- Otra condición lógica combina dos condiciones de componentes con OR.
- Todos los empleados recibirán un aumento si tienen un registro de asistencia perfecto o cumplen con su cuota de ventas mensual.
- Si un empleado cumple cualquiera de estas dos condiciones, el empleado obtiene un aumento.

# Operadores Lógicos

- Un operador lógico combina los resultados de dos o más condiciones para producir un único resultado.
- Se devuelve un resultado SOLO SI el resultado global de la condición es verdadero.
- AND: devuelve TRUE (verdadero) si ambas condiciones son verdaderas.
- OR: devuelve TRUE (verdadero) si cualquier condición es verdadera.
- NOT: devuelve TRUE (verdadero) si la condición es falsa.



# Operador AND

- En la siguiente consulta, los resultados devueltos serán las filas que cumplan ambas condiciones especificadas en la cláusula WHERE.

```
SELECT last_name, department_id, salary
FROM employees
WHERE department_id > 50 AND salary > 12000;
```

LAST_NAME	DEPARTMENT_ID	SALARY
King	90	24000
Kochhar	90	17000
De Haan	90	17000

# Operador AND

- Otro ejemplo del uso de AND en la cláusula WHERE.

```
SELECT last_name, hire_date, job_id
FROM employees
WHERE hire_date > '01-Jan-1998' AND job_id LIKE 'SA%';
```

LAST_NAME	HIRE_DATE	JOB_ID
Zlotkey	29-Jan-2000	SA_MAN
Taylor	24-Mar-1998	SA_REP
Grant	24-May-1999	SA_REP

Recordatorio: las fechas se deben introducir entre comillas simples y el formato por defecto es dd/Mes/aaaa

# Operador OR

- Si la cláusula WHERE utiliza la condición OR, los resultados devueltos de una consulta serán las filas que cumplan una de las condiciones de OR.
- En otras palabras, todas las filas devueltas tienen un location\_id de 2500 o tienen un manager\_id igual a 124.

```
SELECT department_name, manager_id, location_id
FROM departments
WHERE location_id = 2500 OR manager_id=124;
```

DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
Envío	124	1500
Ventas	149	2500

# Operador NOT

- El operador NOT devolverá las filas que no cumplan con la condición de la cláusula WHERE.

```
SELECT department_name, location_id  
FROM departments  
WHERE location_id NOT IN (1700,1800);
```

DEPARTMENT_NAME	LOCATION_ID
Envío	1500
IT	1400
Ventas	2500

# Reglas de Prioridad o Qué Ocurre en Primer Lugar

- Considere la siguiente sentencia SELECT.
- ¿En qué orden se evalúan las expresiones calculadas?

```
SELECT last_name||' '||salary*1.05  
  As "Employee Raise"  
FROM employees  
WHERE department_id IN(50,80)  
AND first_name LIKE 'C%'  
OR last_name LIKE '%s%';
```

- Afortunadamente, cuando las cosas se complican de esta manera, SQL cuenta con algunas reglas básicas fáciles de seguir.

# Reglas de Prioridad o Qué Ocurre en Primer Lugar

- Tenga en cuenta que el operador AND se evalúa antes que el operador OR.
- Esto significa que, para que el ejemplo en la diapositiva anterior, si no se cumple alguna de las condiciones de la sentencia AND, se utilizará el operador OR para seleccionar las filas.
- Es importante recordar este concepto.

ORDEN	OPERADORES
1	Aritméticos + - * /
2	Concatenación
3	Comparación <, <=, >, >=, <>
4	IS (NOT) NULL, LIKE, (NOT) IN
5	(NOT) BETWEEN
6	NOT
7	AND
8	OR

# Reglas de Prioridad o Qué Ocurre en Primer Lugar

- En primer lugar, se evalúa la condición AND, por lo que se devolverán todos los empleados que trabajen en el departamento 80 o 50, y tengan un nombre que empiece por "C".
- La cláusula OR se evalúa a continuación y se devuelven los empleados cuyo apellido contenga "s".

```
SELECT last_name||' '||salary*1.05  
  As "Employee Raise",  
     department_id, first_name  
FROM employees  
WHERE department_id IN(50,80)  
AND first_name LIKE 'C%'  
OR last_name LIKE '%s%';
```

Aumento al Empleado	DEPARTMENT_ID	FIRST_NAME
Higgins 12600	110	Shelley
Mourgos 6090	50	Kevin
Rajs 3675	50	Trenna
Davies 3255	50	Curtis
Matos 2730	50	Randall
Vargas 2625	50	Peter
Ernst 6300	60	Bruce
Hartstein 13650	20	Michael

El empleado Curtis Davis está en el departamento 50 y su nombre empieza por "C", por lo que lo devuelve la cláusula AND.

La cláusula OR devuelve, a continuación, cualquier empleado cuyo apellido contenga la letra "s".

Si alguna de las condiciones no se cumplen en la sentencia AND, la condición OR tendría que evaluar si son verdaderas para devolver alguna fila.

# Reglas de Prioridad o Qué Ocurre en Primer Lugar

- En este ejemplo, el orden de los operadores OR y AND se ha invertido con respecto a la diapositiva anterior.

```
SELECT last_name||' '||salary*1.05  
As "Employee Raise", department_id, first_name  
FROM employees  
WHERE department_id IN(50,80)  
OR first_name LIKE 'C%'  
AND last_name LIKE '%s%';
```

El orden de las operaciones es:

1. first\_name comienza con una "C" y last\_name contiene una "s". Deben cumplirse ambas condiciones para devolverlas.
2. Se devolverá cualquier instancia de los empleados de los departamentos 50 y 80.



# Reglas de Prioridad o Qué Ocurre en Primer Lugar

- Agregar paréntesis cambia la forma en la que se evalúa la cláusula WHERE y las filas devueltas.

```
SELECT last_name||' '||salary*1.05  
As "Employee Raise", department_id, first_name  
FROM employees  
WHERE department_id IN(50,80)  
OR first_name LIKE 'C%'  
AND last_name LIKE '%s%';
```

El orden de las operaciones es:

1. Los valores de los paréntesis están seleccionados.
2. Se devolverán todas las instancias de los valores entre los paréntesis que también contengan la letra "s" en su last\_name (apellido).

# Terminología

Entre los términos clave utilizados en esta lección se incluyen:

- AND
- OR
- NOT
- Reglas de prioridad

# Resumen

En esta lección, debe haber aprendido lo siguiente:

- Evaluar las comparaciones lógicas para restringir las filas devueltas en función de dos o más condiciones
- Aplicar las reglas de prioridad para determinar el orden en el que se evalúan y calculan las expresiones



 **ACADEMY**