



Linguagem SQL

11 – Funções Built-In I

Sand Onofre
Sand.Onofre@gmail.com

Sumário

- Data Query Language – DQL
 - Funções Built-in e seus Tipos:
 - String
- Exercícios

Funções Built-in I

Bancos de Dados possuem várias funções embutidas (built-in) que são automaticamente disponibilizadas após a instalação do sistema, ou seja, disponíveis a qualquer momento para que possamos melhorar as buscas por dados, realizar cálculos ou simplesmente uma melhor visualização de conteúdo de informações.

Cada função trata determinados tipos de dados. Algumas funções são estritamente necessárias no trato da informação e focaremos nas principais funções/tipos de dados:

- Funções String
- Funções Data
- Funções Numéricas
- Funções de Conversão

Existem dezenas de outros tipos de funções que não trataremos nesse tópico !

Funções String

Funções específicas para tratamento de dados do tipo caracteres. Principais funções string Built-in:

- SUBSTRING
- LEFT
- RIGHT
- UPPER
- LOWER
- LTRIM
- RTRIM
- LEN
- SPACE
- REPLICATE
- REPLACE
- REVERSE
- CHARINDEX
- CASE

Funções String

- SUBSTRING

Retorna parte de um texto, seja uma expressão ou uma coluna de tabela, fornecendo a posição inicial e a quantidade de caracteres que queremos:

SUBSTRING (Texto Alvo, posição de início, núm. de caracteres)

Exemplo:

```
SELECT SUBSTRING('Sistemas de Informação', 13, 10) as Extrai
```

```
Extrai
-----
Informação
```

Funções String

- LEFT

Retorna o número de caracteres especificado a esquerda da expressão ou do campo informado:

LEFT(Texto Alvo, núm. de caracteres iniciando à esquerda)

Exemplo:

```
SELECT nome, left(nome, 4) as Esquerda FROM Cliente
```

Nome	Esquerda

João	João
José	José
Marcelo	Marc
Mário	Mári
...	

Funções String

- RIGHT

Retorna o número de caracteres especificado a direita da expressão ou do campo informado:

RIGHT(Texto Alvo, núm. de caracteres iniciando à direita)

Exemplo:

```
SELECT sobrenome, right(sobrenome, 5) as Direita FROM Cliente
```

sobrenome	Direita

Silva	Silva
Nogueira	ueira
Andrade	drade
Pereira	reira
...	

Funções String

- UPPER

Transforma toda a expressão ou coluna em caracteres MAIÚSCULOS:

UPPER(Texto Alvo)

Exemplo:

```
SELECT upper('abcedfg') as Máíúsculas
```

Maiúsculas

ABCDEF G

Funções String

- LOWER

Transforma toda a expressão ou coluna em caracteres MINÚSCULOS:

LOWER (Texto Alvo)

Exemplo:

```
SELECT nome, lower(nome) as Minúsculas FROM Cliente
```

nome	Minúsculas

Marcelo	marcelo
Adriana	adriana
Maria	maria
...	

Funções String

- LTRIM

Remove espaços a direita de uma expressão ou coluna:

LTRIM(Texto Alvo)

Exemplo:

```
SELECT ltrim('  Sistemas de Informação') as 'Extrai Espaços Esquerda'
```

Extrai Espaços Esquerda

Sistemas de Informação

Funções String

- RTRIM

Remove espaços a esquerda de uma expressão ou coluna:

`RTRIM(Texto Alvo)`

Exemplo:

```
SELECT rtrim('Sistemas de Informação   ') as 'Extrai Espaços Direita'
```

```
Extrai Espaços Direita
-----
Sistemas de Informação
```

Funções String

- LEN

Retorna o número de caracteres de uma expressão ou coluna:

LEN(Texto Alvo)

Exemplo:

```
SELECT nome, Len(nome) as 'Número de Caracteres' FROM Cliente
```

Nome	Número de Caracteres
-----	-----
Mário	5
Antenor	7
Adalberto	9
...	

Funções String

- SPACE

Adiciona espaços para alguma concatenação

`SPACE(n)`

Exemplo:

```
SELECT nome + sobrenome as Concatena1
       , nome + space(5) + sobrenome as Concatena2
FROM Cliente
```

Concatena1	Concatena2	
-----	-----	
JoãoSilva	João	Silva
AntônioAndrade	Antônio	Andrade
ThiagoAbrantes	Thiago	Abrantes
...		

Funções String

- REPLICATE

Repete um determinado caractere ou expressão, o número de vezes solicitado:

`REPLICATE(Texto a replicar, Quantidade de Vezes)`

Exemplo:

```
SELECT replicate('0', 4) + '1' as 'Gera Número'
```

Gera Número

00001

Funções String

- REPLACE

Substitui um conjunto de caracter por outro, em uma expressão ou coluna:

REPLACE(Texto Alvo, 'string antigo', 'novo string')

Exemplo:

```
SELECT Placa, replace(Placa, '-', '') as Novo FROM Veiculos
```

Placa	Novo
KST-4356	KST4356
XPT-9811	XPT9811
...	

Funções String

- REVERSE

Reverte a ordem de uma expressão ou coluna:

`REVERSE(Texto Alvo)`

Exemplo:

```
SELECT reverse('123456') as 'String Revertida'
```

String Revertida

654321

Funções String

- CHARINDEX

Pesquisa uma expressão para outra e retorna sua posição inicial, se for localizada.

CHARINDEX (Texto a Procurar, Texto Alvo[, Caracter de Início])

Exemplo:

```
SELECT charindex('Teste', 'Isso é um Teste') as Posição
```

Posição

11

Funções String

- CASE (primeira forma)

Compara uma expressão ou coluna, fornecendo um valor escolhido:

```
CASE expression
    WHEN a THEN b
    WHEN c THEN d
    WHEN e THEN f
    ELSE w
END
```

Funções String

- CASE (primeira forma)

```
SELECT Sexo, CASE Sexo
                WHEN 0 THEN 'Feminino'
                WHEN 1 THEN 'Masculino'
            END as Gênero
FROM Cliente
```

Sexo	Gênero
1	Masculino
0	Feminino
0	Feminino
...	

Funções String

- CASE (segunda forma)

Compara uma expressão ou coluna, fornecendo um valor previamente escolhido:

CASE

WHEN expression THEN b

WHEN expression THEN d

WHEN expression THEN f

ELSE w

END

Funções String

- CASE (segunda forma)

```
SELECT Idade
      , CASE
          WHEN idade <= 14 THEN 'Menor de Idade'
          WHEN idade BETWEEN 15 AND 17 THEN 'Adolescente'
          ELSE 'Adulto'
        END as Maturidade
FROM Cliente
```

Idade	Maturidade

21	Adulto
35	Adulto
12	Menor de Idade
16	Adolescente
...	

Demonstração

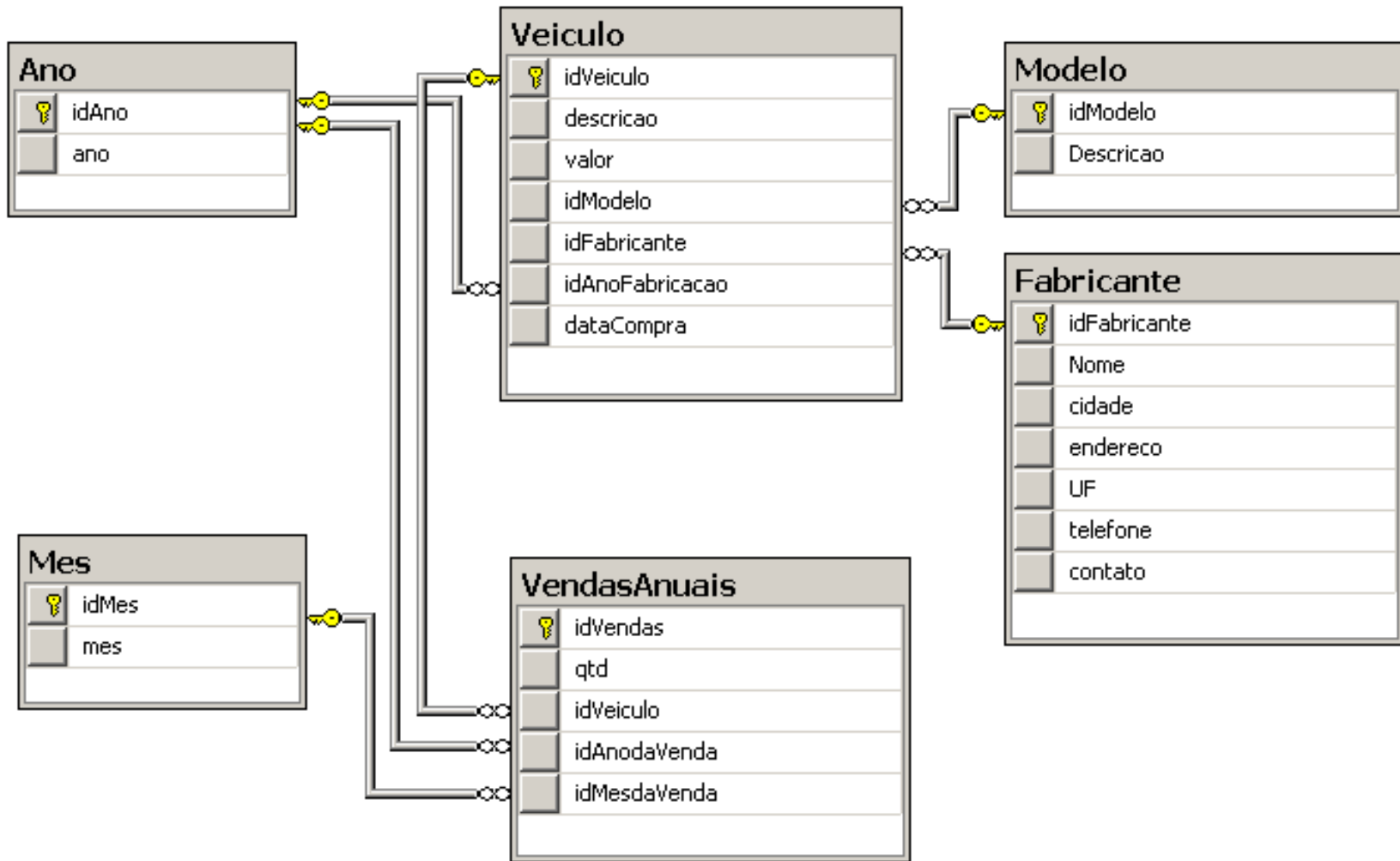
Exemplos Práticos

1. Em um site de venda de jogos, temos uma tabela que guarda o cadastro de clientes (*ClienteJogo*). Queremos fazer uma campanha de vendas para determinados provedores parceiros do site. Precisamos saber todos os provedores dos clientes através do campo *EMAIL*, ou seja, se temos *cliente@uol.com.br*, queremos ter apenas *uol.com.br*. O mesmo para qualquer outro provedor. Utilizando funções que tratam caracteres, elabore um *SELECT* que possibilite esta extração ?

```
SELECT    right(email, len(email) - charindex('@', email))

FROM      ClienteJogo
```

Modelo Concessionaria



Exercícios

No banco de dados Concessionaria, extraia através de funções built-in as seguintes consultas:

1. Traga o nome dos fabricantes que possuam mais de 5 letras.
2. Traga o nome dos fabricantes que terminem com a letra “i”.
3. Concatene numa coluna com o nome “CONCATENA”, composta pela 1ª. e última letra dos nomes de cada fabricante (ambas maiúsculas).
4. Retorne o endereço dos fabricantes substituindo a palavra “Av.” por “Avenida”.
5. Utilizando o predicado LIKE, traga os endereços que contenha a palavra “DA”.
6. Utilizando a função CHARINDEX, traga os endereços que contenha a palavra “DA”.
7. Liste os 8 primeiros telefones da tabela Fabricante, separando por hífen os primeiros 4 dígitos, dos últimos dígitos.
8. Na tabela fabricante, liste uma coluna de nome “NÚMERO” contendo apenas o número dos endereços.
9. Faça um JOIN entre Veículo e Fabricante, trazendo a Descrição do Veículo, a UF do Fabricante e através de um CASE, liste um campo de nome “REGIÃO”, colocando a região do Brasil referente a UF encontrada (ex: Região Sul, Região Norte, ...).
10. Partindo da query anterior, com um outro CASE, liste uma coluna de nome “SEGMENTO DA VENDA”, onde o conteúdo deve seguir a seguinte regra:
 - “Segmento 1” – quando a cidade for São Paulo
 - “Segmento 2” – cidade do Rio de Janeiro
 - “Segmento 3” – cidade de Curitiba
 - “Segmento 4” – qualquer outra cidade



Obrigado !

Sand Onofre
Sand.Onofre@gmail.com