

Calculations with Dates and Times

This page contains:

- [Overview](#)
- [Examples](#)
- [Converting date values to type Date](#)
- [Doing calculations with the now\(\) function](#)
- [Date Formatting](#)
- [Calculations using time question types](#)
- [Additional Resources](#)

Overview

In CommCare you can do calculations with dates and times just like you would with numbers or strings, and there are a few CommCare Functions that are useful for doing so. The key ones explained here are:

<ul style="list-style-type: none">• <code>date()</code> - Takes an input and converts it to be of type date (input must be of type string, numeric, or date)	
<ul style="list-style-type: none">• <code>today()</code> - Returns the current date, according to the mobile device.	For Web Apps it returns the current server date, but in the browser's time zone. E.g.: <pre>server time = 2021-10-18 00:01 UTC browser time zone = ET today() = 2021-10-17</pre>
<ul style="list-style-type: none">• <code>now()</code> - Returns the current date and time, according to the mobile device.	For Web Apps it returns the current server date/time value, but in the browser's time zone. E.g.: <pre>server time = 2021-10-18 11:50 UTC browser time zone = ET now() = 2021-10-18 06:50 ET</pre>

More details about these functions, and other functions that may be useful for performing calculations with dates, can be found [here](#).

For further guidance on doing calculations with dates and times, see [Hidden Values Tutorial Part 2: Calculations Using a Date](#).

Examples

- Age in years (estimate):

```
int((today() - date(#form/dob)) div 365.25)
```

- Age in years (precise):

```
if(format-date(today(), "%m%d") >= format-date(#form/dob, "%m%d"),  
  format-date(today(), "%Y") - format-date(#form/dob, "%Y"),  
  format-date(today(), "%Y") - format-date(#form/dob, "%Y") - 1)
```

- Age in months:

```
int((today() - date(#form/dob)) div 30.4)
```

- Age in months (by day of month)

```
(format-date(today(), "%Y") - format-date(#form/dob, "%Y")) * 12
+ (format-date(today(), "%m") - format-date(#form/dob, "%m"))
+ if(format-date(today(), "%d") < format-date(#form/dob, "%d"), -1, 0)
```

- Age in weeks:

```
int((today() - date(#form/dob)) div 7)
```

- Estimated Date of Delivery (EDD) from Last Menstrual Period (LMP):

```
date(#form/lmp + 280)
```

- This is what a DOB calculation might look like in the Form Builder; the logic statement goes in the "Calculate Condition" box:

Hidden Value Delete

Calculate Condition: ✎

Question ID:

- This is an example of a hidden value where the Estimated Date of Delivery (EDD) is being calculated. The logic statement goes in the "Calculation Condition" box:

Hidden Value Delete

Calculate Condition: ✎

Question ID:

- The following shows how to restrict a Date question's answer to be within a certain range:

Please enter a date:

Date Delete

Display Text: ✎

Question ID:

Required:

Logic ?

Display Condition: ✎

Validation Condition: ✎

Validation Message: ✎

Default Value: ✎

Converting date values to type Date

One important concept to understand about dates in CommCare is that a date value can often be stored in a string or numeric format, rather than as a Date type, *but* any date computations need to be performed on arguments that are actually of the type date. So when you have a date value that may be of type string or numeric rather than of type date, you can use `date()` to convert it before doing computations. Some examples:

- When doing arithmetic on a date from a "Date" type question, you can do the arithmetic directly on that value since it's already a date. But at the end, you should convert the entire result to a date. So if `#form/lmp` is a "Date" type question, you can compute the EDD using the following expression: `date(#form/lmp + 280)`
- When doing arithmetic on a date from a "Hidden Value", you need to convert the value stored in the hidden value to a date first, then do the arithmetic, and then convert the result to a date. So if `#form/lmp` is a hidden value that contains the LMP, then to compute the EDD you should use the following expression: `date(date(#form/lmp) + 280)`

Doing calculations with the `now()` function

In CommCare, you can use the `now()` function to get the current date *and* time.

To perform any computations using `now()`, make sure to consider the following:

Because the result of `now()` contains time data in addition to date data, you cannot do math or comparisons on this data type directly, and you also should *not* convert it to a date using `date()`, or the time portion will be dropped. Instead, you must first convert it to a numeric representation, using the function `double()`, and then do any desired computations.

Similarly, if you are saving the value of `now()` as a case property, you must convert it to a number *before* saving it as a case property, i.e. `double(now())`. This will convert the `now()` value to a number that represents the date/time in days since January 1, 1970. **Saving `now()` directly will only save the date portion without the time portion.**

As an example of doing computations using `now()`, suppose you want to compute the amount of time a user spends doing some activity/workflow. You could achieve this by doing the following:

- In the form that represents the beginning of the workflow, you can capture the full date and time in a hidden value with the calculation:

```
double(now())
```

- Save that hidden value to the case as a case property
- In the form that represents the end of the workflow, you can load the value you saved in the previous form (let's call it `#form/start_time`)
- Then to compute the time elapsed, you would use:

```
double(now()) - double(#form/start_time)
```

The result is a decimal value in days/fractions of days. To convert to the # of hours (rounded down), you could use the following calculation:

```
int(#form/time_worked * 24)
```

To convert `now()` to a date and time in excel:

- First convert it to a decimal using the "double()" function. The double function will convert the `now()` value to a number that represents the date/ time in days since 1/1/1970. This is similar to doing calculations on dates, when you use `int()` to convert a date value into number of days.
- Save this as a case property or use it as a hidden value in form exports
- After exporting this value to excel, you can convert it to a date and time by 1/1/1970 to it
In excel, if you want to take that value and show the current time (in UTC), you simply add it to 1/1/1970 to it.
- Time values are stored as strings, and require similar manipulation to do any validation conditions or comparisons in calculations. For example, here is the validation on a 'time' question type if you wanted to make sure the answers were between 9am and 9pm:
 - `int(substr(., 0, 2)) >= 9 and int(substr(., 0, 2)) < 21`
 - To convert a time saved to a case into HH:MM AM/PM so it can be displayed in the app, create an additional hidden value with the following calculate condition:
 - `if(int(substr(#form/time_value, 0, 2)) < 12, concat(substr(#form/time_value, 0, 5), ' AM'), if(int(substr(#form/time_value, 0, 2)) = 12, concat(substr(#form/time_value, 0, 5), ' PM'), concat(int(substr(#form/time_value, 0, 2)) - 12, substr(#form/time_value, 2, 5), ' PM')))`
 - For more information on `substr` and other string functions, see [CommCare Functions#substr](#)
- When using `double()` with values from other questions to do calculations you may need to cast the question as a date. For example: `double(date(#form/somequestion))`

To extract a nicely-formatted time from `now()`:

Create a hidden value called **double_now** to store the value of `now()`:

#form/double_now

```
double(now())
```

Create a hidden value called **seconds_since_midnight** to store only the time, represented as the number of seconds since midnight:

#form/seconds_since_midnight

```
(#form/double_now - floor(#form/double_now)) * 24 * 60 * 60
```

You can then create an **hours** hidden value:

#form/hours

```
floor(#form/seconds_since_midnight div 60 div 60)
```

And a **minutes** hidden value:

#form/minutes

```
floor((#form/seconds_since_midnight - (#form/hours * 60 * 60)) div 60)
```

And a **seconds** hidden value:

#form/seconds

```
floor(#form/seconds_since_midnight mod 60)
```

And finally display the time nicely:

#form/time_formatted

```
concat(#form/hours, ':', #form/minutes, ':', #form/seconds)
```

Note that this will display the time in 24-hour format and will not zero-pad, so 2:05:30 in the afternoon would display as 14:5:30. You can use if statements in the formatted time calculation to address this.

Date Formatting

To convert or format a date into another format (e.g. day of week, name of month, etc.) see the [format-date](#) function.

When you enter day, month, and year in as three separate numerical responses, you can convert those three responses into one date as follows

- First, remember that dates are typically saved in the format YYYY-MM-DD
- This means that the number entered for month and the number entered for day must be a two-digit value
- To ensure that the format is correct, create hidden values for month and day using the calculate condition of: `if(data/month<10, concat('0', data/month), data/month)` or `if(data/day<10, concat('0', data/day), data/day)` to ensure that if someone types in a single-digit day, a zero will be added in front of it to create a two-digit number
- To create the date using the three inputs, create another hidden value (we'll call this example `calc_date`), and use the following syntax:
`if(data/day != "" and data/month != "" and data/year != "", concat(data/year, '-', data/hiddenvaluemonth, '-', data/hiddenvalueday), "")`
 - NOTE: the reason we have to use an if statement is because otherwise, you will return a type mismatch error. This will happen because until the day, month, and year are entered into their respective questions, the `calc_day` command will try to create a date but cannot because the values from those questions will be blank. To avoid that, the if statement tells the `calc_date` to only calculate the date if the day, month, and year questions are NOT blank. If they are blank, the if statement tells the `calc_day` to return a blank value.

Calculations using time question types

You can use time collected in forms to do different calculations, e.g. the number of hours between the time of birth and now. Some things to keep in mind when doing these calculations:

- When you use "double(#form/dob)", the value you get back is a number like "16925", which is the number of days it's been since January 1st, 1970.
- When you use "double(now())", the value is similar but contains a decimal, like "16925.50833333", which is the same number of days, but it also includes the time, represented as a fraction of a day. So "16925" is May 4th, 2016, and doesn't include time information. Whereas "16925.5" is May 4th, 2016, at noon (halfway through the day), and "16925.75" is May 4th, 2016 at 6:00 p.m. "16925.50833333" happens to be May 4th, 2016 at 12:12 p.m.

To perform calculations with the results of date & time questions and now(), you need to first convert the values from the questions into decimals, in order for them to be compatible with double(now()).

In the following example, we show how to compute the number of hours since time of birth (this is done assuming that you are collecting date and time as 2 different question types to trigger different calculations; similar calculations can also be triggered using the date/time question type, which collects both together):

1. Collect date and time of birth in date and time questions in your form (call these #form/dob and #form/time_of_birth)
2. Calculate minutes since midnight for the time of birth (call this #form/minutes_since_midnight_birth):

```
substr(#form/time_of_birth, 0, 2) * 60 + substr(#form/time_of_birth, 3, 5)
```

3. Convert minutes since midnight into a decimal, by dividing it by the total number of minutes in a day (call this #form/time_of_birth_decimal):

```
#form/minutes_since_midnight div 1440
```

4. Sum that numeric time with a numeric version of the dob. This will get you to a numeric datetime value like 16925.50833333 (call this #form/birth_datetime):

```
double(#form/dob) + #form/time_of_birth_decimal
```

5. Then, in the form where you need the hours calculation, subtract your datetime from double(now()) and convert to hours (call this #form/hours):

```
(double(now()) - #form/birth_datetime) * 24
```

6. If you wish, you can also use the round() function to output a rounded # of hours:

```
round(#form/hours)
```

Additional Resources

- More details on EDD calculations: <http://www.perinatology.com/calculators/Due-Date.htm> and <http://nrhm-mctsrpt.nic.in/MHFW/UI/FAQAnc.aspx>