

# Rounding numbers

## Overview

You can put the relevant expression into a hidden value that represents the integer form of the value.

### Round a number down to the nearest integer

If you want to round a number down you can use the int() function to cut off the decimal point (i.e. 2.2 -> 2; 2.7 -> 2)

```
int(#form/value)
```

### Round a number up to the nearest integer

Use the following calculation to always round a number up (i.e. 2.2 -> 3; 2.7 -> 3)

```
If( int(#form/value) < #form/value, int(#form/value) + 1, int(#form/value))
```

This compares whether the decimal form of the lowest integer is smaller than the current value, and if so, rounds up, and otherwise truncates the value.

### Round a number up from .5, otherwise down

As of CommCare 2.19, the "round" function is available. Simply write round(#form/value). Read more [here](#).

You can also use the int() function in combination with multiplying the input by 2, adding 1, and then dividing by 2 (i.e. 2.2 -> 2; 2.7 -> 3)

```
int ( ( #form/value*2 +1 ) div 2 )
```

To round to a different number of decimal places, appropriately adjust the constant '2' in the preceding calculation. The constant should be twice the inverse of the desired precision. For example, for a precision of 0.1, constant = 1 / precision \* 2 = 1 / 0.1 \* 2 = 20. Consequently use int ( ( #form/value\*20 +1 ) div 20 ).

### Round a number to a certain decimal place

You can use the same principles to round to the nearest .1, .01, .001, etc. with the following formula:

Round to nearest decimal: **round(#form/value\*10) div 10**

To round to two decimals change the 10's in the formula above to 100. To three decimals change them to 1000, etc.