



Unidade 3

Bancos de Dados Relacionais e Linguagem SQL

3.1 Comandos DML e Introdução às Funções

Comandos DML - Classificação de linhas

- A capacidade de classificar resultados é um recurso conveniente da linguagem SQL e permite aos programadores exibir informações de muitas maneiras diferentes.
- Para o design de banco de dados, as funções de negócios são ordenadas por entidades e atributos. Nas informações do banco de dados, a linguagem SQL usa a cláusula **ORDER BY**.

Comandos DML - Classificação de linhas

- A cláusula ORDER BY pode especificar várias maneiras de ordenar as linhas retornadas em uma consulta.
- A ordem de classificação padrão é crescente e assim os valores numéricos são exibidos do menor para o maior, os de data a partir do mais antigo e os de caractere são exibidos em ordem alfabética.
- Valores nulos são exibidos por último em ordem crescente e primeiro em ordem decrescente mas a ordem pode ser modificada utilizando-se, em alguns SGBDs, **NULLS FIRST** ou **NULLS LAST**.

Comandos DML - Classificação de linhas

```
SELECT *  
FROM sakila.staff  
ORDER BY picture;
```

Ordenando nulos

```
SELECT *  
FROM actor  
ORDER BY first_name ASC;
```

Ordem crescente

```
SELECT *  
FROM sakila.actor  
ORDER BY first_name DESC;
```

Ordem decrescente

```
SELECT *  
FROM sakila.actor  
ORDER BY last_name, first_name ASC;
```

Duas colunas de forma crescente

```
SELECT *  
FROM actor  
ORDER BY first_name ASC, last_name DESC ;
```

Ordenação distinta nas colunas

```
SELECT *  
FROM actor  
WHERE actor_id BETWEEN 150 AND 300  
ORDER BY first_name ASC, last_name DESC ;
```

Ordenação distinta nas colunas incluindo cláusula de seleção WHERE.

Ordem (Plano) de execução:

FROM: Localiza a tabela que contém os dados

WHERE: Restringe as linhas a serem retornadas

SELECT: seleciona colunas solicitadas no conjunto de dados reduzido.

ORDER BY: Ordena o conjunto de resultados.

Comandos DML - Funções

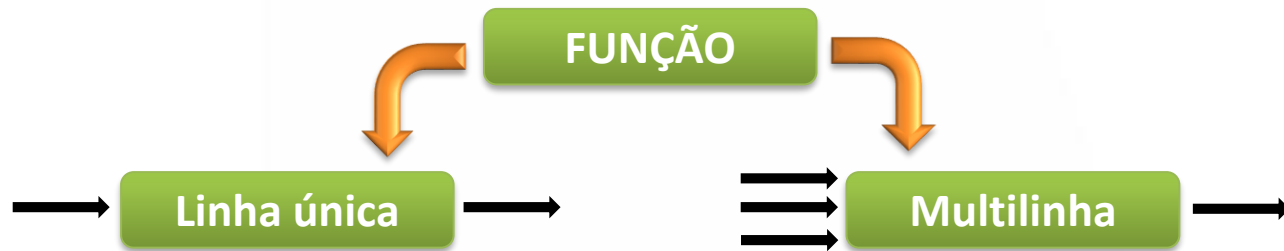
- Em SQL, há muitos tipos de funções que são usadas para transformar uma forma de entrada, e são usadas para manipulação e modificação destes dados em outra forma de saída.
- Na prática estas funções são como pequenos programas que executam uma ação em um valor ou coluna produzindo o resultado de saída do no formato esperado.

Comandos DML - Funções

- Em geral, o funcionamento das funções em SQL é o seguinte:



- Possuímos assim, dois tipos de retorno e por este motivo classificam-se as funções da seguinte forma:



Comandos DML - Funções de linha única

- **Funções de linha única:** operam somente em linhas únicas e **retornam um resultado por linha**. Existem tipos diferentes de funções de linha única, incluindo funções de caractere, número, data e conversão de tipos para exibição.
- Podem ser utilizadas também para fazer cálculos, arredondar números para uma casa decimal específica e modificar itens de dados individuais, como converter valores de caractere de letra maiúscula para minúscula.

Comandos DML - Funções multilinha

- **Funções multilinha:** manipulam grupos de linhas para fornecer um resultado por grupo de linhas. São conhecidas como funções de grupo e **retornam apenas um valor para cada conjunto de dados.**
- Na prática as funções multilinha aceitam muitas linhas como entrada e retornam um único valor como saída sendo que, a entrada de linhas pode ser a tabela inteira ou dividida em grupos menores.

Comandos DML - Tabela DUAL

- Para que possamos testar a aplicação de funções em certos tipos de dados sem termos necessariamente que possuímos uma tabela física criada, existe um recurso interessante.
- A tabela **dual** que é utilizada para criar instruções SELECT e executar funções não diretamente relacionadas a uma tabela de banco de dados específica.
- Utiliza-se esta tabela para fazer cálculos e também avaliar expressões que não sejam derivadas de uma tabela.

Comandos DML - Tabela DUAL

- Vamos aplicar um teste utilizando a tabela **dual**, neste teste, simularemos um cálculo para demonstrar que não precisamos ter o dado no banco para conseguirmos apresentar o resultado.

```
SELECT (400/20) + 15  
FROM DUAL;
```

```
SELECT 400/(20 + 15)  
FROM DUAL;
```

```
SELECT SYSDATE() FROM DUAL
```

```
SELECT USER() FROM DUAL
```

```
SELECT * FROM DUAL;
```


Comandos DML - Manipulação de strings

- A manipulação de *strings*, textos ou caracteres é um recurso de muita utilidade no tratamento dos dados em qualquer banco de dados relacional.
- Importante conhecermos estas funções que são de **linha única**, ou seja, são aplicadas como possibilidade de mudar como os dados são apresentados de acordo com o requisito do banco de dados.

Funções de caractere de linha única

- Possuímos duas categorias que classificam estas funções com os conjuntos de ações:
 - Converter as letras maiúsculas e minúsculas de *strings* de caracteres;
 - Juntar, extrair, mostrar, localizar, preencher e reduzir *strings* de caracteres;
- Estas funções podem ser utilizadas nas cláusulas SELECT, WHERE e ORDER BY

Funções de caractere de linha única

-  Estas funções podem nos ajudar a padronizar os registros no banco de dados, principalmente no que se refere a letras maiúsculas, minúsculas ou um misto entre as duas.
- Lembramos que esta conversão, na recuperação **SELECT** é temporária e pode não alterar a originalidade dos dados.
- É um belo recurso para evitar incompatibilidades entre a gravação de letras maiúsculas e minúsculas armazenadas no banco de dados e as solicitações de recuperação nas consultas, basta utilizá-las no **INSERT**.

Funções de caractere de linha única



Manipulação de letras

LOWER : Todas minúsculas
UPPER : Todas maiúsculas
INITCAP: Primeira maiúscula

Função de Caractere

Manipulação de Caracteres

CONCAT	: Junta dois valores
SUBSTR	: Extrai uma string com determinado tamanho
LENGTH	: Mostra o tamanho de uma string em número
INSTR	: Mostra a primeira ocorrência numérica do caractere
LPAD	: Justifica a string preenchendo valores a esquerda
RPAD	: Justifica a string preenchendo valores a direita
TRIM	: Remove os caracteres especificados na string
REPLACE	: Substitui uma sequência de caracteres na string

Funções de caractere de linha única

- Para aplicarmos o exemplo vamos restaurar o banco *employees* que está disponível no CANVAS.
- **Faça o download do arquivo.**
- Opção no Workbench *Administration/Data Import/Restore*
- Aponte o arquivo que foi feito o download.

Let's test all functions!

Now!

Funções de caractere de linha única

```
USE employees;
```

```
SELECT first_name, LOWER(first_name) "CONVERTI PARA MINÚSCULO"  
FROM customer;
```

```
SELECT CONCAT('Hello', 'World')  
FROM DUAL;
```

```
SELECT CONCAT_WS("-",first_name, "*",last_name) as "Complete Name"  
FROM employees limit 3;
```

```
SELECT LENGTH(last_name) as "Tamanho da string"  
FROM employees limit 3;
```

```
SELECT first_name, last_name  
FROM employees  
WHERE first_name = (CONCAT(UPPER(LEFT(first_name, 1)),  
                           LOWER(SUBSTRING(first_name, 2, LENGTH(first_name)))));
```

```
USE sakila;
```

```
SELECT first_name, LOWER(first_name) "CONVERTI PARA MINÚSCULO"  
FROM customer;
```

```
SELECT first_name,  
       CONCAT(UPPER(LEFT(first_name, 1)),  
              LOWER(SUBSTRING(first_name, 2, LENGTH(first_name)))) as "INITCAP ON MYSQL"  
FROM customer;
```

```
SELECT LENGTH('HelloWorld') as "Tamanho da string"  
FROM DUAL;
```

```
SELECT INSTR('HelloWorld', 'W') as "Posição que está a letra"  
FROM DUAL;
```

```
SELECT LPAD('HelloWorld',15, '-') as "Preenchendo a esquerda"  
FROM DUAL;
```

```
SELECT RPAD('HelloWorld',15, '-') as "Preenchendo a direita"  
FROM DUAL;
```

Funções de caractere de linha única

- Estas funções são muito utilizadas no processamento de conteúdos de texto armazenados em bancos de dados.

```
SELECT TRIM(LEADING 'a' FROM 'abcba')  
FROM DUAL;
```

```
SELECT TRIM(TRAILING 'a' FROM 'abcba')  
FROM DUAL;
```

```
SELECT TRIM(BOTH 'a' FROM 'abcba')  
FROM DUAL;
```

```
USE employees;
```

```
SELECT REPLACE(last_name, 'a', '*')  
FROM employees limit 10;
```

```
SELECT REPLACE('JACK and JUE', 'J', 'BL')  
FROM DUAL;
```


Funções de número

- Cálculos com números para apresentação de relatórios consolidados, simulações e principalmente atualização de valores são muito utilizados.
- As funções de número apoiam na execução e entrega destes cálculos de forma efetiva.
- Listaremos então as três funções de números que são tratadas na linguagem SQL.

Funções de número

- **ROUND:** pode ser usado com números e datas, para arredondá-los numa quantidade específica de casas decimais, usa-se também para arredondá-los para o valor à esquerda do ponto decimal.
- Quando a quantidade de casas decimais não for especificada ou for zero, o número será arredondado sem casas decimais.
- Se a quantidade de casas decimais for um número positivo, o número será arredondado para essa quantidade de casas decimais, à direita do ponto decimal, se for negativo à esquerda do ponto decimal.

Funções de número

- **TRUNCATE:** pode ser usado com números e datas, usada principalmente para terminar a coluna, expressão ou o valor com uma quantidade específica de casas decimais.
- Quando a quantidade de casas decimais não for especificada ou for zero, o número será truncado sem casas decimais.
-  **TRUNCATE** não arredonda o número ele simplesmente termina o número em um determinado ponto.

Funções de número

- **MOD:** encontra o resto da divisão entre dois números.
- Geralmente utiliza-se para determinar se o número é par ou ímpar.

More tests!

Funções de número

```
USE sakila;
```

```
SELECT amount, ROUND(amount,1)
```

```
FROM payment
```

```
WHERE amount > 2;
```

```
SELECT amount, truncate(amount,1)
```

```
FROM payment
```

```
WHERE amount > 2;
```

```
SELECT amount, MOD(amount,2)
```

```
FROM payment;
```

```
SELECT MOD(4,2) FROM DUAL;
```

```
SELECT MOD(5,2) FROM DUAL;
```

Funções de data

- As funções de data nos auxiliam a registrar de forma padronizada e, além disto, em toda a questão de cálculos que precisaremos fazer em relação as datas registradas.
- Os formatos irão depender do tipo de dados declarado nas colunas.

Data Type	"Zero" Value
<u>DATE</u>	'0000-00-00'
<u>TIME</u>	'00:00:00'
<u>DATETIME</u>	'0000-00-00 00:00:00'
<u>TIMESTAMP</u>	'0000-00-00 00:00:00'
<u>YEAR</u>	0000

Do you want more?

Tests, of course!

Funções de data

```
SELECT CURDATE() as "Data Corrente do SGBD" FROM DUAL;
```

```
SELECT SYSDATE() as "Data do Sistema" FROM DUAL;
```

```
USE employees;
```

```
SELECT last_name, hire_date, DATE_ADD(hire_date, INTERVAL 180 DAY) as "Daqui a 6 meses (convertidos em dias)"  
FROM employees limit 10;
```

```
SELECT last_name, hire_date, DATE_ADD(hire_date, INTERVAL 5 YEAR) as "Daqui a 5 anos"  
FROM employees limit 10;
```

```
SELECT last_name, hire_date, (DATEDIFF(CURDATE(),hire_date)/365) "Quanto tempo o empregado possui de empresa"  
FROM employees limit 10;
```

```
SELECT dayofweek(hire_date)  
FROM employees limit 10;
```

Atenção ao usar isto!



```
SELECT NOW(), SLEEP(3), NOW() FROM DUAL;
```

```
SELECT SYSDATE(), SLEEP(3), SYSDATE() FROM DUAL;
```

Referências bibliográficas

Neto, Olibário, 2020. Álgebra Relacional em Bancos de Dados - Operações Unárias (Seleção, Projeção e Renomeação). Disponível em:
<<https://www.youtube.com/watch?v=E-tIwv8jDhE>>. Acesso em: 24 out. 2021.

Borges, Vanessa, 2021. Linguagens relacionais - Álgebra relacional. Disponível em:
<<https://www.youtube.com/watch?v=e39T5H8gwRY>>. Acesso em: 24 out. 2021.

Macoratti, José Carlos. SQL - Álgebra Relacional - Operações Fundamentais - Conceitos básicos. Disponível em:
<http://www.macoratti.net/13/06/sql_arcb.htm>. Acesso em: 24 out. 2021.