

BANCO DE DADOS

Predicados de Procura

Predicados Relacionais

São aqueles que relacionam duas expressões segundo um operador. Os seguintes operadores são suportados:

Operador	Significado
=	Igual
!=	Diferente
<>	Diferente
>	Maior
!>	Não maior (menor ou igual)
<	Menor
!<	Não menor (maior ou igual)
>=	Maior ou igual
<=	Menor ou igual

Um predicado relacional segue a seguinte sintaxe: <expressão> <operador> <expressão>. Logo abaixo encontra-se um exemplo utilizando operadores relacionais.

Neste exemplo, queremos obter todos os quartos cujo desconto é maior que 10%.

```
SELECT * FROM quarto WHERE desconto >= 0.10
```

Predicado BETWEEN

Compara um valor com uma faixa de valores. Os limites da faixa estão inclusos. Possui a seguinte sintaxe: <expressão> | not | between <expressão> and <expressão>.

No exemplo seguinte recuperamos todas os quartos, cujo desconto está entre 10% e 20%:

```
SELECT nome, quarto FROM quarto WHERE desconto BETWEEN 0.10 AND 0.20
```

Predicado NULL

Utilizado para testar os valores nulos. Verifica, por exemplo, se colunas não contém nenhum valor armazenado. Possui a seguinte sintaxe: <nome da coluna> is | not | NULL. Caso seja necessário conhecer os quartos não possuem qualquer desconto, utilizamos a seguinte instrução:

```
SELECT nome, quarto FROM quarto WHERE desconto IS NULL
```

É possível utilizar o operador not juntamente com a palavra-chave is (do modo, is not NULL) para, por exemplo, montar condições de colunas que não possuem valores nulos.

Predicado LIKE

O predicado LIKE procura por strings que se encontram dentro de um determinado padrão. O predicado LIKE só pode ser usado com tipos de dados CHAR e VARCHAR.

A sintaxe para o predicado é seguinte:

<nome da coluna> | not | like < padrão de pesquisa >

O padrão de pesquisa trata-se de uma string a ser comparada com a coluna especificada. Este padrão pode ser formatado com a utilização de coringas, permitindo a procura de substrings. Os dois caracteres coringas existentes são o símbolo porcentagem '%' e o underscore '_'.

%	Equivale a zero ou mais caracteres
_	Equivale a um caractere qualquer

Os exemplos abaixo mostram a utilização do predicado LIKE:

- Verificar os hóspedes que possuem o final de sua descrição a palavra "EL" (p.ex. MIGUEL).

```
SELECT * FROM hospede WHERE NOME LIKE '%EL'
```

- Verificar os hóspedes que possuem o início de sua descrição a palavra "MIG" (p.ex. MIGUEL).

```
SELECT * FROM hospede WHERE NOME LIKE 'MIG%'
```

- Verificar os hóspedes que possuem em sua descrição a palavra "GU" (p.ex. MIGUEL).

```
SELECT * FROM departamento WHERE localidade LIKE '%GU%'
```

Predicado EXISTS

É utilizado para testar se o conjunto de linhas resultantes de uma consulta é vazio. Sintaxe:

| not | exists < subselect >

O exemplo seguinte mostra a utilização do predicado exists. Neste caso desejamos todos os nomes dos hóspedes que tenham pelo menos uma reserva. Para fazermos isto, utilizamos a seguinte lógica: para cada Hóspede, pesquisamos em Quarto se reservou algo.

```
select NOME from HOSPEDE where exists
      (select * from QUARTO where HOSPEDE.NOME = QUARTO.NOME)
```

Predicado IN

O predicado in compara um valor com um conjunto de valores. Este conjunto de valores pode ser uma lista ou o resultado de um subselect. Sintaxe:

<expressão> | not | in < subselect >
ou
<expressão> | not | in < lista de valores >

Os seguintes exemplos demonstram o uso do predicado in:

- Selecionar todos os quartos que não possuem os descontos de 10% e 20%

```
select * from QUARTO where DESCONTO not in (10,20)
```

- Selecionar todos os quartos reservados pelos hóspedes que possuem a altura superior a 2 metros.

```
select * from QUARTO where NOME in
      (select NOME from HOSPEDE where ALTURA > 2)
```

União

Cria uma consulta união, que combina os resultados de duas ou mais consultas ou tabelas independentes, é importante que ambas consultas retornem os mesmos tipos de dados na sequência.

Sintaxe: [TABLE] consulta1 UNION [ALL] [TABLE] consulta2 [UNION [ALL] [TABLE] consultaN [...]]

Podemos mesclar os resultados de duas ou mais consultas, tabelas e instruções SELECT, em qualquer combinação, em uma única operação UNION.