

# SubTimesLocal

## %SubTimesLocal function

Function	The function deducts absolute time <b>TimeB</b> from absolute time <b>TimeA</b> . The result is value of <i>Relative time</i> type.						
Declaration	<pre>REAL %SubTimesLocal(     TIME in TimeA,     TIME in TimeB,     TEXT in timeZone := %GetCurrentTimeZone() )</pre>						
Parameters	<table><tr><td><b>T i m eA</b></td><td>Absolute time.</td></tr><tr><td><b>T i m eB</b></td><td>Absolute time.</td></tr><tr><td><b>ti m e Z o ne</b></td><td>Name of the time zone used for conversion to local time (e.g. "Europe/London") or definition of fixed offset from UTC using format "<i>(+/-)hh[:mi[:ss]]</i>", where <i>hh</i> defines a number of hours, <i>mi</i> defines a number of minutes, and <i>ss</i> defines a number of seconds. Sign as well as a number of hours are mandatory parts of offset definition, number of minutes and seconds are optional and default to 0 (e.g. "+02:30" defines offset of 2 hours and 30 minutes from UTC). The empty text has the same meaning as function <a href="#">%GetCurrentTimeZone</a>. Note: For historical reasons, an integer parameter is also accepted. Its interpretation is as follows: 0 - zone "Europe/London", 3600 - zone "Europe/Bratislava", 7200 - zone "Europe/Kiev", 21600 - zone "Asia/Almaty". Usage of integer parameter is deprecated and generates warning into log file!</td></tr></table>	<b>T i m eA</b>	Absolute time.	<b>T i m eB</b>	Absolute time.	<b>ti m e Z o ne</b>	Name of the time zone used for conversion to local time (e.g. "Europe/London") or definition of fixed offset from UTC using format " <i>(+/-)hh[:mi[:ss]]</i> ", where <i>hh</i> defines a number of hours, <i>mi</i> defines a number of minutes, and <i>ss</i> defines a number of seconds. Sign as well as a number of hours are mandatory parts of offset definition, number of minutes and seconds are optional and default to 0 (e.g. "+02:30" defines offset of 2 hours and 30 minutes from UTC). The empty text has the same meaning as function <a href="#">%GetCurrentTimeZone</a> . Note: For historical reasons, an integer parameter is also accepted. Its interpretation is as follows: 0 - zone "Europe/London", 3600 - zone "Europe/Bratislava", 7200 - zone "Europe/Kiev", 21600 - zone "Asia/Almaty". Usage of integer parameter is deprecated and generates warning into log file!
<b>T i m eA</b>	Absolute time.						
<b>T i m eB</b>	Absolute time.						
<b>ti m e Z o ne</b>	Name of the time zone used for conversion to local time (e.g. "Europe/London") or definition of fixed offset from UTC using format " <i>(+/-)hh[:mi[:ss]]</i> ", where <i>hh</i> defines a number of hours, <i>mi</i> defines a number of minutes, and <i>ss</i> defines a number of seconds. Sign as well as a number of hours are mandatory parts of offset definition, number of minutes and seconds are optional and default to 0 (e.g. "+02:30" defines offset of 2 hours and 30 minutes from UTC). The empty text has the same meaning as function <a href="#">%GetCurrentTimeZone</a> . Note: For historical reasons, an integer parameter is also accepted. Its interpretation is as follows: 0 - zone "Europe/London", 3600 - zone "Europe/Bratislava", 7200 - zone "Europe/Kiev", 21600 - zone "Asia/Almaty". Usage of integer parameter is deprecated and generates warning into log file!						
Description	The function %SubTimesLocal deducts one absolute time from another one. The result <b>will not be</b> influenced by the fact if the time offset was or was not in this interval. For example, the difference between 4:00 and 1:00 a.m. is always three hours.						
Example							

```

; =====
; Using time zone "Europe/Bratislava"
; Daylight saving time was observed in year 2009, therefore time shifts
occured.
; Winter [B] time was 1 hour ahead of UTC, summer [A] time was 2 hours
ahead of UTC.
; Summer time was from 29th March 2009 to 24th October 2009.
; A3:00:00 is changed to B2:00:00 on Sunday (25th October 2009).
; =====

BEGIN
    TIME _timeA
    TIME _timeB
    TIME _baseTime

    REAL _subLocal1
    REAL _subLocal2

    ; 2009-10-25 00:30:00 UTC
    _timeA := %StrToTimeEx("2009-10-25 A2:30:00", "yyyy-mm-dd hh:mi:ss",
"Europe/Bratislava")
    ; 2009-10-25 01:30:00 UTC
    _timeB := %StrToTimeEx("2009-10-25 B2:30:00", "yyyy-mm-dd hh:mi:ss",
"Europe/Bratislava")
    ; 2009-10-24 23:30:00 UTC
    _baseTime := %StrToTimeEx("2009-10-25 01:30:00", "yyyy-mm-dd hh:mi:ss",
"Europe/Bratislava")

    ; difference 1 hour(s) (3600 seconds)
    _subLocal1 := %SubTimesLocal(_timeA, _baseTime, "Europe/Bratislava")
    ; difference 1 hour(s) (3600 seconds)
    _subLocal2 := %SubTimesLocal(_timeB, _baseTime, "Europe/Bratislava")

END

```



#### Related pages:

[Implemented functions](#)  
[Function arguments - types](#)  
[%SubTimesMono](#)