

Re: Truncating Numbers

Source:

<http://www.tech-archive.net/Archive/Microsoft.Public.Access.Queries/2006-08/msg01134.html>

- *From:* FJquestioner <FJquestioner@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Sat, 29 Jul 2006 17:34:01 -0700
-

Thanks a lot Gary for the quick and helpful solution. I tried it on a couple of formulas and it worked like a charm.

Much appreciated.

"Gary Walter" wrote:

"FJquestioner" wrote:

I'm creating an accounting database.

I'm running into trouble when I calculate totals such as the amount I owe on several bills.

The system stores data to several decimal places and calculates totals based on these amounts rather than the rounded amounts (to 2 decimal places) that are actually displayed. Example:

Displayed Stored in System
Add this \$ 2.00 \$ 1.995

Plus this \$ 2.00 \$ 1.995

Totals this \$ 3.99 \$ 3.99

Does anyone know how can I get the system to store all calculated monetary fields to 2 decimal places, rounded. Or another way to resolve this issue?

Thanks

Re: Truncating Numbers

I have all monetary fields set to the "Currency" Data Type.

Most invoices consist of a units * price

I have a program that keeps track of tax rates for individual cities and counties in our state, and when we sell a product other than "in-store", we have to apply the appropriate tax for the city/county where that product will be sent.

At certain intervals, we have to provide the state with a report that sums these values "across and down."

Even though these values ($pSale * pTax$) are converted to Currency, they still contain 4 digits to the right of the decimal.

So, we might see \$12.35 (formatted to 2 decimals), but the number is still 12.3546.

When we sum them "across and down" those extra 2 digits at the end can cause the "across" sums to not match up with "down" sums.

So, when we compute

$Ccur(pSale * pTax)$

we want the 2 ending digits to always be "00"

There are 2 methods to round the computation.

1) $Ccur(Round(pSale * pTax, 2))$

con:

the Round function uses Banker's Rounding so it will round to the nearest even number in the specific case of $\$x.xx50$

for example,

$Ccur(Round(1.2850, 2)) = 1.2800$

BUT...we wanted it to round 1.2850 up to 1.2900

2) to round to $\$x.xx00$ and to always round $\$x.xx50$ UP, we use a formula such as:

Re: Truncating Numbers

$\text{CCur}(\text{Int}((\text{pSale} * \text{pTax}) / 0.01 + 0.5) * 0.01)$

Examples from Immediate Window:
(actually, the results in the window won't show ending zeroes, but I have added them to show what result would be if we had formatted the result to "#.0000")

```
pSale=CCur(64.25)
pTax=0.02
?pSale*pTax
1.285
?CCur(Round(pSale*pTax, 2))
1.2800
?CCur(Int((pSale * pTax) / 0.01 + 0.5) * 0.01)
1.2900
```

```
pSale=CCur(64.75)
pTax=0.02
?pSale*pTax
1.295
?CCur(Round(pSale*pTax, 2))
1.3000
?CCur(Int((pSale * pTax) / 0.01 + 0.5) * 0.01)
1.3000
```