



Linguagem SQL

12 – Funções Built-In II

Sand Onofre
Sand.Onofre@gmail.com

Sumário

- Data Query Language – DQL
 - Funções Built-in II:
 - Data
 - Numérica
 - Conversão
- Exercícios

Funções Built-in II

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 Data

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

- YEAR
- MONTH
- DAY
- GETDATE
- DATEDIFF
- DATEADD
- DATEPART

Funções Data

- YEAR

Retorna um **valor inteiro** para o ano da data na expressão ou coluna fornecida:

`YEAR(expression)`

Exemplo:

```
SELECT dtNascimento, Year(dtNascimento) FROM Cliente
```

dtNascimento	No column Name
19690921	1969
19911231	1991
19080101	1908
20150331	2015
...	

Funções Data

- MONTH

Retorna um **valor inteiro** para o mês da data na expressão ou coluna fornecida:

`MONTH(expression)`

Exemplo:

```
SELECT dtNascimento, Month(dtNascimento) FROM Cliente
```

dtNascimento	No column Name
19690921	9
19911231	12
19080101	1
20150331	3
...	

Funções Data

- DAY

Retorna um **valor inteiro** para o dia da data na expressão ou coluna fornecida:

`DAY(expression)`

Exemplo:

```
SELECT dtNascimento, Day(dtNascimento) FROM Cliente
```

dtNascimento	No column Name
19690921	21
19911231	31
19080101	1
20150331	31
...	

Funções Data

- GETDATE

Retorna um **valor datetime**, trazendo a data e a hora (em milissegundos) do servidor que estiver conectado:

Getdate ()

Exemplo:

```
SELECT Getdate() as 'Data e hora do Servidor'
```

```
Data e hora do Servidor
-----
2015-03-21 19:22:25.667
```


Funções Data

- DATEDIFF

Retorna um **valor inteiro**, trazendo a diferença entre duas datas, de acordo com a parte de dados requisitada na função:

DATEDIFF (datepart , startdate , enddate)

O valor datepart pode ser um dos abaixo:

<i>datepart</i>	Abreviações
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww

<i>datepart</i>	Abreviações
hour	hh
minute	mi, n
second	ss, s
millisecond	ms
microsecond	mcs
nanosecond	ns

Funções Data

- DATEDIFF

Exemplo:

```
SELECT DateDiff(yy, '20150301', '20140301') as Ano  
      , DateDiff(mm, '20151201', '20150101') as Mês  
      , DateDiff(dd, '20150321', '20150301') as Dia
```

Ano	Mês	Dia
1	11	20

Funções Data

- DATEADD

Adiciona um valor em uma data, de acordo com a parte de dados solicitada na função:

DATEADD (datepart , number , date)

O valor datepart pode ser um dos abaixo:

<i>datepart</i>	Abreviações
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww

<i>datepart</i>	Abreviações
hour	hh
minute	mi, n
second	ss, s
millisecond	ms
microsecond	mcs
nanosecond	ns

Funções Data

- DATEADD

Exemplo:

```
SELECT Dateadd(yy, 1, '20140526') as Ano
      , Dateadd(mm, -2, '20150101') as Mês
      , Dateadd(dd, 30, '20150301') as Dia
```

Ano	Mês	Dia
2015-05-26	2014-11-01	2015-03-31

Funções Data

- DATEPART

Retorna um valor inteiro, de acordo com a parte de dados solicitada na função:

DATEPART (datepart , date)

O valor datepart pode ser um dos abaixo:

<i>datepart</i>	Abreviações
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww

<i>datepart</i>	Abreviações
hour	hh
minute	mi, n
second	ss, s
millisecond	ms
microsecond	mcs
nanosecond	ns

Funções Data

- DATEPART

<i>datepart</i>	Valor de retorno
year, yyyy, yy	2007
quarter, qq, q	4
month, mm, m	10
dayofyear, dy, y	303
day, dd, d	30
week, wk, ww	45
weekday, dw	1
hour, hh	12
minute, n	15
second, ss, s	32
millisecond, ms	123

Demonstração de Funções Data

Funções Numéricas

Também chamada de funções Matemáticas, são específicas para tratamento de dados do tipo numérico. Principais funções Built-in:

- ABS
- CEILING
- FLOOR
- POWER
- SQRT
- RAND
- ROUND

Funções Numéricas

- ABS

Uma função matemática que retorna o valor absoluto (positivo) da expressão numérica especificada.

`ABS(numeric_expression)`

Exemplo:

```
SELECT ABS(-1.0), ABS(0.0), ABS(1.0)
```

No column Name	No column Name	No column Name
1	0	1

Funções Numéricas

- CEILING

Retorna o inteiro maior ou igual a expressão numérica especificada.

`CEILING (numeric_expression)`

Exemplo:

```
SELECT CEILING(123.45), CEILING(-123.45), CEILING(0.0)
```

No column Name	No column Name	No column Name
-----	-----	-----
124.00	-123.00	0.0

Funções Numéricas

- FLOOR

Retorna o inteiro menor ou igual à expressão numérica especificada.

`FLOOR (numeric_expression)`

Exemplo:

```
SELECT FLOOR(123.45), FLOOR(-123.45), FLOOR(0.0)
```

No column Name	No column Name	No column Name
123.00	-124.00	0.0

Funções Numéricas

- POWER

Retorna o valor da expressão especificada elevada à potência especificada.

`POWER (float_expression, y)`

Exemplo:

```
SELECT POWER(2, 2), POWER(3, 4), POWER(5, 3)
```

No column Name	No column Name	No column Name
4	81	125

Funções Numéricas

- SQRT

Retorna a raiz quadrada do valor flutuante especificado.

`SQRT (float_expression)`

Exemplo:

```
SELECT SQRT(4), SQRT(9), SQRT(25)
```

No column Name	No column Name	No column Name
2	3	5

Funções Numéricas

- RAND

Retorna um valor **float** pseudoaleatório de 0 a 1, exclusivo.

RAND ()

Exemplo:

```
SELECT RAND()
```

```
No column Name
```

```
-----
```

```
0.367392786982856743308
```

Funções Numéricas

- ROUND

Retorna um valor numérico, arredondado, para o comprimento ou precisão especificados.

`ROUND (numeric_expression , length)`

Exemplo:

```
SELECT ROUND(748.58, 0) as R0, ROUND(748.58, 1) as R1, ROUND(748.58, 2) as R2,
ROUND(748.58, -1) as Rm1, ROUND(748.58, -2) ) ) as Rm2
```

R0	R1	R2	Rm1	Rm2
749.00	748.6	748.58	750.00	700.00

Demonstração de Funções Numéricas

Funções de Conversão

Convertem um tipo de dados para outro tipo. Principais funções Built-in:

- CAST
- CONVERT

Funções de Conversão

- CAST

Converte uma expressão de um tipo de dados em outro.

`CAST (expression AS data_type [(length)])`

Exemplo:

```
SELECT 'Cliente ' + CAST(idCliente as varchar) as NumCli, idCliente
FROM   TBCliente
```

NumCli	idCliente

Cliente 1	1
Cliente 2	2
Cliente 3	3
...	

Funções de Conversão

- CONVERT

Converte uma expressão de um tipo de dados em outro.

`CONVERT (data_type [(length)] , expression [, style])`

Onde Style, mais utilizado para Datas pode ser:

Sem século (aa)	Com século (aaaa)	Standard	Entrada/Saída
-	0 ou 100	DEFAULT	mês dd aaaa hh:mi AM (ou PM)
1	101	EUA	1 = mm/dd/aa 101 = mm/dd/aaaa
2	102	ANSI	2 = aa.mm.dd 102 = aaaa.mm.dd
3	103	Britânico/francês	3 = dd/mm/aa 103 = dd/mm/aaaa
4	104	Alemão	4 = dd.mm.aa 104 = dd.mm.aaaa
5	105	Italiano	5 = dd-mm-aa 105 = dd-mm-aaaa

Funções de Conversão

Sem século (aa)	Com século (aaaa)	Standard	Entrada/Saída
6	106	-	6 = dd mês aa 106 = dd mês aaaa
7	107	-	7 = Mês dd, aa 107 = Mês dd, aaaa
8	108	-	hh:mi:ss
-	9 ou 109	Padrão + milissegundos	mês dd aaaa hh:mi:ss:mmmAM (ou PM)
10	110	EUA	10 = mm-dd-aa 110 = mm-dd-aaaa
11	111	JAPÃO	11 = aa/mm/dd 111 = aaaa/mm/dd
12	112	ISO	12 = aammdd 112 = aaaammdd
-	13 ou 113	Padrão Europa + milissegundos	dd mês aaaa hh:mi:ss:mmm (24h)
14	114	-	hh:mi:ss:mmm(24h)
-	20 ou 120	ODBC canônico	aaaa-mm-dd hh:mi:ss(24h)

Funções de Conversão

Sem século (aa)	Com século (aaaa)	Standard	Entrada/Saída ⁽³⁾
-	21 ou 121	ODBC canônico (com milissegundos)	aaaa-mm-dd hh:mi:ss.mmm(24h)
-	126	ISO8601	aaaa-mm-ddThh:mi:ss.mmm (sem espaços)
-	130	Islâmico	dd mmm aaaa hh:mi:ss:mmmAM Neste estilo, mon representa uma representação unicode Hijri de vários tokens do nome completo do mês. Este valor não será renderizado corretamente em uma instalação US padrão do SSMS.
-	131	Islâmico	dd/mm/aaaa hh:mi:ss:mmmAM

Funções de Conversão

- CONVERT

Exemplo:

```
SELECT GETDATE() AS Dt1
      , CAST(GETDATE() AS varchar(30)) AS DtCast
      , CONVERT(varchar(8), GETDATE(), 112) AS DtConvert
```

Dt1	DtCast	DtConvert
2006-04-18 09:58:04.570	Apr 18 2006 9:58AM	20060418

Demonstração de Funções de Conversão

Exemplos Práticos

1. A página do site de jogos online precisa mostrar a pontuação de todos os jogadores nas últimas 6 horas de jogo, da tabela *JogoScore*, coluna *lastDateGame*. Utilizando funções de data, desenvolva o WHERE que faria esta condição ?

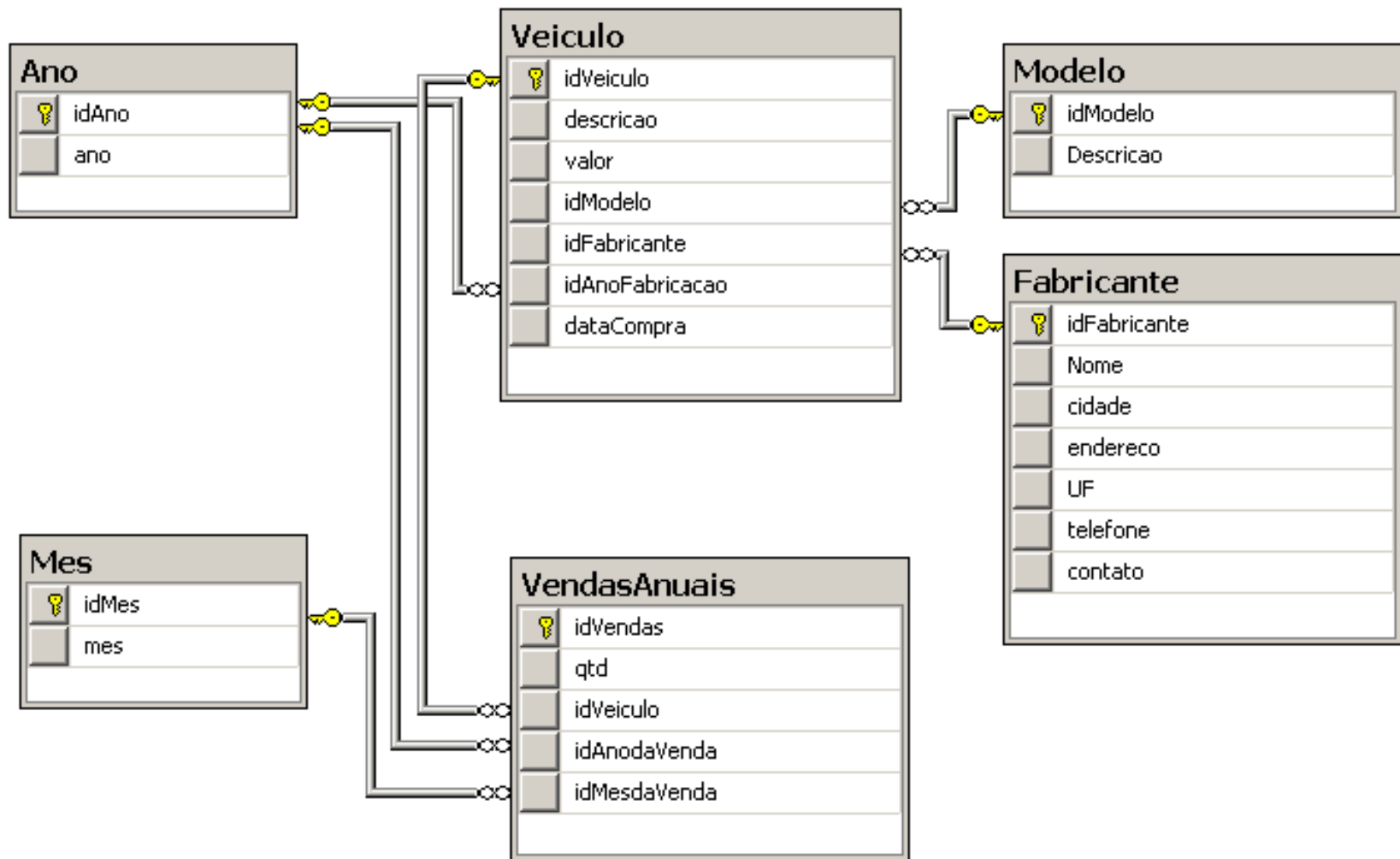
```
SELECT *
FROM    JogoScore
WHERE    Datediff(hh, lastDateGame, getdate()) <= 6
```


Exemplos Práticos

2. Em uma determinada fase de um jogo, queremos gerar uma pontuação aleatória entre 0 e 1000, adicionando ao score do usuário. Utilizando funções numéricas, mostre um SELECT que faça isso ?

```
SELECT    round(rand() * 1000, 0)
```

Modelo Concessionaria



Exercícios

No banco de dados Concessionaria, extraia os seguintes SELECTs, utilizando as funções da aula:

1. Na tabela **Veículo**, encontre a média (com casas decimais) para todos os valores de veículos que foram comprados entre os anos de 2003 e 2006.
2. Utilizando funções de arredondamento, faça com que a média encontrada na query anterior tenha (executar um SELECT por vez para cada uma das solicitações abaixo) :
 - 2a. Aproximação para a 2ª. casa decimal
 - 2b. Seja truncado para o inteiro menor que a média encontrada
 - 2c. Seja truncado para o inteiro maior que a média encontrada
3. Faça um SELECT que traga a data de compra no formato de nosso idioma, ou seja, DD/MM/YYYY.
4. Na tabela de VendasAnuais, faça a soma das quantidades para cada mês ímpar do ano de 2004.
5. Faça uma query similar a anterior, trazendo agora a soma das quantidades para cada ano e trimestre.



Obrigado !

Sand Onofre
Sand.Onofre@gmail.com