

What is an expression?.....	1
Business Math	2
Total Cost, Total Price.....	2
Dates and times	3
Converting date and time data to varchar format	4
DATEPART – when none of the standard styles fit your need.....	5
Case Statement – Flag some items.....	6

What is an expression?

https://en.wikipedia.org/wiki/Expression_%28computer_science%29

Adapting the Wikipedia definition:

An expression is a combination of one or more columns, constants, operators, datatypes, and functions that EQQ® and SQL interpret to produce another value.

New Name
Contact
Data Type
Text
Expression
[CCFirstName] + ' ' + [CCLastName]

This is a simple example. The user clicked/tapped the New Column button and entered what you see here.

The expression uses two columns [CCFirstName] and [CCLastName], one constant ' ' (a single space) and the + operator twice to combine them.

Results display and export as you would expect
Marie Curie
Jonas Salk

New Name
Date Created
Data Type
Text
Expression
CONVERT(varchar(10), [SODateTimeCreated], 101)

Here is another example.

[SODateTimeCreated] is a column. CONVERT is a function. varchar is a datatype. 10 is a constant specifying the number of characters to select. 101 is a constant specifying the U.S. mm/dd/yyyy date format.

Results display and export as you would expect:
07/04/2022
01/01/2023

Business Math

Total Cost, Total Price

These expressions are in the view named Demo View for User Guidance

Simple example. TotalQtyShipped = 10; UnitCost = 12.34; Total Cost = 123.40

You can probably get away with simply keying in

TotalQtyShipped * UnitCost

Specifying the table alias and using the brackets avoids ambiguity

New Name
Total Cost
Expression
ROUND([SOI].[TotalQtyShipped] * [SOI].[UnitCost], 2)

Simple example: TotalQtyShipped = 10; UnitPrice = 18.99; Total Price = 189.90

You can probably get away with simply keying in

TotalQtyShipped * UnitPrice

New Name
Total Price
Expression
ROUND([SOI].[TotalQtyShipped] * [SOI].[UnitPrice], 2)

The purpose of the ROUND function is to make sure the total is accurate to two decimal places.

Here is an example of why that is important.

Suppose the item is 2.3 yards of a specific fabric at \$12.99 per yard. $2.3 * \$12.99 = \29.877 .

Dates and times

You often need to deal with dates and times. And you often need to manipulate how those dates and times are displayed to the query result recipient.

Behind the scenes, dates and times are stored in SQL tables and Excel worksheets simply as numbers with no formatting. This makes it easy to work with them.

We will work with three distinct data types: Date, Time, and Datetime.

A date is just an integer (whole number) beginning on January 1, 1900. SQL can deal with earlier dates as negative numbers, but Excel cannot.

U.S. Independence Day	Numerically	Difference
07/04/1776	-45105	
07/04/1900	184	45289
07/04/1990	33056	32872
07/04/2019	43648	10592
07/04/2020	44014	366
07/04/2021	44379	365
07/04/2022	44744	365

Here are some Time examples in AMPM mode and 24-hour mode

SQL can capture and store time down to seconds and milliseconds, but we will focus on hours and minutes. For that, six decimal places are enough.

Time AMPM	Time 24 hour	Numeric Value
12:00 AM	0:00	0.000000
12:01 AM	0:01	0.000694
12:15 AM	0:15	0.010417
12:30 AM	0:30	0.020833
12:45 AM	0:45	0.031250
01:00 AM	1:00	0.041667
08:00 AM	8:00	0.333333
11:25 AM	11:25	0.475694
12:00 PM	12:00	0.500000
01:15 PM	13:15	0.552083
06:00 PM	18:00	0.750000
11:59 PM	23:59	0.999306

Put them together and we have some Datetime examples.

Datetime AMPM	Datetime 24 hour	Numeric Value
08/02/2021 12:00 AM	08/02/2021 00:00	44410.000000
08/02/2021 12:01 AM	08/02/2021 00:01	44410.000694
08/02/2021 12:15 AM	08/02/2021 00:15	44410.010417
08/02/2021 12:30 AM	08/02/2021 00:30	44410.020833
08/02/2021 12:45 AM	08/02/2021 00:45	44410.031250
08/02/2021 01:00 AM	08/02/2021 01:00	44410.041667
08/02/2021 08:00 AM	08/02/2021 08:00	44410.333333
08/02/2021 11:25 AM	08/02/2021 11:25	44410.475694
08/02/2021 12:00 PM	08/02/2021 12:00	44410.500000
08/02/2021 01:15 PM	08/02/2021 13:15	44410.552083
08/02/2021 06:00 PM	08/02/2021 18:00	44410.750000
08/02/2021 11:59 PM	08/02/2021 23:59	44410.999306

Converting date and time data to varchar format

SQL has its own default display format: Year, Month, Day, Hours, Minutes, Seconds, Milliseconds.

Technically accurate and understandable, but not suitable for most routine business use.

Here are three Datetime fields from the tSalesOrderHeader table showing the SQL default display and two of the most common U.S. business formats

Sales Order Header Id	[SODateTimeCreated] – SQL default display format	Normal business report	More detailed analysis
121004	2019-05-02 08:35:42.440	05/02/2019	05/02/2019 08:35
121005	2019-05-07 09:10:57.617	05/07/2019	05/07/2019 09:10
121006	2019-05-12 09:46:12.793	05/12/2019	05/12/2019 09:46
Format:	yyyy-mm-dd hh:mi:ss.nnn	mm/dd/yyyy	mm/dd/yyyy hh:mi

Excel knows a lot about date and time formats. And it is always ready to help in its own way.

Use the SQL Convert function to structure dates exactly the way the users of the Excel spreadsheet want them to be.

T-SQL has a set of 'styles' you can use to define how to display dates and times.

<https://docs.microsoft.com/en-us/sql/t-sql/functions/cast-and-convert-transact-sql?view=sql-server-ver15>

Here are the most relevant ones for common U.S. formats.

-	0 or 100 ^(1,2)	Default for datetime and smalldatetime	mon dd yyyy hh:miAM (or PM)
1	101	U.S.	1 = mm/dd/yy 101 = mm/dd/yyyy
14	114	-	hh:mi:ss:mmm (24h)
22	-	U.S.	mm/dd/yy hh:mi:ss AM (or PM)

The examples below specify Text as the output to lower the risk of unwanted help from Excel

The one on the left uses style 101.

The one on the right uses style 101 and 114 and for the latter needs to first convert the source field from Datetime to Time. Then it connects the two together with a space between: " ' ' +

New Name
Date Created
Data Type
Text
Expression
CONVERT(varchar(10), [SODateTimeCreated], 101)

New Name
Date-Time Created
Data Type
Text
Expression
CONVERT(varchar(10), [SODateTimeCreated], 101) + ' ' + CONVERT(varchar(5), CONVERT(time, [SODateTimeCreated]), 114))

And see some results below.

Supplier	Supplier State	Date Created	Date-Time Created
Eastern Microbrew Distributor, Inc.	IL	11/23/2019	11/23/2019 15:03
Western Craft Beer Distributor, LLC	CO	11/23/2019	11/23/2019 15:03
Western Craft Beer Distributor, LLC	CO	03/02/2020	03/02/2020 09:10

DATEPART – when none of the standard styles fit your need

Recipients of Excel exports you create often have their own very specific requirements in for date and time formats. They often will be importing the Excel workbook you deliver into an application that was written some time ago and with no knowledge of Windows or SQL.

You can use the DATEPART function to extract various parts of a date or datetime value.

<https://docs.microsoft.com/en-us/sql/t-sql/functions/datepart-transact-sql?view=sql-server-ver15>

datepart	Abbreviations
year	yy, yyyy
quarter	qq, q
month	mm, m
dayofyear	dy, y
day	dd, d
week	wk, ww
weekday	dw
hour	hh
minute	mi, n
second	ss, s
millisecond	ms

Here is a somewhat frivolous example to show how you can select parts of datetime values and rearrange them however you wish. You convert each part to varchar so you can string them together as shown.

New Name

Custom Date

Data Type

Text

Expression

```

CONVERT(varchar, (DATEPART (year, [SODateTimeCreated])))
+ '*' + CONVERT(varchar, (DATEPART (dayofyear, [SODateTimeCreated])))
+ '*' + CONVERT(varchar, (DATEPART (month, [SODateTimeCreated])))
+ '*' + CONVERT(varchar, (DATEPART (hour, [SODateTimeCreated])))
    
```

Supplier	Supplier State	Date Created	Date-Time Created	Custom Date
Eastern Microbrew Distributor, Inc.	IL	11/23/2019	11/23/2019 15:03	2019 * 327 * 11 * 15
Western Craft Beer Distributor, LLC	CO	11/23/2019	11/23/2019 15:03	2019 * 327 * 11 * 15
Western Craft Beer Distributor, LLC	CO	03/02/2020	03/02/2020 09:10	2020 * 62 * 3 * 9

Case Statement – Flag some items

Case statements are like If-Then-Else statements, but somewhat more flexible and easier to write. The structure is like this. New lines and indenting are just for visual convenience. SQL and EQQ do not care.

Case

When {expression} then {value}

When {expression} then {value}

When {expression} then {value}

Else {value}

End

Here is a simple example. Some items in some sales orders may come from a supplier that is not in the United States. Let's flag those items

Q						Scountry	Supplier Country	<input type="checkbox"/>
R							Import Duty	<input checked="" type="checkbox"/>

New Name
Import Duty
Data Type
Text
Expression
Case when [SCountry] <> 'USA' then 'May apply' else " End

Item Code	Supplier	Supplier State	Supplier Country	Import Duty
NE-CrP	New England Crafts Cooperative	ME	USA	
NM-NAC	New Mexico Craft Association	NM	USA	
Sca-HS	San Diego Swimwear, Inc.	CA	USA	
TH-Pewt	Thai Craft Exporters Ltd.		Thailand	May apply