

# Administrador de Banco de Dados

## **Módulo II**

SQL Server: Consultas avançadas com Microsoft SQL Server 2019;

# O que iremos aprender?



- Funções de string;
- Funções de datas;
- Funções matemáticas;
- Conversão de dados



## Aula 10 – Funções do SQL Server

# Funções de string

LTRIM – Retorna uma expressão de caractere depois de remover espaços em branco à esquerda.

LTRIM (expressão de caracteres)

# Funções de string

RTRIM – Retorna uma de caracteres depois de truncar todos os espaços à direita.

RTRIM (expressão de caracteres)

# Funções de string

LEFT – Retorna a parte da esquerda de uma cadeia de caracteres com o número de caracteres especificado.

LEFT (expressão de caracteres, expressão inteiro)

# Funções de string

RIGHTH – Retorna a parte direita de uma cadeia de caracteres com o número caracteres especificado.

RIGHT (expressão de caracteres, expressão inteiro)

# Funções de string

CONCAT – Retorna uma cadeia de caracteres que é o resultado da concatenação de dois ou mais valores.

CONCAT (expressão de caracteres 1, expressão de caracteres 2, expressão de caracteres N)



# Funções de string

CONCAT – Retorna uma cadeia de caracteres que é o resultado da concatenação de dois ou mais valores.

CONCAT (expressão de caracteres 1, expressão de caracteres 2, expressão de caracteres N)

# Funções de string

SUBSTRING – Retorna parte de uma expressão de caracteres, binária, de texto ou de imagem no SQL Server.

SUBSTRING (expressão, início, tamanho)

# Funções de string

LEN – Retorna o número de caracteres da expressão da cadeia de caracteres especificada, excluindo espaços em branco à direita.

LEN (expressão de caracteres)

# Funções de string

UPPER – Retorna uma expressão de caracteres com dados de caracteres em minúsculas convertidos em maiúsculas.

UPPER (expressão de caracteres)

# Funções de string

LOWER – Retorna uma expressão de caracteres depois de converter para minúsculas os dados de caracteres em maiúsculas.

LOWER (expressão de caracteres)

# Funções de string

## Diversas funções de string

|            |           |               |
|------------|-----------|---------------|
| ▼ String   | LOWER     | SPACE         |
| String     | LTRIM     | STR           |
| ASCII      | NCHAR     | STRING_AGG    |
| CHAR       | PATINDEX  | STRING_ESCAPE |
| CHARINDEX  | QUOTENAME | STRING_SPLIT  |
| CONCAT     | REPLACE   | STUFF         |
| CONCAT_WS  | REPLICATE | SUBSTRING     |
| DIFFERENCE | REVERSE   | TRANSLATE     |
| FORMAT     | RIGHT     | TRIM          |
| LEFT       | RTRIM     | UNICODE       |
| LEN        | SOUNDEX   | UPPER         |

<https://docs.microsoft.com/pt-br/sql/t-sql/functions/functions?view=sql-server-ver15>

**DESAFIO**



# Desafio

Faça uma consulta que lista o nome do cliente e o endereço completo (com rua, bairro, cidade e estado).

```
SELECT NOME, CONCAT([ENDERECO], ' ', BAIRRO, ' ', CIDADE, ' ', ESTADO) AS 'ENDERECO COMPLETO'  
FROM [TABELA DE CLIENTES]
```



# Funções de Datas

| Tipo de dados         | Formato                                 | Intervalo  | Precisão         | Tamanho de armazenamento (bytes) |
|-----------------------|---|--|------------------|----------------------------------|
| <b>time</b>           | hh:mm:ss[nnnnnn]                        | 00:00:00.0000000 a 23:59:59.9999999                                | 100 nanosegundos | 3 a 5                            |
| <b>date</b>           | AAAA-MM-DD                              | 0001-01-01 a 9999-12-31  | 1 dia            | 3                                |
| <b>smalldatetime</b>  | AAAA-MM-DD hh:mm:ss                     | 01.01.00 a 06.06.79  | 1 min            | 4                                |
| <b>datetime</b>       | AAAA-MM-DD hh:mm:ss[.nnn]               | 1753-01-01 a 9999-12-31  | 0,00333 segundo  | 8                                |
| <b>datetime2</b>      | AAAA-MM-DD<br>hh:mm:ss[.nnnnnnnn]       | 0001-01-01 00:00:00.0000000 a 9999-12-31 23:59:59.9999999          | 100 nanosegundos | 6 a 8                            |
| <b>datetimeoffset</b> | AAAA-MM-DD HH[.nnnnnnnn][+ -]<br>]hh:mm | 0001-01-01 00:00:00.0000000 a 9999-12-31 23:59:59.9999999 (em UTC) | 100 nanosegundos | 8 a 10                           |

# Funções de Datas

**SYSDATETIME** – Retorna uma **datetime2** valor que contém a data e hora do computador no qual a instância do SQL Server está em execução.

**SYSDATETIMEOFFSET** – Retorna um **datetimeoffset**.

**SYSUTCDATETIME**- Retorna **datetime2**. A data e hora é retornada como hora UTC

**SYSDATETIME( )**

**SYSDATETIMEOFFSET( )**

**SYSUTCDATETIME( )**

# Funções de Datas

**CURRENT\_TIMESTAMP** – Retorna uma **datetime**

**GETDATE** – Retorna um **datetime**.

**GETUTCDATE** – Retorna **datetime**. A data e hora é retornada como hora UTC

**CURRENT\_TIMESTAMP**

**GETDATE( )**

**GETUTCDATE( )**

# Funções de Datas

Funções que obtêm partes de data e hora

| Função          | Sintaxe                     | Valor do retorno  | Tipo de dados de retorno |
|-----------------|-----------------------------|---|--------------------------|
| <b>DATENAME</b> | DATENAME<br>(datepart,data) | Retorna uma cadeia de caracteres que representa a datapart da data especificada | nvarchar                 |
| <b>DATEPART</b> | DATEPART(datepart,data)     | Retorna um inteiro que representa a datepart especificada data                  | int                      |
| <b>DAY</b>      | DAY(data)                   | Retorna um inteiro que representa a parte do dia especificada data              | int                      |
| <b>MONTH</b>    | MONTH(data)                 | Retorna um inteiro que representa a parte do mês especificada data              | int                      |
| <b>YEAR</b>     | YEAR(data)                  | Retorna um inteiro que representa a parte do ano especificada data              | int                      |

# Funções de Datas

Funções que obtêm partes valores de data e hora e suas partes

| Função                         | Sintaxe  | Valor do retorno   | Tipo de dados de retorno |
|--------------------------------|--|--|--------------------------|
| <b>DATEFROMPARTS</b>           | DATEFROMPARTS(ano,mês,dia)   | Retorna uma data valor para o ano especificado, mês e dia. | date                     |
| <b>DATETIME2FROMPARTS</b>      | DATETIME2FROMPARTS (ano,mês,dia, hora,minuto,segundos,frações,precisão)                                  | Retorna um datetime2.                                      | datetime2                |
| <b>DATETIMEFROMPARTS</b>       | DATETIMEFROMPARTS(ano.mes,dia, hora,minuto,segundos,milisegundos)  | Retorna um datetime  | datetime                 |
| <b>DATETIMEOFFSETFROMPARTS</b> | DATETIMEOFFSETFROMPARTS(ano,mês, dia,hora,minuto,segundos,frações, hour_offset, minute_offset, precisão) | Retorna um datetimeoffset                                  | datetimeoffset           |
| <b>SMALLDATETIMEFROMPARTS</b>  | SMALLDATETIMEFROMPARTS(ano,mês, dia,hora,minuto)   | Retorna um smalldate                                       | smalldate                |
| <b>TIMEFROMPARTS</b>           | TIMEFROMPARTS  | Retorna um tempo valor para o tempo especificado           | tempo                    |

# Funções de Datas

Funções que obtêm diferença de data e hora

| Função              | Sintaxe                                   | Valor do retorno  | Tipo de dados de retorno |
|---------------------|---|---|--------------------------|
| <b>DATEDIFF</b>     | DATEIFF (datepart,startdate,enddate)      | Retorna o número de data ou hora dateparts que são cruzados entre duas datas especificadas.     | int                      |
| <b>DATEDIFF_BIG</b> | DATEDIFF_BIG (datepart,stardate, enddate) | Retorna número de data ou hora datepart limites que são cruzados entr duas datas especificadas. | bigint                   |

# Funções de Datas

Funções que modificam os valores de data e hora

| Função                  | Sintaxe  | Valor do retorno   | Tipo de dados de retorno                                  |
|-------------------------|--|--|---|
| <b>DATEADD</b>          | DATEADD (datepart,numero,data)                   | Retorna um novo datetime valor adicionado um intervalo especificado datepart especificada data | o tipo de dados de data argumento                         |
| <b>EOMONTH</b>          | EOMONTH(start_date[month_to_add])                | Retorna o último dia do mês que contém a data especificada com um deslocamento opcional        | Tipo de retorno é o tipo de start_date ou data            |
| <b>SWITCHOFFSET</b>     | opcao deslocamento(DATETIMEOFFSET, fuso_horario) | Altera o deslocamento de fuso horário do valor DATETIMEOFFSET e preserva o valor UTC           | DateTimeOffset com precisão fracionária de datetimeoffset |
| <b>TODATETIMEOFFSET</b> | TODATETIMEOFFSET(expressão, fuso_horario)        | Transforma um valor datetime2 em um valor datetimeoffset.                                      | DateTimeOffset com precisão fracionária de datetimeoffset |

# Funções de Datas

Funções que validam valores de data e hora

| Função | Sintaxe           | Valor do retorno   | Tipo de dados de retorno |
|--------|-------------------|--|--------------------------|
| ISDATE | ISDATE(expressão) | Determina se um datetime ou smalldatetime expressão de entrada é uma data válida ou um valor de tempo. | int                      |



# Funções de Datas

| Função         | Descrição  |
|----------------|--|
| CAST e CONVERT | Fornece informações sobre a conversão de valores de data e hora para e de literais de cadeia de caracteres, bem como outros formatos de data e hora. |

**DESAFIO**



# Desafio

Crie uma consulta que mostre o nome e a idade dos clientes.

```
SELECT NOME, DATEDIFF(YEAR, [DATA DE NASCIMENTO], GETDATE()) AS 'IDADE'  
FROM [TABELA DE CLIENTES]
```

# Funções Matemáticas

| Função  | Descrição   |
|---------|---|
| CEILING | Retorna o maior inteiro maior que ou igual a expressão numérica especificada. |

CEILING (expressão numérica)

# Funções Matemáticas

| Função | Descrição   |
|--------|---|
| FLOOR  | Retorna o menor inteiro menor que ou igual a expressão numérica especificada. |

FLOOR (expressão numérica)

# Funções Matemáticas

| Função      | Descrição   |
|-------------|---|
| <b>RAND</b> | Retorna um valor float pseudoaleatório de 0 a 1, exclusivo. |

RAND ([seed])

# Funções Matemáticas

| Função | Descrição   |
|--------|---|
| ROUND  | Retorna um valor numérico, arredondado, para o comprimento ou precisão especificados. |

ROUND (expressão numérica, tamanho)

# Funções Matemáticas

|                             |                                |                                |
|-----------------------------|--------------------------------|--------------------------------|
| <a href="#"><u>ABS</u></a>  | <a href="#"><u>DEGREES</u></a> | <a href="#"><u>TAN</u></a>     |
| <a href="#"><u>ACOS</u></a> | <a href="#"><u>EXP</u></a>     | <a href="#"><u>SIGN</u></a>    |
| <a href="#"><u>ASIN</u></a> | <a href="#"><u>LOG</u></a>     | <a href="#"><u>SIN</u></a>     |
| <a href="#"><u>ATAN</u></a> | <a href="#"><u>LOG10</u></a>   | <a href="#"><u>SQRT</u></a>    |
| <a href="#"><u>ATN2</u></a> | <a href="#"><u>SQUARE</u></a>  | <a href="#"><u>COT</u></a>     |
| <a href="#"><u>PI</u></a>   | <a href="#"><u>POWER</u></a>   | <a href="#"><u>RADIANS</u></a> |
| <a href="#"><u>COS</u></a>  |                                |                                |



**DESAFIO**



# Desafio

Na tabela de notas fiscais, temos o valor do imposto. Já na tabela de itens, temos a quantidade e o faturamento. Calcule o valor do imposto pago no ano de 2016, arredondando para o menor inteiro.

```
SELECT YEAR(DATA) AS 'ANO', FLOOR(SUM(IMPOSTO * (QUANTIDADE * PRECO))) AS 'IMPOSTO PAGO'
FROM [TABELA DE NOTAS FISCAIS] NF
INNER JOIN [TABELA DE ITENS NOTAS FISCAIS] INF
ON NF.NUMERO = INF.NUMERO
WHERE YEAR(DATA) = 2016
GROUP BY YEAR(DATA)
```

# Conversão de dados

| Função          | Descrição  |
|-----------------|--|
| CAST or CONVERT | Converte uma expressão de um tipo de dados para outro. |

# Conversão de dados

Qual é a diferença entre CAST e CONVERT?

- Ambos fazem a mesma coisa.
- Porém: CAST é padrão SQL. Pode ser usado em outros bancos de dados.
- CONVERT é específico para SQL Server

Porém, pelo fato do CONVERT ser específico, ele possui funcionalidades mais complexas.

# Conversão de dados

Usando o CONVERT

- Data e hora conversão

CONVERT (<tipo>,<tipo data>, <estilo>)

EX.: CONVERT (VARCHAR, GETDATE ( ), 101)

# Estilos de data e hora

| Sem século (yy) <sup>(1)</sup> | Com século (aaaa)         | Standard                             | Entrada/saída <sup>(3)</sup>         |
|--------------------------------|---------------------------|--------------------------------------|--------------------------------------|
| -                              | 0 ou 100 <sup>(1,2)</sup> | Padrão para datetime e smalldatetime | mês dd aaaa hh:miAM (ou PM)          |
| 1                              | 101                       | EUA                                  | 1 = mm/dd/aa<br>101 = mm/dd/aaaa     |
| 2                              | 102                       | ANSI                                 | 2 = aa.mm.dd<br>102 = aaaa.mm.dd     |
| 3                              | 103                       | Britânico/francês                    | 3 = dd/mm/aa<br>103 = dd/mm/aaaa     |
| 4                              | 104                       | Alemão                               | 4 = dd.mm.aa<br>104 = dd.mm.aaaa     |
| 5                              | 105                       | Italiano                             | 5 = dd-mm-aa<br>105 = dd-mm-aaaa     |
| 6                              | 106 <sup>(1)</sup>        | -                                    | 6 = dd mês aa<br>106 = dd mês aaaa   |
| 7                              | 107 <sup>(1)</sup>        | -                                    | 7 = Mês dd, aa<br>107 = Mês dd, aaaa |

# Estilos de data e hora

|         |                                |  |                                    |
|---------|--------------------------------|--|------------------------------------|
| 8 ou 24 | 108                            | -  | hh:mi:ss                           |
| -       | 9 ou 109 <sup>(1,2)</sup>      | Padrão + milissegundos   | mês dd aaaa hh:mi:ss:mmmAM (ou PM) |
| 10      | 110                            | EUA  | 10 = mm-dd-aa<br>110 = mm-dd-aaaa  |
| 11      | 111                            | JAPÃO  | 11 = aa/mm/dd<br>111 = aaaa/mm/dd  |
| 12      | 112                            | ISO  | 12 = aammdd<br>112 = aaaammdd      |
| -       | 13 ou 113 <sup>(1,2)</sup>     | Padrão Europa + milissegundos  | dd mês aaaa hh:mi:ss:mmm (24h)     |
| 14      | 114                            | -  | hh:mi:ss:mmm (24h)                 |
| -       | 20 ou 120 <sup>(2)</sup>       | ODBC canônico  | aaaa-mm-dd hh:mi:ss (24h)          |
| -       | 21 ou 25 ou 121 <sup>(2)</sup> | ODBC canônico (com milissegundos) padrão para hora, data, datetime2 e datetimeoffset | aaaa-mm-dd hh:mi:ss:mmm (24h)      |
| 22      | -                              | EUA  | mm/dd/aa hh:mi:ss AM (ou PM)       |

# Estilos de data e hora

|    |                       |                             |  |
|----|-----------------------|-----------------------------|--|
| 22 | -                     | EUA                         | mm/dd/aa hh:mi:ss AM (ou PM)   |
| -  | 23                    | ISO8601                     | aaaa-mm-dd   |
| -  | 126 <sup>(4)</sup>    | ISO8601                     | aaaa-mm-ddThh:mi:ss.mmm (sem espaços)<br><br><b>Observação:</b> Para um valor de milissegundos (mmm) igual a 0, o valor da fração decimal de milissegundo não será exibido. Por exemplo, o valor '2012-11-07T18:26:20.000' é exibido como '2012-11-07T18:26:20'. |
| -  | 127 <sup>(6, 7)</sup> | ISO8601 com fuso horário Z. | yyyy-MM-ddThh:mm:ss.fffZ (sem espaços)<br><br><b>Observação:</b> Para um valor de milissegundos (mmm) igual a 0, o valor decimal de milissegundo não será exibido. Por exemplo, o valor '2012-11-07T18:26:20.000' é exibido como '2012-11-07T18:26:20'.          |
| -  | 130 <sup>(1,2)</sup>  | Islâmico <sup>(5)</sup>     | dd mmm aaaa hh:mi:ss:mmmAM<br><br>Neste estilo, <b>mon</b> representa uma representação Unicode islâmico de vários tokens do nome completo do mês. Esse valor não é renderizado corretamente em uma instalação em inglês dos EUA padrão do SSMS.                 |
| -  | 131 <sup>(2)</sup>    | Islâmico <sup>(5)</sup>     | dd/mm/aaaa hh:mi:ss:mmmAM  |



# Conversão de dados

| From \ To        | binary | varbinary | char | nchar | nvarchar | datetime | smalldatetime | time | datetimeoffset | datetime2 | decimal | numeric | float | real | bigint | int(INT4) | smallint(INT2) | tinyint(INT1) | money | smallmoney | bit | timestamp | uniqueidentifier | image | ntext | text | sql_variant | xml | CLR UDT | hierarchyid |
|------------------|--------|-----------|------|-------|----------|----------|---------------|------|----------------|-----------|---------|---------|-------|------|--------|-----------|----------------|---------------|-------|------------|-----|-----------|------------------|-------|-------|------|-------------|-----|---------|-------------|
| binary           |        | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ✗     | ✗    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ✗     | ✗     | ●    | ●           | ●   | ●       |             |
| varbinary        | ●      |           | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ✗     | ✗    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ✗     | ✗     | ●    | ●           | ●   | ●       |             |
| char             | ●      | ●         |      | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ●     | ●     | ●    | ●           | ●   | ●       |             |
| varchar          | ●      | ●         | ●    |       | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ●     | ●     | ●    | ●           | ●   | ●       |             |
| nchar            | ●      | ●         | ●    | ●     |          | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ●     | ●     | ●    | ●           | ●   | ●       |             |
| nvarchar         | ●      | ●         | ●    | ●     | ●        |          | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ●                | ✗     | ●     | ●    | ●           | ●   | ●       |             |
| datetime         | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ✗                | ✗     | ✗     | ✗    | ●           | ✗   | ✗       |             |
| smalldatetime    | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ✗                | ✗     | ✗     | ✗    | ●           | ✗   | ✗       |             |
| date             | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ✗    | ●              | ●         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| time             | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ●    | ●              | ●         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| datetimeoffset   | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| datetime2        | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| decimal          | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ◆       | ◆       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| numeric          | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ◆       | ◆       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| float            | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| real             | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| bigint           | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| int(INT4)        | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| smallint(INT2)   | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| tinyint(INT1)    | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| money            | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| smallmoney       | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| bit              | ●      | ●         | ●    | ●     | ●        | ●        | ✗             | ✗    | ✗              | ✗         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| timestamp        | ●      | ●         | ●    | ●     | ✗        | ✗        | ✗             | ✗    | ✗              | ✗         | ●       | ✗       | ✗     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ✗         | ●                | ✗     | ✗     | ✗    | ✗           | ✗   |         |             |
| uniqueidentifier | ●      | ●         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ●         | ✗                | ●     | ●     | ✗    | ✗           | ✗   |         |             |
| image            | ●      | ●         | ✗    | ✗     | ✗        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ●         | ●                | ●     | ●     | ✗    | ✗           | ✗   |         |             |
| ntext            | ✗      | ✗         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ●                | ●     | ●     | ✗    | ●           | ✗   |         |             |
| text             | ✗      | ✗         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ●                | ●     | ●     | ✗    | ●           | ✗   |         |             |
| sql_variant      | ●      | ●         | ●    | ●     | ●        | ●        | ●             | ●    | ●              | ●         | ●       | ●       | ●     | ●    | ●      | ●         | ●              | ●             | ●     | ●          | ●   | ●         | ✗                | ✗     | ✗     | ●    | ✗           | ✗   |         |             |
| xml              | ●      | ●         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ○     | ○    | ○           | ○   |         |             |
| CLR UDT          | ●      | ●         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ○    | ○           | ○   |         |             |
| hierarchyid      | ●      | ●         | ●    | ●     | ●        | ✗        | ✗             | ✗    | ✗              | ✗         | ✗       | ✗       | ✗     | ✗    | ✗      | ✗         | ✗              | ✗             | ✗     | ✗          | ✗   | ✗         | ✗                | ✗     | ✗     | ✗    | ✗           | ✗   |         |             |

●

 Explicit conversion

●

 Implicit conversion

✗

 Conversion not allowed

◆

 Requires explicit CAST to prevent the loss of precision or scale that might occur in an implicit conversion.

○

 Implicit conversions between xml data types are supported only if the source or target is untyped xml. Otherwise, the conversion must be explicit.

**DESAFIO**



# Desafio

Queremos construir um SQL cujo resultado seja, para cada cliente: "O cliente João da Silva faturou 120000 no ano de 2016".

Somente para o ano de 2016.

```
SELECT
CONCAT('O cliente ', TC.NOME, ' faturou ',
CONVERT(VARCHAR, CONVERT(DECIMAL(15,2),
SUM(INF.QUANTIDADE * INF.[PRECO]))), ' no ano ',
CONVERT(VARCHAR, YEAR(NF.DATA))) AS SENTENCA FROM [TABELA DE NOTAS FISCAIS]NF
INNER JOIN [TABELA DE ITENS NOTAS FISCAIS] INF
ON NF.NUMERO = INF.NUMERO
INNER JOIN [TABELA DE CLIENTES] TC
ON NF.CPF = TC.CPF
WHERE YEAR(DATA) = 2016
GROUP BY TC.NOME, YEAR(DATA)
```

# O que aprendemos nesta aula

- As funções do SQL Server

- As funções do tipo string, de data e numéricas

- A converter dados.

