

# Mod Function

## Description

Returns the modulo (remainder) of two expressions.

## Syntax

*modulo* = **Mod**(*expression*, *expression*)

## Parameters

The Mod function has the following parameters.

Parameter	Description
<i>expression</i>	Both expressions must be numeric and the second expression cannot be 0 (zero).

The modulo (remainder) is calculated by the following formula:

```
Mod(X,Y) = X - (Int(X/Y) * Y)
```

The first expression, X, is divided by Y and the Int function truncates the quotient. The resulting integer is multiplied by Y, the second expression, and that result is subtracted from the first expression X.

The following example results in the value 4 assigned to the variable identifier COST:

```
COST = Mod(16,6)
```

The expressions must be of numeric value, and the second expression cannot be 0 (zero).

Mod will calculate a remainder after division by a divisor. This is useful when working with complete and incomplete sets.

## Example

```
* Value is assigned the value of 9.
A = 99
B = 10
VAL = Mod(A, B)
/* This code segment can be used to determine the day of the week based on a date in internal format. */
/* The system date is returned and is divided by 7.
The Case statement swaps the remainder with the day of the week.
Since the result of the program is assigned to @ANS,
this program could be modified and placed in a dictionary record. */
Date = Date()
DAY = Mod(Date, 7)
Begin Case
  Case DAY = 0 ; DAY = "SUNDAY"
  Case DAY = 1 ; DAY = "MONDAY"
  Case DAY = 2 ; DAY = "TUESDAY"
  Case DAY = 3 ; DAY = "WEDNESDAY"
  Case DAY = 4 ; DAY = "THURSDAY"
  Case DAY = 5 ; DAY = "FRIDAY"
  Case DAY = 6 ; DAY = "SATURDAY"
End Case
@ANS = DAY
```