

ORACLE

Academy

Database Programming with SQL

2-3

Operadores de Comparaço

ORACLE
Academy



Objetivos

- Esta lição abrange os seguintes objetivos:
 - Aplicar o operador de comparação apropriado para retornar o resultado desejado
 - Demonstrar o uso apropriado das condições BETWEEN, IN e LIKE para retornar o resultado desejado
 - Distinguir entre zero e NULO, sendo este um valor que está indisponível, não foi atribuído, é desconhecido ou é inaplicável
 - Explicar o uso das condições de comparação e de NULO

Finalidade

- Usamos comparações em nossas conversas diárias e nem pensamos sobre isso
 - "Posso encontrar você entre 10:00h e 11:00h."
 - "Estou procurando uma calça jeans como a que você está usando."
 - "Se me lembro bem, os melhores lugares para o show estão nas fileiras 100, 200 e 300."



Finalidade

- A necessidade de expressar esses tipos de comparações também existem em SQL
- As comparações são usadas para encontrar dados em uma tabela que atendam a certas condições
- Formular uma cláusula SELECT para retornar dados específicos é um recurso avançado de SQL

Operadores de Comparaço

- Você ja est familiarizado com os operadores de comparao, como igual a (=), menor que (<) e maior que (>)
- A linguagem SQL tem outros operadores para recuperar conjuntos de dados especificos
- So eles:
 - BETWEEN...AND
 - IN
 - LIKE

BETWEEN...AND

- O operador BETWEEN...AND é usado para selecionar e exibir linhas com base em uma faixa de valores
- Quando usado com a cláusula WHERE, a condição BETWEEN...AND retornará uma faixa de valores entre os limites mais baixo e mais alto especificados, incluindo-os

BETWEEN...AND

- Observe que, no exemplo do banco de dados de funcionários, os valores retornados incluem o valor do limite mais baixo e o valor do limite mais alto
- Os valores especificados com a condição BETWEEN são inclusivos
- Observe também que o valor do limite inferior deve ser listado primeiro

```
SELECT last_name, salary  
FROM employees  
WHERE salary BETWEEN 9000 AND 11000;
```

- Observe que a saída incluiu os valores dos limites mais baixo e mais alto

| LAST_NAME | SALARY |
|-----------|--------|
| Zlotkey | 10500 |
| Abel | 11000 |
| Hunold | 9000 |

BETWEEN...AND

- Usar BETWEEN...AND é o mesmo que usar a seguinte expressão:

```
WHERE salary >= 9000 AND salary <=11000;
```

- Na verdade, não existe benefício em usar uma expressão no lugar da outra
- Usamos BETWEEN...AND para tornar o código mais simples de ler

IN

- A condição IN também é conhecida como "condição de associação"
- Ela é usada para testar se um valor está DENTRO (IN, em inglês) de um conjunto específico de valores
- Por exemplo, a condição IN poderia ser usada para identificar alunos cujos números de identificação são 2349, 7354 ou 4333 ou pessoas cujo código de ligação internacional é 1735, 82 ou 10

```
SELECT city, state_province,  
country_id  
FROM locations  
WHERE country_id IN('UK', 'CA');
```

| CITY | STATE_PROVINCE | COUNTRY_ID |
|---------|----------------|------------|
| Toronto | Ontario | CA |
| Oxford | Oxford | UK |

IN

- Nesse exemplo, a cláusula WHERE também poderia ser gravada como um conjunto de condições OR:

```
SELECT city, state_province, country_id  
FROM locations  
WHERE country_id IN('UK', 'CA');  
...  
WHERE country_id = 'UK' OR country_id = 'CA';
```

- Assim como acontece com BETWEEN...AND, a condição IN pode ser gravada com qualquer uma dessas sintaxes e fornecer o mesmo resultado

LIKE

- Você já foi comprar algo que viu em uma revista ou na televisão, mas não tinha certeza de como era o item?
- A pesquisa em bancos de dados é muito parecida com isso
- Um gerente pode saber que o sobrenome de um funcionário começa com "S", mas não conhecer o nome inteiro dele
- Felizmente, em SQL, a condição LIKE permite selecionar linhas correspondentes a caracteres, datas ou padrões de números
- Dois símbolos — (%) e (_) — chamados de caracteres curinga, podem ser usados para construir uma string de pesquisa

LIKE

- O símbolo de porcentagem (%) é usado para representar qualquer sequência de zeros ou mais caracteres
- O símbolo de sublinhado (_) é usado para representar um caractere
- No exemplo mostrado abaixo, serão retornados todos os funcionários com sobrenomes (last_name) que começam com qualquer letra seguida por "o" e, depois, por qualquer número de letras

```
SELECT last_name  
FROM employees  
WHERE last_name LIKE '_o%';
```

| LAST_NAME |
|-----------|
| Kochhar |
| Lorentz |
| Mourgos |

LIKE

```
SELECT last_name  
FROM employees  
WHERE last_name LIKE '_o%';
```

- Quais dos sobrenomes a seguir poderiam ter sido retornados pela consulta acima?
 - 1. Sommersmith
 - 2. Oog
 - 3. Fong
 - 4. Mo

LIKE

- Mais uma opção importante:
 - Quando você precisar ter uma correspondência exata para uma string que tenha um caractere % ou _, será preciso indicar que % ou _ não são curingas, mas parte do item que está sendo pesquisado

LIKE

- A opção ESCAPE pode ser usada para indicar que _ ou % faz parte do nome e não é um valor curinga
- Por exemplo, se quiséssemos recuperar, da tabela de funcionários, o item JOB_ID contendo o padrão _R, teríamos de usar um caractere de escape para mostrar que estávamos procurando por um sublinhado, e não um caractere qualquer

```
SELECT last_name, job_id  
FROM EMPLOYEES  
WHERE job_id LIKE '%\_R%' ESCAPE '\';
```

- Esse exemplo usa a barra invertida (\) como caractere de escape, mas é possível usar qualquer um

LIKE

- Sem a opção ESCAPE, todos os funcionários que tivessem a letra R em JOB_ID seriam retornados

```
SELECT last_name, job_id  
FROM EMPLOYEES  
WHERE job_id LIKE '%_R%'
```

| LAST_NAME | JOB_ID |
|-----------|----------|
| Abel | SA_REP |
| Davies | ST_CLERK |
| Ernst | IT_PROG |
| Fay | MK_REP |
| Fay | MK_REP |
| Grant | SA_REP |
| Higgins | AC_MGR |
| Hunold | IT_PROG |
| King | AD_PRES |
| Lorentz | IT_PROG |
| Matos | ST_CLERK |
| Rajs | ST_CLERK |
| Taylor | SA_REP |
| Vargas | ST_CLERK |

IS NULL, IS NOT NULL

- Lembra do NULO?
- Ele é o valor que está indisponível, não foi atribuído, é desconhecido ou é inaplicável
- Sempre é preferível ser capaz de testar o valor NULO
- Você pode querer saber todas as datas em junho nas quais, até o momento, um show não está marcado
- Você também pode querer saber todos os clientes cujos números de telefone não estão registrados no seu banco de dados

IS NULL, IS NOT NULL

- A condição IS NULL testa os dados indisponíveis, não atribuídos ou desconhecidos
- IS NOT NULL testa os dados que estão disponíveis no banco de dados
- No exemplo no slide a seguir, a cláusula WHERE é gravada para recuperar todos os sobrenomes (last_name) dos funcionários que não têm um gerente

IS NULL, IS NOT NULL

```
SELECT last_name, manager_id  
FROM employees  
WHERE manager_id IS NULL;
```

| LAST_NAME | MANAGER_ID |
|-----------|------------|
| King | |

- O funcionário King é o presidente da empresa, portanto, não tem um gerente

```
SELECT last_name, commission_pct  
FROM employees  
WHERE commission_pct IS NOT NULL;
```

| LAST_NAME | COMMISSION_PCT |
|-----------|----------------|
| Zlotkey | .2 |
| Abel | .3 |
| Taylor | .2 |
| Grant | .15 |

- IS NOT NULL retorna as linhas que têm um valor na coluna commission_pct

Terminologia

- Estes são os principais termos usados nesta lição:
 - BETWEEN...AND
 - IN
 - LIKE
 - IS NULL
 - IS NOT NULL

Resumo

- Nesta lição, você deverá ter aprendido a:
 - Aplicar o operador de comparação apropriado para retornar o resultado desejado
 - Demonstrar o uso apropriado das condições BETWEEN, IN e LIKE para retornar o resultado desejado
 - Distinguir entre zero e NULO, sendo este um valor que está indisponível, não foi atribuído, é desconhecido ou é inaplicável
 - Explicar o uso das condições de comparação e de NULO

ORACLE

Academy