

FIAP GRADUAÇÃO

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

DATABASE MODELING & SQL

Profa. Rita de Cássia Rodrigues



rita@fiap.com.br

Prof. William Maximiano



profwilliam.junior@fiap.com.br

Comandos DQL * DRS: Linguagem para Consulta e Recuperação de Dados
OPERADORES * EXPRESSÕES * CLÁUSULAS

- Objetivo
- Conceitos referentes a linguagem de manipulação de dados
- Revisão dos Conceitos
- Exercícios

- ❑ Aplicar os conceitos da linguagem SQL durante a implementação do banco de dados

☐ Linguagem de definição de dados

☐ DRS/DQL (SELECT)

☐ Operadores

☐ Expressões

☐ Cláusulas

☐ Exercícios

A principal função de um sistema de banco de dados é proporcionar recursos para armazenamento, manutenção e consulta dos dados, de maneira segura e eficaz;

A **consulta de dados é uma pesquisa** realizada **no banco** de dados, cujo propósito é recuperar as linhas (registros) **que atendem** às **condições pré-estabelecidas**. As consultas podem envolver diferentes recursos e, por isso, apresentam as seguintes definições:

- Consultas simples;
- Consultas baseadas em condições simples ou compostas;
- Consultas que recuperam dados de diferentes tabelas (junções);
- Consultas que utilizam funções para realização de diferentes operações;
- Consultas que manipulam grupos de dados;
- Consultas baseadas no resultado de outras consultas.

A instrução utilizada para a realização das consultas é o **SELECT**, que faz parte da categoria de operações DML (Data Manipulation Language);

Alguns autores criam um grupo a parte para tratar somente do SELECT, o **DQL**, Data Query Language (**Linguagem de Consulta de Dados**) ou **DRS**, Data Retrieve Language (**Linguagem de Recuperação de Dados**).

SINTAXE:

```
SELECT [ DISTINCT | ALL ] { * | coluna [, coluna, ... ] }  
FROM tabela  
WHERE condição  
ORDER BY coluna [, coluna, ... ]
```



Onde	Descrição
SELECT	Especifica as colunas (campos) desejadas na pesquisa.
DISTINCT	Não mostra eventuais valores repetidos de colunas.
ALL	Mostra todos os valores, mesmo que repetidos. Esse é o padrão se o DISTINCT não for definido. Portanto, não precisamos escrevê-lo.
*	Indica que devem ser mostradas todas as colunas da tabela
FROM	Determina em que tabelas serão efetuadas estas pesquisas.
WHERE	Condição para que se execute a pesquisa (filtra dados).
ORDER BY	Especifica em que ordem deverá ser apresentada a pesquisa desejada – por qual campo estará ordenada (de forma crescente ou decrescente).

CONSULTA SIMPLES

Possibilita a **recuperação de todas as linhas e colunas de uma tabela**; a seleção de linhas que atendam aos **requisitos pré-estabelecidos**; e a definição das colunas, cujos dados serão selecionados.

Linguagem SQL




Consulta dos Dados utilizando a linguagem SQL

Selecionando todas as linhas e colunas (campos) da tabela FUNCIONARIO.

```
-- CONSULTANDO TODAS AS LINHAS E COLUNAS DA TABELA FUNCIONARIO
SELECT * FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

Resultado da Consulta x

 SQL | Todas as Linhas Extraídas: 15 em 0,605 segundos

	NR_MATRICULA	CD_DEPTO	NM_FUNCIONARIO	DT_NASCIMENTO	DT_ADMISSAO	DS_ENDERECO	VL_SALARIO
1	12345	1	JOAO DA SILVA	10/05/85	15/09/12	RUA X, 49	5684,66
2	12346	1	MANUEL DA SILVA	05/10/98	10/11/15	RUA X, 31	3542,11
3	12347	1	JANDIRA DA SILVA	10/12/00	15/09/18	RUA X, 25	1875,96
4	12348	2	KATIA REGINA SOUZA	15/01/95	03/10/15	RUA Y, 49	3894,63
5	12349	2	MARIA DAS DORES SOUZA	18/08/83	23/10/17	RUA Y, 35	1542,55
6	12350	2	ALFREDO DE SOUZA	04/05/99	05/06/14	RUA Y, 27	5874,52
7	12351	3	GISELE DE JESUS	15/04/99	20/03/17	RUA Z, 49	1060,66
8	12352	3	RAFAEL DE JESUS	10/08/98	10/08/12	RUA Z, 55	2563,44
9	12353	3	ROSANA DE JESUS	14/03/87	15/08/19	RUA Z, 79	4879,55
10	12354	4	JOSEFINA DE ALMEIDA	16/10/97	25/03/13	RUA Y, 33	4561,88
11	12355	4	LUCIANA DE ALMEIDA	10/02/84	28/09/11	RUA Y, 44	2345,52
12	12356	4	THIAGO DE ALMEIDA	10/03/98	24/10/18	RUA Y, 55	1254,22
13	12357	5	LARISSA DE CAMARGO	14/02/97	04/08/15	RUA V, 22	1245,55
14	12358	5	ANTONIO DE CAMARGO	25/01/85	12/08/16	RUA V, 44	2451,33
15	12359	5	JOSE DE CAMARGO	23/10/98	20/04/17	RUA V, 88	6541,22

Atenção: O comando acima retornará todas as colunas (*) e linhas da tabela funcionário.

Obs.: Não é uma boa prática utilizar a cláusula `Select *`, pois todas as colunas são retornadas, independentemente das colunas que serão utilizadas realmente, e isso compromete o desempenho do ambiente, pois no conceito correto, somente o que é necessário selecionar deve ser solicitado.

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



FIAP

Especificando as colunas que deseja consultar e recuperando todas as linhas da tabela FUNCIONARIO.

```
-- Especificando as colunas que deseja consultar e recuperando todas as linhas da tabela FUNCIONARIO
SELECT NR_MATRICULA, NM_FUNCIONARIO, VL_SALARIO
FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	VL_SALARIO
12345	JOAO DA SILVA	5684,66
12346	MANUEL DA SILVA	3542,11
12347	JANDIRA DA SILVA	1875,96
12348	KATIA REGINA SOUZA	3894,63
12349	MARIA DAS DORES SOUZA	1542,55
12350	ALFREDO DE SOUZA	5874,52
12351	GISELE DE JESUS	1060,66
12352	RAFAEL DE JESUS	2563,44
12353	ROSANA DE JESUS	4879,55
12354	JOSEFINA DE ALMEIDA	4561,88
12355	LUCIANA DE ALMEIDA	2345,52
12356	THIAGO DE ALMEIDA	1254,22
12357	LARISSA DE CAMARGO	1245,55
12358	ANTONIO DE CAMARGO	2451,33
12359	JOSE DE CAMARGO	6541,22

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



Especificando as colunas que deseja consultar e filtrando as linhas da tabela FUNCIONARIO.

```
-- Especificando as colunas que deseja consultar e filtrando as linhas da tabela FUNCIONARIO.  
SELECT NR_MATRICULA, CD_DEPTO, NM_FUNCIONARIO, VL_SALARIO  
  FROM T_SIP_FUNCIONARIO  
 WHERE CD_DEPTO = 3;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	CD_DEPTO	NM_FUNCIONARIO	VL_SALARIO
12351	3	GISELE DE JESUS	1060,66
12352	3	RAFAEL DE JESUS	2563,44
12353	3	ROSANA DE JESUS	4879,55

Atenção: O comando acima retornará apenas as colunas especificadas na consulta e, em relação as linhas, somente aquelas que satisfaçam a condição especificada.

Dê preferência a consultas onde as colunas e filtro são especificados, com isso evitamos retornar informações que não serão utilizadas pelo usuário.

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



Exemplo utilizando DISTINCT (OU UNIQUE)

A **cláusula** DISTINCT é **utilizada dentro** de um comando **SELECT**;

Serve para listar os **registros** de uma busca **sem duplica-los**;

Não funciona em campos do tipo LOB.

```
-- RECUPERANDO OS DEPARTAMENTOS (SEM REPETIÇÃO) QUE POSSUEM FUNCIONARIOS.  
SELECT DISTINCT CD_DEPTO FROM T_SIP_FUNCIONARIO;  
-- OU  
SELECT UNIQUE CD_DEPTO FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

CD_DEPTO
1
2
4
5
3

Consulta dos Dados utilizando a linguagem SQL



Operadores Aritméticos

Os **operadores aritméticos** também **podem ser utilizados** nas cláusulas de **SQL**;

Sempre que houver **mais de um operador**, a **precedência matemática** será respeitada, ou seja, a multiplicação e a divisão serão avaliadas em primeiro lugar e, o uso do parênteses pode alterar a ordem de execução dos operadores.

OPERADOR	DESCRIÇÃO
+	Adição
-	Subtração
*	Multiplicação
/	Divisão

Consulta dos Dados utilizando a linguagem SQL



Exemplo de consulta utilizando operadores aritméticos

```
-- Exemplo de consulta utilizando operadores aritméticos
SELECT NR_MATRICULA, NM_FUNCIONARIO, VL_SALARIO * 1.05
FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	VL_SALARIO*1.05
12345	JOAO DA SILVA	5968,893
12346	MANUEL DA SILVA	3719,2155
12347	JANDIRA DA SILVA	1969,758
12348	KATIA REGINA SOUZA	4089,3615
12349	MARIA DAS DORES SOUZA	1619,6775
12350	ALFREDO DE SOUZA	6168,246
12351	GISELE DE JESUS	1113,693
12352	RAFAEL DE JESUS	2691,612
12353	ROSANA DE JESUS	5123,5275
12354	JOSEFINA DE ALMEIDA	4789,974
12355	LUCIANA DE ALMEIDA	2462,796
12356	THIAGO DE ALMEIDA	1316,931
12357	LARISSA DE CAMARGO	1307,8275
12358	ANTONIO DE CAMARGO	2573,8965
12359	JOSE DE CAMARGO	6868,281

Consulta dos Dados utilizando a linguagem SQL



Exemplo de consulta utilizando operadores aritméticos

```
SELECT NR_MATRICULA, NM_FUNCIONARIO, (VL_SALARIO + 120 ) * 1.10  
FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	(VL_SALARIO+120)*1.10
12345	JOAO DA SILVA	6385,126
12346	MANUEL DA SILVA	4028,321
12347	JANDIRA DA SILVA	2195,556
12348	KATIA REGINA SOUZA	4416,093
12349	MARIA DAS DORES SOUZA	1828,805
12350	ALFREDO DE SOUZA	6593,972
12351	GISELE DE JESUS	1298,726
12352	RAFAEL DE JESUS	2951,784
12353	ROSANA DE JESUS	5499,505
12354	JOSEFINA DE ALMEIDA	5150,068
12355	LUCIANA DE ALMEIDA	2712,072
12356	THIAGO DE ALMEIDA	1511,642
12357	LARISSA DE CAMARGO	1502,105
12358	ANTONIO DE CAMARGO	2828,463
12359	JOSE DE CAMARGO	7327,342

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



ALIAS: APELIDO PARA COLUNAS

Às vezes, ao retornar dados de um select, o nome da coluna (campo) não é muito claro ou apropriado para a apresentação dos dados.

Os apelidos são úteis para melhorar a legibilidade do resultado da consulta, facilitando o entendimento por parte do usuário.

Por isso, podemos usar um “apelido” para a coluna, que chamamos de “ALIAS”.

É possível renomear o cabeçalho da coluna. Muito útil quando queremos retornar um nome adequado para o resultado de um cálculo.

Deve ser escrito após o nome da coluna/cálculo, se for uma palavra composta, ou tenha caracteres especiais (acentos, por exemplo) ou que faça distinção entre maiúsculas e minúsculas, deve se escritos entre aspas dupla.

Não deve ser uma palavra reservada (SELECT, WHERE, etc);

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ALIAS



ALIAS (APELIDO) PARA COLUNAS – EXEMPLO

```
-- APELIDOS DE COLUNAS
SELECT NR_MATRICULA, NM_FUNCIONARIO, VL_SALARIO * 1.05 AS "VALOR AJUSTADO"
FROM T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	VALOR AJUSTADO
12345	JOAO DA SILVA	5968,893
12346	MANUEL DA SILVA	3719,2155
12347	JANDIRA DA SILVA	1969,758
12348	KATIA REGINA SOUZA	4089,3615
12349	MARIA DAS DORES SOUZA	1619,6775
12350	ALFREDO DE SOUZA	6168,246
12351	GISELE DE JESUS	1113,693
12352	RAFAEL DE JESUS	2691,612
12353	ROSANA DE JESUS	5123,5275
12354	JOSEFINA DE ALMEIDA	4789,974
12355	LUCIANA DE ALMEIDA	2462,796
12356	THIAGO DE ALMEIDA	1316,931
12357	LARISSA DE CAMARGO	1307,8275
12358	ANTONIO DE CAMARGO	2573,8965
12359	JOSE DE CAMARGO	6868,281

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – CLAUSULA ORDER BY

O resultado de uma consulta, pode ser apresentado de forma ordenada, utilizando a clausula ORDER BY.

Esta clausula vem por último na declaração SELECT e pode ordenar o resultado em ordem ascendente (ASC, que é o default) e descendente (DESC).

Podemos ordenar por mais de uma coluna.

A cláusula ORDER BY aceita números representando a coluna na ordem em que foi declarada na cláusula SELECT, além de aceitar o alias dado a alguma coluna.

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – CLAUSULA ORDER BY

EXEMPLOS utilizando a clausula ORDER BY

```
-- APRESENTANDO RESULTADOS DE FORMA ORDENADA
-- ORDER BY ASC
SELECT      CD_DEPTO, NM_DEPTO
FROM        T_SIP_DEPTO
ORDER BY    NM_DEPTO;
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
3	CONTAS A PAGAR
4	FATURAMENTO
1	FINANCEIRO
5	RECURSOS HUMANOS
2	TECNOLOGIA DA INFORMAÇÃO

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – CLAUSULA ORDER BY

EXEMPLOS utilizando a clausula ORDER BY

```
-- ORDER BY DESC
SELECT      CD_DEPTO, NM_DEPTO
FROM        T_SIP_DEPTO
ORDER BY    NM_DEPTO DESC;
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
2	TECNOLOGIA DA INFORMAÇÃO
5	RECURSOS HUMANOS
1	FINANCEIRO
4	FATURAMENTO
3	CONTAS A PAGAR

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – CLAUSULA ORDER BY

EXEMPLOS utilizando a clausula **ORDER BY**. Utilizando os números que representam as colunas

```
-- ORDER BY UTILIZANDO NÚMEROS QUE REPRESENTAM AS COLUNAS
SELECT      NR_MATRICULA, CD_DEPTO, NM_FUNCIONARIO, VL_SALARIO
FROM        T_SIP_FUNCIONARIO
WHERE       CD_DEPTO >1
ORDER BY    2 ASC, 3 DESC;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	CD_DEPTO	NM_FUNCIONARIO	VL_SALARIO
12349	2	MARIA DAS DORES SOUZA	1542,55
12348	2	KATIA REGINA SOUZA	3894,63
12350	2	ALFREDO DE SOUZA	5874,52
12353	3	ROSANA DE JESUS	4879,55
12352	3	RAFAEL DE JESUS	2563,44
12351	3	GISELE DE JESUS	1060,66
12356	4	THIAGO DE ALMEIDA	1254,22
12355	4	LUCIANA DE ALMEIDA	2345,52
12354	4	JOSEFINA DE ALMEIDA	4561,88
12357	5	LARISSA DE CAMARGO	1245,55
12359	5	JOSE DE CAMARGO	6541,22
12358	5	ANTONIO DE CAMARGO	2451,33

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – CLAUSULA ORDER BY

EXEMPLOS utilizando a clausula **ORDER BY**. Utilizando o apelido das colunas

```
-- ORDER BY UTILIZANDO O APELIDO DAS COLUNAS
SELECT      NR_MATRICULA           "MATRICULA",
            NM_FUNCIONARIO         "NOME ",
            VL_SALARIO             "SALÁRIO MENSAL"
FROM        T_SIP_FUNCIONARIO
ORDER BY    "SALÁRIO MENSAL" DESC;
```

Resultado obtido após a execução do comando:

MATRICULA	NOME	SALÁRIO MENSAL
12359	JOSE DE CAMARGO	6541,22
12350	ALFREDO DE SOUZA	5874,52
12345	JOAO DA SILVA	5684,66
12353	ROSANA DE JESUS	4879,55
12354	JOSEFINA DE ALMEIDA	4561,88
12348	KATIA REGINA SOUZA	3894,63
12346	MANUEL DA SILVA	3542,11
12352	RAFAEL DE JESUS	2563,44
12358	ANTONIO DE CAMARGO	2451,33
12355	LUCIANA DE ALMEIDA	2345,52
12347	JANDIRA DA SILVA	1875,96
12349	MARIA DAS DORES SOUZA	1542,55
12356	THIAGO DE ALMEIDA	1254,22
12357	LARISSA DE CAMARGO	1245,55
12351	GISELE DE JESUS	1060,66

I OPERADORES DE COMPARAÇÃO

Consulta dos Dados utilizando a linguagem SQL



Pesquisa em Tabelas utilizando FILTROS, a partir do uso de operadores

Podem ser utilizados para estabelecer uma relação de comparação entre valores ou expressões, resultando sempre em um valor lógico (booleano), sendo verdadeiro ou falso;

Os operadores podem ser utilizados em consultas, atualizações ou exclusões de dados.

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Relacionais



OPERADOR	SIGNIFICADO	EXEMPLO
=	Igual	codigo=2
<	Menor que	preco<10
<=	Menor ou igual a	preco<=10
>	Maior que	preco>10
>=	Maior ou igual a	preco>=10
<> Ou !=	diferente	codigo<>2

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Lógicos

OPERADOR	DESCRIÇÃO
AND	Retorna TRUE se ambas as condições forem verdadeiras
OR	Retorna TRUE se ao menos uma das condições for verdadeira
NOT	Retorna TRUE se a condição for falsa

Atenção no momento de montar uma condição composta, pois existe ordem de precedência:

- ✓ NOT, AND e OR, que pode ser alterada com a inclusão de parênteses.

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Especiais

OPERADOR	DESCRIÇÃO
BETWEEN... AND ...	Entre dois valores (inclusive)
IN (<i>lista</i>)	Compara o valor de uma coluna com um conjunto de valores.
LIKE <i>valor</i>	Compara cadeia de caracteres
IS NULL / IS NOT NULL	É um valor nulo. Não nulo.

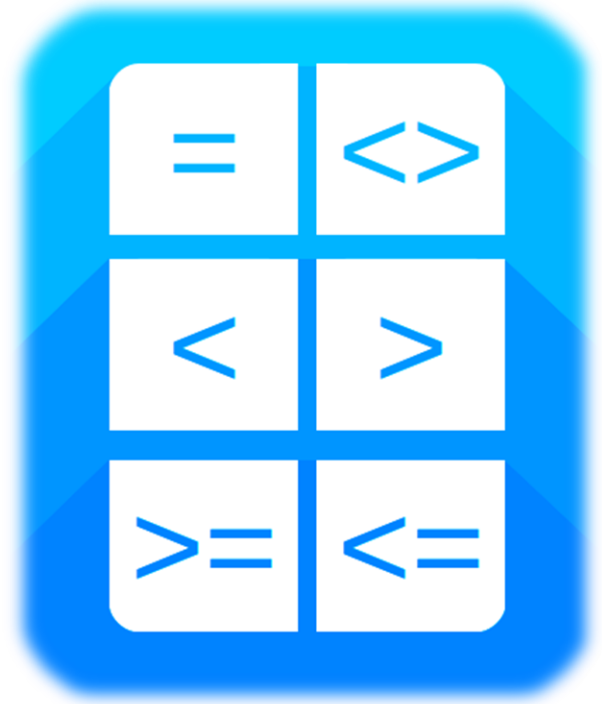


Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores (Resumo)

OPERADOR	DESCRIÇÃO	EXEMPLO
=	Igualdade	VALOR = 100
<>	Diferente	VALOR <> 100
>	Maior do que	VALOR > 100
<	Menor do que	VALOR < 100
>=	Maior ou igual a	VALOR >= 100
<=	Menor ou igual a	VALOR <= 100
Between valor_1 and valor_2	Entre valor_1 e valor_2	VALOR BETWEEN 80 AND 100
not Between valor_1 and valor_2	Não está entre valor_1 e valor_1	VALOR NOT BETWEEN 80 AND 100
In (conjunto de valores)	Igual a um dos valores do conjunto	VALOR IN (70, 100, 120)
not In (conjunto de valores)	Não é igual a um dos valores	VALOR NOT IN (70, 100, 120)
Is Null	Nulo	VALOR IS NULL
Is Not Null	Não é Nulo	VALOR IS NOT NULL
Like (expressão)	Contém a expressão	PRODUTO LIKE ('CA%')
not Like (expressão)	Não contém as expressão	PRODUTO NOT LIKE ('CA%')

OPERADORES RELACIONAIS



Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Relacionais

EXEMPLOS utilizando operadores RELACIONAIS

```
-- FILTRANDO POR NÚMEROS
SELECT NR_MATRICULA, NM_FUNCIONARIO, VL_SALARIO
FROM T_SIP_FUNCIONARIO
WHERE VL_SALARIO > 1500.50;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	VL_SALARIO
12345	JOAO DA SILVA	5684,66
12346	MANUEL DA SILVA	3542,11
12347	JANDIRA DA SILVA	1875,96
12348	KATIA REGINA SOUZA	3894,63
12349	MARIA DAS DORES SOUZA	1542,55
12350	ALFREDO DE SOUZA	5874,52
12352	RAFAEL DE JESUS	2563,44
12353	ROSANA DE JESUS	4879,55
12354	JOSEFINA DE ALMEIDA	4561,88
12355	LUCIANA DE ALMEIDA	2345,52
12358	ANTONIO DE CAMARGO	2451,33
12359	JOSE DE CAMARGO	6541,22

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Relacionais

EXEMPLOS utilizando operadores RELACIONAIS

```
-- FILTRANDO POR TEXTO
SELECT NR_MATRICULA, NM_FUNCIONARIO
FROM T_SIP_FUNCIONARIO
WHERE NM_FUNCIONARIO='JOAO DA SILVA';
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO
12345	JOAO DA SILVA

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Relacionais



EXEMPLOS utilizando operadores RELACIONAIS

```
-- FILTRANDO POR DATA
SELECT NR_MATRICULA, NM_FUNCIONARIO, DT_ADMISSAO
FROM T_SIP_FUNCIONARIO
WHERE DT_ADMISSAO > TO_DATE('01/08/2015', 'DD/MM/YYYY');
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	DT_ADMISSAO
12346	MANUEL DA SILVA	10/11/15
12347	JANDIRA DA SILVA	15/09/18
12348	KATIA REGINA SOUZA	03/10/15
12349	MARIA DAS DORES SOUZA	23/10/17
12351	GISELE DE JESUS	20/03/17
12353	ROSANA DE JESUS	15/08/19
12356	THIAGO DE ALMEIDA	24/10/18
12357	LARISSA DE CAMARGO	04/08/15
12358	ANTONIO DE CAMARGO	12/08/16
12359	JOSE DE CAMARGO	20/04/17

OPERADORES LÓGICOS

OPERADOR
AND
OR
NOT

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Lógicos



EXEMPLOS utilizando operadores LÓGICOS

```
-- OPERADOR AND
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO, DT_ADMISSAO
FROM T_SIP_FUNCIONARIO
WHERE DT_ADMISSAO > TO_DATE('01/08/2010', 'DD/MM/YYYY') AND
      CD_DEPTO=3;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO	DT_ADMISSAO
12351	GISELE DE JESUS	3	20/03/17
12352	RAFAEL DE JESUS	3	10/08/12
12353	ROSANA DE JESUS	3	15/08/19

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Lógicos



EXEMPLOS utilizando operadores LÓGICOS

```
-- OPERADOR OR
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO, VL_SALARIO
FROM T_SIP_FUNCIONARIO
WHERE CD_DEPTO =4 OR VL_SALARIO>3000;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO	VL_SALARIO
12345	JOAO DA SILVA	1	5684,66
12346	MANUEL DA SILVA	1	3542,11
12348	KATIA REGINA SOUZA	2	3894,63
12350	ALFREDO DE SOUZA	2	5874,52
12353	ROSANA DE JESUS	3	4879,55
12354	JOSEFINA DE ALMEIDA	4	4561,88
12355	LUCIANA DE ALMEIDA	4	2345,52
12356	THIAGO DE ALMEIDA	4	1254,22
12359	JOSE DE CAMARGO	5	6541,22

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Lógicos



EXEMPLOS utilizando operadores LÓGICOS

```
-- OPERADOR NOT
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO, VL_SALARIO
FROM T_SIP_FUNCIONARIO
WHERE NOT CD_DEPTO =4;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO	VL_SALARIO
12345	JOAO DA SILVA	1	5684,66
12346	MANUEL DA SILVA	1	3542,11
12347	JANDIRA DA SILVA	1	1875,96
12348	KATIA REGINA SOUZA	2	3894,63
12349	MARIA DAS DORES SOUZA	2	1542,55
12350	ALFREDO DE SOUZA	2	5874,52
12351	GISELE DE JESUS	3	1060,66
12352	RAFAEL DE JESUS	3	2563,44
12353	ROSANA DE JESUS	3	4879,55
12357	LARISSA DE CAMARGO	5	1245,55
12358	ANTONIO DE CAMARGO	5	2451,33
12359	JOSE DE CAMARGO	5	6541,22

OPERADORES ESPECIAIS



Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- EXEMPLOS utilizando operadores ESPECIAIS
-- OPERADOR BETWEEN

SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO, VL_SALARIO
FROM   T_SIP_FUNCIONARIO
WHERE  VL_SALARIO BETWEEN 1000 AND 2500;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO	VL_SALARIO
12347	JANDIRA DA SILVA	1	1875,96
12349	MARIA DAS DORES SOUZA	2	1542,55
12351	GISELE DE JESUS	3	1060,66
12355	LUCIANA DE ALMEIDA	4	2345,52
12356	THIAGO DE ALMEIDA	4	1254,22
12357	LARISSA DE CAMARGO	5	1245,55
12358	ANTONIO DE CAMARGO	5	2451,33

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Especiais

EXEMPLOS utilizando operadores ESPECIAIS

```
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO, DT_ADMISSAO
FROM T_SIP_FUNCIONARIO
WHERE DT_ADMISSAO BETWEEN TO_DATE('01/01/2013','DD/MM/YYYY') AND
                           TO_DATE('30/12/2015','DD/MM/YYYY');
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO	DT_ADMISSAO
12346	MANUEL DA SILVA	1	10/11/15
12348	KATIA REGINA SOUZA	2	03/10/15
12350	ALFREDO DE SOUZA	2	05/06/14
12354	JOSEFINA DE ALMEIDA	4	25/03/13
12357	LARISSA DE CAMARGO	5	04/08/15

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR IN
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO
FROM T_SIP_FUNCIONARIO
WHERE CD_DEPTO IN (1,2,3);
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO
12345	JOAO DA SILVA	1
12346	MANUEL DA SILVA	1
12347	JANDIRA DA SILVA	1
12348	KATIA REGINA SOUZA	2
12349	MARIA DAS DORES SOUZA	2
12350	ALFREDO DE SOUZA	2
12351	GISELE DE JESUS	3
12352	RAFAEL DE JESUS	3
12353	ROSANA DE JESUS	3

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR NOT IN
SELECT NR_MATRICULA, NM_FUNCIONARIO, CD_DEPTO
FROM T_SIP_FUNCIONARIO
WHERE CD_DEPTO NOT IN (1,2,3);
```

Resultado obtido após a execução do comando:

NR_MATRICULA	NM_FUNCIONARIO	CD_DEPTO
12354	JOSEFINA DE ALMEIDA	4
12355	LUCIANA DE ALMEIDA	4
12356	THIAGO DE ALMEIDA	4
12357	LARISSA DE CAMARGO	5
12358	ANTONIO DE CAMARGO	5
12359	JOSE DE CAMARGO	5

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR IN
SELECT      NR_MATRICULA ,
            DT_ADMISSAO ,
            VL_SALARIO
FROM        T_SIP_FUNCIONARIO
WHERE       TO_CHAR(DT_ADMISSAO, 'YYYY') IN ('2012', '2014');
```

Resultado obtido após a execução do comando:

NR_MATRICULA	DT_ADMISSAO	VL_SALARIO
12345	15/09/12	5684,66
12350	05/06/14	5874,52
12352	10/08/12	2563,44

Nota: **TO_CHAR** (<conteudo>,<formato>): Converte qualquer conteúdo para texto.

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR IS NOT NULL
SELECT CD_PROJETO, NM_PROJETO, DT_INICIO, DT_TERMINO
FROM T_SIP_PROJETO
WHERE DT_TERMINO IS NOT NULL;
```

Resultado obtido após a execução do comando:

CD_PROJETO	NM_PROJETO	DT_INICIO	DT_TERMINO
10002	PROJETO 2	25/10/13	15/04/15
10005	PROJETO 5	28/09/14	15/09/18

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR IS NULL
SELECT CD_PROJETO, NM_PROJETO, DT_INICIO, DT_TERMINO
FROM T_SIP_PROJETO
WHERE DT_TERMINO IS NULL;
```

Resultado obtido após a execução do comando:

CD_PROJETO	NM_PROJETO	DT_INICIO	DT_TERMINO
2	APP MOBILE SAKSPILDAP	09/05/21	(null)
3	PORTAL SAKSPILDAP	20/03/21	(null)
4	BANCO DE DADOS SQLSERVER NO AZURE	14/06/21	(null)
5	ATUALIZACAO DO SISTEMA DE CLIENTES	15/09/21	(null)

Consulta dos Dados utilizando a linguagem SQL



Pesquisa Básica em Tabelas – Operadores Especiais (LIKE)

EXPRESSÃO	DESCRIÇÃO
LIKE 'A%'	Todas as palavras que iniciem com A.
LIKE '%A'	Todas as palavras que terminem com a letra A.
LIKE '%A%'	Todas as palavras que tenham a letra A em qualquer posição.
LIKE 'A_'	String de dois caracteres que tenham a primeira letra A e o segundo caractere seja qualquer outro.
LIKE '_A'	String de dois caracteres cujo primeiro caractere seja qualquer um e a última letra seja A.
LIKE '_A_'	String de três caracteres cuja segunda letra seja A independentemente do primeiro ou do último caractere.
LIKE '%A_'	Todas as palavras que tenham a letra A na penúltima posição e a última seja qualquer outro caractere.
LIKE '_A%'	Todos que tenham a letra A na segunda posição e o primeiro caractere seja qualquer um.

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR LIKE  
SELECT CD_DEPTO, NM_DEPTO  
FROM T_SIP_DEPTO  
WHERE NM_DEPTO LIKE 'FINA%';
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
1	FINANCEIRO

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR LIKE
SELECT CD_DEPTO, NM_DEPTO
FROM   T_SIP_DEPTO
WHERE  NM_DEPTO LIKE '_A%';
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
4	FATURAMENTO

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR LIKE
SELECT CD_DEPTO, NM_DEPTO
FROM   T_SIP_DEPTO
WHERE  NM_DEPTO LIKE '____UR%';
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
4	FATURAMENTO
5	RECURSOS HUMANOS

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – Operadores Especiais



EXEMPLOS utilizando operadores ESPECIAIS

```
-- OPERADOR LIKE  
SELECT CD_DEPTO, NM_DEPTO  
FROM T_SIP_DEPTO  
WHERE NM_DEPTO LIKE '%ME%';
```

Resultado obtido após a execução do comando:

CD_DEPTO	NM_DEPTO
4	FATURAMENTO

Consulta dos Dados utilizando a linguagem SQL



OPERADOR DE CONCATENAÇÃO (||)

Este operador permite a concatenação de colunas ou string de caracteres com outras colunas.

É representado por duas colunas verticais ||.

É criada uma coluna resultante que é uma expressão de caracteres.

Consulta dos Dados utilizando a linguagem SQL Pesquisa Básica em Tabelas – CONCATENAÇÃO (II)



CONSULTAS CONTENDO COLUNAS CONCATENADAS – EXEMPLO

```
-- EXIBIR O TEXTO: "O FUNCIONARIO <NOME> FOI
--   ADMITIDO EM <DATA ADMISSAO>"
SELECT      NR_MATRICULA ,
            'O FUNCIONARIO ' ||
            NM_FUNCIONARIO ||
            ', FOI ADMITIDO EM ' ||
            DT_ADMISSAO
FROM        T_SIP_FUNCIONARIO;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	'O FUNCIONARIO'	NM_FUNCIONARIO	'FOI ADMITIDO EM'	DT_ADMISSAO
123450	FUNCIONARIO	JOAO DA SILVA,	FOI ADMITIDO EM	15/09/12
123460	FUNCIONARIO	MANUEL DA SILVA,	FOI ADMITIDO EM	10/11/15
123470	FUNCIONARIO	JANDIRA DA SILVA,	FOI ADMITIDO EM	15/09/18
123480	FUNCIONARIO	KATIA REGINA SOUZA,	FOI ADMITIDO EM	03/10/15
123490	FUNCIONARIO	MARIA DAS DORES SOUZA,	FOI ADMITIDO EM	23/10/17
123500	FUNCIONARIO	ALFREDO DE SOUZA,	FOI ADMITIDO EM	05/06/14
123510	FUNCIONARIO	GISELE DE JESUS,	FOI ADMITIDO EM	20/03/17
123520	FUNCIONARIO	RAFAEL DE JESUS,	FOI ADMITIDO EM	10/08/12
123530	FUNCIONARIO	ROSANA DE JESUS,	FOI ADMITIDO EM	15/08/19
123540	FUNCIONARIO	JOSEFINA DE ALMEIDA,	FOI ADMITIDO EM	25/03/13
123550	FUNCIONARIO	LUCIANA DE ALMEIDA,	FOI ADMITIDO EM	28/09/11
123560	FUNCIONARIO	THIAGO DE ALMEIDA,	FOI ADMITIDO EM	24/10/18
123570	FUNCIONARIO	LARISSA DE CAMARGO,	FOI ADMITIDO EM	04/08/15
123580	FUNCIONARIO	ANTONIO DE CAMARGO,	FOI ADMITIDO EM	12/08/16
123590	FUNCIONARIO	JOSE DE CAMARGO,	FOI ADMITIDO EM	20/04/17

Consulta dos Dados utilizando a linguagem SQL



RESTRINGINDO A QUANTIDADE DE LINHAS DURANTE UMA CONSULTA

UTILIZANDO PSEUDOCOLUNA ROWNUM

Utilizada para limitar o retorno de uma consulta (em número de linhas).

É retornada uma amostra dos dados, podemos dizer que o resultado é um subconjunto do total de linhas a ser retornada.

Todas as tabelas possuem a pseudocoluna ROWNUM (retorna um número que indica a ordem em que o Oracle seleciona a linha a partir de uma tabela).

Podemos utilizar, por exemplo, para recuperar um certo número de linhas por vez, para realizar uma paginação em uma página WEB.

Consulta dos Dados utilizando a linguagem SQL Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

```
-- UTILIZANDO A PSEUDOCOLUNA ROWNUM  
SELECT ROWNUM ,  
       CD_DEPTO ,  
       NM_DEPTO  
FROM   T_SIP_DEPTO;
```

Resultado obtido após a execução do comando:

ROWNUM	CD_DEPTO	NM_DEPTO
1	1	FINANCEIRO
2	2	TECNOLOGIA DA INFORMAÇÃO
3	3	CONTAS A PAGAR
4	4	FATURAMENTO
5	5	RECURSOS HUMANOS

Consulta dos Dados utilizando a linguagem SQL Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

RESTRINGINDO A QUANTIDADE DE LINHAS RECUPERADAS DURANTE A

CONSULTA

```
-- RESTRINGINDO A QUANTIDADE DE LINHAS RECUPERADAS DURANTE A CONSULTA
SELECT      NR_MATRICULA ,
            CD_DEPTO ,
            DT_ADMISSAO ,
            VL_SALARIO ,
            (VL_SALARIO * 12) "SALÁRIO ANUAL"
FROM        T_SIP_FUNCIONARIO
WHERE       ROWNUM < 4;
```

Resultado obtido após a execução do comando:

NR_MATRICULA	CD_DEPTO	DT_ADMISSAO	VL_SALARIO	SALÁRIO ANUAL
12345	1	15/09/12	5684,66	68215,92
12346	1	10/11/15	3542,11	42505,32
12347	1	15/09/18	1875,96	22511,52

Consulta dos Dados utilizando a linguagem SQL



CONSULTAS UTILIZANDO ROWNUM (ANÁLISE TOP-N)

Esse tipo de consulta é útil em situações onde precisamos encontrar, por exemplo, os N maiores ou menores dentro de determinado grupo, os funcionários mais bem pagos, os que vendem mais, as mercadorias menos vendidas e assim por diante.

Para conseguirmos isso no Oracle precisaremos implementar uma consulta interna, que irá retornar todos os registros necessários, e na consulta externa iremos limitar a exibição num total desejado.

Este processo é conhecido como consulta TOP-N no Oracle.

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL

Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

Consulta principal que irá apresentar o “rank”, nome do funcionário e salário retornados pela subconsulta

```
-- PARA EXIBIR OS NOMES E OS SALÁRIOS DOS TRÊS FUNCIONÁRIOS
-- MAIS BEM REMUNERADOS DA TABELA FUNCIONARIO.
SELECT ROWNUM as RANK ,
        NM_FUNCIONARIO ,
        VL_SALARIO
FROM
    (
        SELECT NM_FUNCIONARIO ,
               VL_SALARIO
        FROM T_SIP_FUNCIONARIO
        ORDER BY VL_SALARIO DESC
    )
WHERE ROWNUM <= 3;
```

Subconsulta (subquery), que irá recuperar (consultar) todos os funcionários ordenados pelo salário em ordem decrescente

A pseudocoluna ROWNUM irá restringir a quantidade de linhas que serão apresentadas pela consulta principal

Linguagem SQL

Consulta dos Dados utilizando a linguagem SQL Pesquisa Básica em Tabelas – ROWNUM



CONSULTAS UTILIZANDO ROWNUM – EXEMPLO

```
-- PARA EXIBIR OS NOMES E OS SALÁRIOS DOS TRÊS FUNCIONÁRIOS
-- MAIS BEM REMUNERADOS DA TABELA FUNCIONARIO.
SELECT  ROWNUM as RANK ,
        NM_FUNCIONARIO ,
        VL_SALARIO
FROM
(
    SELECT NM_FUNCIONARIO ,
           VL_SALARIO
    FROM T_SIP_FUNCIONARIO
    ORDER BY VL_SALARIO DESC
)
WHERE ROWNUM <= 3;
```

Resultado obtido após a execução do comando:

RANK	NM_FUNCIONARIO	VL_SALARIO
1	JOSE DE CAMARGO	6541,22
2	ALFREDO DE SOUZA	5874,52
3	JOAO DA SILVA	5684,66

FORMATO DE DATA PARA EXIBIÇÃO

O padrão de exibição, de um campo do tipo data, no Oracle é:

- dia, mês e ano (com dois dígitos);

```
-- FORMATO DE DATA PARA EXIBIÇÃO  
SELECT CD_PROJETO, NM_PROJETO, DT_INICIO, DT_TERMINO  
FROM T_SIP_PROJETO;
```

CD_PROJETO	NM_PROJETO	DT_INICIO	DT_TERMINO
10001	PROJETO 1	11/10/16	(null)
10002	PROJETO 2	25/10/13	15/04/15
10003	PROJETO 3	28/04/12	(null)
10004	PROJETO 4	25/03/17	(null)
10005	PROJETO 5	28/09/14	15/09/18
10006	PROJETO 6	05/04/20	(null)



FORMATO DE DATA PARA EXIBIÇÃO



Para alterar o formato de exibição é utilizado a sintaxe:

```
-- Para alterar o formato de exibição é utilizado a sintaxe:  
-- ALTER SESSION SET NLS_DATE_FORMAT = formato_data;  
  
ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MM/YYYY';
```

```
-- FORMATO DE DATA PARA EXIBIÇÃO  
SELECT CD_PROJETO, NM_PROJETO, DT_INICIO, DT_TERMINO  
FROM T_SIP_PROJETO;
```

CD_PROJETO	NM_PROJETO	DT_INICIO	DT_TERMINO
10001	PROJETO 1	11/10/2016	(null)
10002	PROJETO 2	25/10/2013	15/04/2015
10003	PROJETO 3	28/04/2012	(null)
10004	PROJETO 4	25/03/2017	(null)
10005	PROJETO 5	28/09/2014	15/09/2018
10006	PROJETO 6	05/04/2020	(null)





VALORES DE formatação DE DATA

YY = ano com 2 dígitos;

YYYY = ano com 4 dígitos;

RR = ano com 2 dígitos, com determinação do século conforme regras;

RRRR = ano com 4 dígitos, para entrada com 2 dígitos haverá determinação do século;

MM = o número do mês;

MONTH = o nome completo do mês;

MON = o nome abreviado do mês (jan, fev, mar e etc);

DDD = o dia do ano;

DD = o dia do mês;

D = o dia da semana;

DAY = o nome do dia;

HH = a hora do dia, no formato de 12 horas;

HH24= a hora do dia, no formato 24 horas;

MI = os minutos;

SS = os segundos;

SS.FF ou SSXFF = segundos com frações de segundo.



FORMATO DE IDIOMA PARA EXIBIÇÃO

Para alterar o idioma de exibição da data é utilizado a sintaxe:

```
--Para alterar o idioma de exibição da data é utilizado a sintaxe:  
-- ALTER SESSION SET NLS_DATE_LANGUAGE = idioma;  
ALTER SESSION SET NLS_DATE_LANGUAGE = 'PORTUGUESE';
```

```
SELECT  SYSDATE,  
         TO_CHAR(SYSDATE, 'MON')    "MÊS ABREVIADO",  
         TO_CHAR(SYSDATE, 'MONTH')  "MÊS POR EXTENSO"  
FROM DUAL;
```

SYSDATE	MÊS ABREVIADO	MÊS POR EXTENSO
12/09/2020	SET	SETEMBRO





TABELAS PRÓPRIAS DO ORACLE

Para auxiliar o desenvolvimento e a organização dos dados, o SGBD Oracle oferece algumas tabelas próprias que são criadas de forma automáticas:

```
-- TABELAS PRÓPRIAS DO ORACLE  
SELECT * FROM USER_OBJECTS;  
SELECT * FROM USER_TAB_COLUMNS;  
SELECT * FROM USER_CONSTRAINTS;
```

Próxima aula estudaremos

- ❑ Revisão de conceitos através de exercícios

REFERÊNCIAS



- MACHADO, Felipe Nery R. Banco de Dados - Projeto e Implementação. Érica, 2004.
- Páginas: 330, 331.
- ELMASRI, R.; NAVATHE, S.B. Sistemas de Banco de Dados: Fundamentos e Aplicações. Pearson, 2005. Páginas: 153, 154.
- PRICE, JASON, ORACLE DATABASE 11 g – SQL Domine SQL e PL-SQL no banco de Dados Oracle, Bookman, 2008. Capítulos: 2, 5 e 6.
- **Outros:**
- Manual Oficial Oracle – Introdução ao Oracle 9i (SQL) - Oracle Corporation, 2000, 2001.

Copyright © 2022 Profa. Rita de Cássia Rodrigues

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).