



Criação de relatórios utilizando filtros, operadores aritméticos e de banco de dados

Criar relatórios a partir de dados existentes no Banco de dados, utilizando comandos de seleção, aplicando filtros, realizando cálculos, usando operadores próprios de banco de dados.

- NESTE TÓPICO
- > Operadores de banco de dados

> Pesquisa de caracteres (LIKE)

> Referências

Marcar
tópico



A criação de relatórios em SQL pode conter processamento, e este não tem obrigação de armazenar o resultado na tabela que o gerou. Assim, o resultado pode ser apenas processado e exibido em tempo de processamento, ou seja, no momento de execução da instrução. Para realizar esse processo, usa-se, como na programação, os operadores aritméticos cuja função é a realização de cálculos matemáticos simples, como soma, subtração, multiplicação e divisão. Sua representação segue na tabela a seguir:



Símbolo	Descrição
+	Adição ou Soma
-	Subtração
*	Multiplicação
/	Divisão

Os operadores aritméticos podem ser utilizados em quaisquer cláusulas do comando SELECT, exceto a cláusula FROM.

Exemplo

1. `SELECT nome, salario, salario * 280 / 100 COMISSAO FROM funcionario WHERE salario = 2500 AND salario * 280 / 100 > 200 ORDER BY salario * 280 / 100.`

Nessa instrução, serão exibidos o nome do funcionário, seu salário, o resultado de um processamento, que é o salário atual cadastrado na base de dados multiplicado por 280, e o resultado dessa multiplicação dividido por 100 com o apelido de COMISSAO. Essas colunas e esses dados pertencem à tabela funcionário, mas só retornarão em tela se a condição for obedecida, na qual o salário do funcionário deve ser igual a R\$ 2.500,00 e o resultado do processamento deve ser maior que R\$ 200,00. Ao final os dados serão ordenados pelo processamento.

Para utilizar os operadores aritméticos, devemos obedecer às regras de precedência (ordem de processamento).

Regra:

1 Multiplicação e divisão (*, /)

2 Adição e subtração (+, -)

Exemplo:

1. `SELECT nome, salario, (salario + 100) * 12 Bonus FROM funcionario WHERE matricula = 1;`

A instrução anterior irá mostrar o nome do funcionário, seu salário atual, um reajuste no salário atual de R\$ 100,00, o resultado desse processamento multiplicado por 12 mostrará no lugar da fórmula o apelido Bonus, mas isso só ocorrerá se a matrícula do funcionário for igual a 1.

Operadores de banco de dados

Esses operadores têm como função melhorar uma instrução que use os tradicionais operadores lógicos, otimizando assim a digitação do comando e melhorando a sua visualização.

São eles:

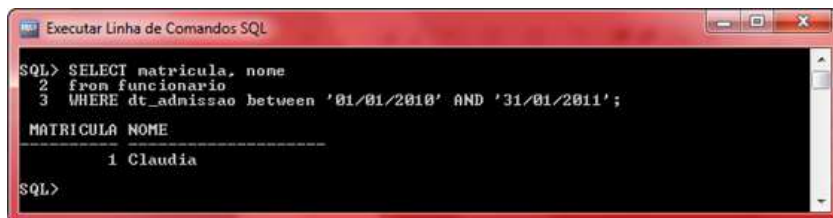
Operador	Descrição
BETWEEN ... AND...	ENTRE DOIS VALORES
NOT BETWEEN ... AND	FORA DA FAIXA ESTABELECIDADA
IN(lista)	IGUAL A VALOR(ES) DA LISTA
NOT IN	DIFERENTE DO(S) VALOR(ES) DA LISTA
LIKE	PESQUISA DE CARACTERES
IS NULL	VALORES NULOS
IS NOT NULL	VALORES DIFERENTES DE NULO

Entre dois valores (BETWEEN VALOR 1 AND VALOR 2).

Todas as linhas que possuam valores dentro da faixa determinada serão listadas.

Exemplo:

1. `SELECT matricula, nome from funcionario WHERE dt_admissao between '01/01/2010' AND '31/01/2011'`



```
SQL> SELECT matricula, nome
2 from funcionario
3 WHERE dt_admissao between '01/01/2010' AND '31/01/2011';

MATRICULA NOME
-----
1 Claudia

SQL>
```

Todos os funcionários admitidos entre as datas listadas na cláusula WHERE serão mostrados em tela.

Fora da faixa estabelecida (NOT BETWEEN VALOR 1 AND VALOR 2).

Retorna todas as linhas que se encontram fora da faixa discriminada na cláusula WHERE.

Exemplo:

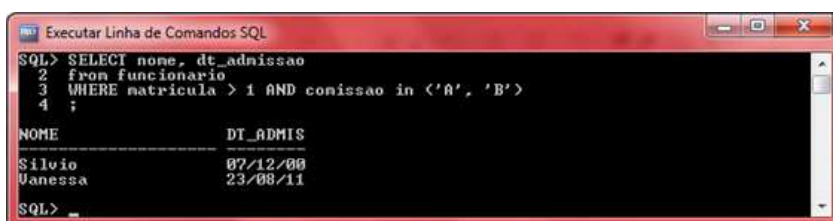
1. `SELECT nome, salario from funcionario WHERE matricula NOT BETWEEN '101' AND '150'.`

Igual a um dos valores da lista (IN (lista)):

Usando o comparador IN na cláusula WHERE, podemos determinar valores de comparação.

Exemplo:

1. `SELECT nome, dt_admissao from funcionario WHERE matricula > 1 AND comissao in ('A', 'B').`



```
SQL> SELECT nome, dt_admissao
2 from funcionario
3 WHERE matricula > 1 AND comissao in ('A', 'B')
4 ;

NOME          DT_ADMIS
-----
Silvio        07/12/00
Vanessa       23/08/11

SQL>
```

Todos os funcionários com matrícula maior que 1 e comissão igual a A ou igual a B serão mostrados.

Diferente dos valores da lista (NOT IN (lista)):

Retorna as linhas que possuem valores diferentes dos valores descritos na lista.

Exemplo:

1. `SELECT * from funcionario WHERE matricula != 1 AND comissao not in ('A').`

Todos os dados da tabela funcionário em que o valor da matrícula for diferente de 1 e o valor da comissão não for A serão listados.

Pesquisa de caracteres (LIKE)

Para mostrar linhas que possuem parte dos valores comuns, é utilizado o operador LIKE. Esse operador pode ter um dos dois símbolos listados abaixo:

% (percentual) – Qualquer número de caracteres

_ (underline) – Um único caractere

Exemplo:

1. `SELECT * FROM funcionario WHERE nome LIKE 'V%'.`



Cria um relatório onde a condição de exibição é o nome do funcionário obrigatoriamente iniciar coma letra A e não importa a quantidade de caracteres que venham depois. A localização deste símbolo pode ser no início, meio ou fim da cadeia de caracteres, exemplo: %a, %a%, C%a.

Podemos utilizar também em datas:

Exemplo:

1. `SELECT * FROM funcionario WHERE dt_admissao LIKE '%/10/10';`

O SQL também aceita apenas os dois últimos caracteres do ano da data.

Todos os funcionários com data de admissão em outubro de 2010 serão listados.

O exemplo abaixo utiliza o outro símbolo (_).

Listar os funcionários cujo nome comece com a letra 'S', termine com a letra 'o' e possua 5 letras.

Exemplo:

1. `SELECT nome FROM funcionario WHERE nome LIKE 'S____o';`

Obs: podemos utilizar os dois símbolos (%) e _ em uma mesma comparação.

Exemplo:

1. `SELECT nome FROM funcionario WHERE nome LIKE '_i%o';`



Essa instrução procura na coluna ?nome? o conteúdo que a segunda posição é a letra ?i? e que a última é a letra ?o?, não importando a quantidade de caracteres que exista entre ?i? e ?o?.

Referências

BEIGHLEY, Lynn. *Use a cabeça SQL*, Alta Books.

FANDERUFF, Damaris. *Dominando o Oracle 9i: modelagem e desenvolvimento*, São Paulo, Makron.

GRAVES, Mark. *Projeto de banco de dados com XML*. 1.ed. São Paulo: Pearson, 2003.

MORELLI, Eduardo Terra. *Oracle 9i Fundamental: SQL, PL/SQL e administração*, São Paulo: Érica.

PRICE, JASON. *Oracle Database 11g SQL/Jason Price*. Tradução de João Eduardo Nóbrega Tortello. Porto Alegre: Bookman, 2009.

SILVA, Robson S., Oracle Database 10g express edition. 1. ed. São Paulo: Érica, 2007.



Avalie este tópico



ANTERIOR

Criação de relatórios utilizando filtros, operadores relacionais e operadores lógicos

Biblioteca

(<https://www.uninove.br/conhec-a->

a-



Índice

Ajuda?

PRÓXIMO: (<https://ava.un>

oper

Criação de relatórios utilizando filtros, operadores e funções em banco de dados

| [uninove/biblioteca/sobre-a-biblioteca/apresentacao/](#)
[Portal Uninove](#)
<http://www.uninove.br>
[Mapa do Site](#)

