CHANGE DATE AND TIME FORMAT Readme

Bot Overview

This bot performs various date time operations. The function of each logic in the Metabot are listed in the More Information Section-1.

Pre-Requisites

- Automation Anywhere Enterprise v10.7 sp2

Installation

- Download the bot from Bot Store.
- Double click on <Bot Name>.msi and follow the installation instructions.

  For first time users, the "Bot Store" folder is created under <AA Directory>/My Tasks (on your local disk).

- ***Installer creates the following folder structure with content under the <AA Directory>***

- <AA Directory>
  - My Tasks
    - Bot Store
      - AA_Date Time (Folder)
        - My Tasks
          - DateTimeLib Stub_Empty.atmx
        - Error Folder
          - Logs (Folder)
            - Error Logs Month-Day-Year.txt
          - Snapshots (Folder)
            - Error Snap Month-Day-Year.png
  - My Metabot
    - DateTime.mbot

How to Use the Bot:

Use the following information to configure your bot:

Parameter Name	Type	Direction	Additional Info
vStr	String	Input	Provide the present-day Date Ex: 11/26/2018 10:56:50 AM
vOffset	String	Input	Provide number of days/months/min/seconds to add. Ex:2
vFormat	String	Input	Provide the format specifiers. Refer to More Information section- 2 for details. Ex: F
vStr	String	Output	Provide a variable in which the output of the executed logic needs to be stored. Ex: $Prompt-Assignment

vErrorFolder	String	Input	This is error folder inside bot folder which contains Logs and Snapshots folder (Already defined, no need to add value unless want to change location of file).
vLogFolder	String	Input	This folder contains Log file in case of error (Already defined, no need to add value unless want to change location of file).
vSnapshotFolder	String	Input	This folder contains Screenshot in case of error (Already defined, no need to add value unless want to change location of file).

Note:

1. If the input string cannot be parsed, an exception will be raised.
2. Negative offsets can be used to subtract.
3. Check '*More Information*' section below to know more about date format supported.

4. The metabot contains many logics which take the inputs as mentioned above. User can choose the required logic and provide necessary inputs.
5. Elapsed time logics can be used in parallel with Start/Stop timer logics.
6. In this bot, the input string is the present-day date and the output variable is displayed on the $Prompt-assignment variable in the variable list.

Error Handling

- Each Bot folder contains the below hierarchy.
    - Error Folder
        - Logs
            - Error Logs Month-Day-Year.txt: In case of any error, this file logs error message along with time stamp.
        - Snapshots
            - Error Snap Month- Day-Year.png: In case of any error, this file captures screenshot of error.
- Task Status of bot is set to failed in case of error.

More Information

1. Functions in the Metabot –

Function	Inputs	Outputs	Comments
GetDateTime	str – Date/Time*	str – Date/Time	Accepts a string and attempts to interpret the string as a Date/Time value. Returns the standard system Date/Time string.
Day	str – Date/Time*	Str – The day component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the day component of the Date/Time.
Month	str – Date/Time*	str – The month component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the month component of the Date/Time.
Year	str – Date/Time*	str – The year component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the year component of the Date/Time.
DayOfWeek	str – Date/Time*	str – The day of the week of the Date/Time provided ("Monday", "Tuesday", etc.).	Accepts a string and attempts to interpret the string as a date/time value. Returns the day of the week of the Date/Time.
DayOfYear	str – Date/Time*	str – The Julian day of the year of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the day of year of the Date/Time.
Hour	str – Date/Time*	str – The hour component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the hour component of the Date/Time.
Minute	str – Date/Time*	str – The minute component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the minute component of the Date/Time.
Second	str – Date/Time*	str – The second component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the second component of the Date/Time.
Millisecond	str – Date/Time*	str – The millisecond component of the Date/Time proved.	Accepts a string and attempts to interpret the string as a date/time value. Returns the millisecond component of the Date/Time.
TimeOfDay	str – Date/Time*	str – The time component of the Date/Time provided.	Accepts a string and attempts to interpret the string as a date/time value. Returns the time component of the Date/Time.
Now		str – The current Date/Time.	Returns a standard system Date/Time string of the current date & time.
LocalToUniversalTime	str – Date/Time*	str – Local Date/Time provided converted to UTC.	Accepts a string and attempts to interpret the string as a date/time value (local time). Returns the UTC equivalent Date/Time string.
UniversalToLocalTime	str – Date/Time*	str – The UTC Date/Time provided converted to Local.	Accepts a string and attempts to interpret the string as a date/time value (UTC time). Returns the local equivalent Date/Time string.
IsLeapYear	str – Date/Time*	str – "True" or "False"	Accepts a string and attempts to interpret the string as a date/time value. Returns "True" or "False" if the year of the Date/Time is a leap year.
IsDaylightSavingTime	str – Date/Time*	str – "True" or "False"	Accepts a string and attempts to interpret the string as a date/time value. Returns "True" or "False" if the date/time is adjusted for Daylight Saving Time.

DaysInMonth	str - Date/Time*	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Returns number of days in the month of the date/time.
AddDays	str - Date/Time* offset – The number of days to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of days to the date/time and returns the resulting date/time string.
AddMonths	str - Date/Time* offset – The number of months to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of months to the date/time and returns the resulting date/time string.
AddYears	str - Date/Time* offset – The number of years to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of years to the date/time and returns the resulting date/time string.
AddHours	str - Date/Time* offset – The number of hours to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of hours to the date/time and returns the resulting date/time string.
AddMinutes	str - Date/Time* offset – The number of minutes to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of minutes to the date/time and returns the resulting date/time string.
AddSeconds	str - Date/Time* offset – The number of seconds to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of seconds to the date/time and returns the resulting date/time string.
AddMilliseconds	str - Date/Time* offset – The number of milliseconds to add**	str – Date/Time	Accepts a string and attempts to interpret the string as a date/time value. Adds the specified number of milliseconds to the date/time and returns the resulting date/time string.
FormatStringWithDateTime	str – Date/Time* format – a C#.net Date/Time format string.	str – The formatted string.	Accepts a string and attempts to interpret the string as a date/time value. Formats the date/time as specified in the format string and returns the resulting string. The format string uses the c#.NET Standard Date and Time Format Strings and Custom Date and Time Format Strings
Compare	str1 – Date/Time* str2 – Date/Time*	str1 – “Equal”, “Later” or “Earlier”.	Returns “Equal” if the dates are the same, “Later” if the date in str2 is after the date in str1, or “Earlier”. if the date in str2 is before the date in str1.
Diff	str1 – Date/Time* str2 – Date/Time*	str1 - TimeSpan	Returns a TimeSpam (±DDD.HH:MM:SS.mmmmmm) showing the difference between str1 and str2.
DiffTotalDays	str1 – Date/Time* str2 – Date/Time*	str1 -	Returns total number of days between str1 and str2.
DiffTotalHours	str1 – Date/Time* str2 – Date/Time*	str1 -	Returns total number of hours between str1 and str2.
DiffTotalMinutes	str1 – Date/Time* str2 – Date/Time*	str1 -	Returns total number of minutes between str1 and str2.
DiffTotalSeconds	str1 – Date/Time* str2 – Date/Time*	str1 -	Returns total number of seconds between str1 and str2.
DiffTotalMilliseconds	str1 – Date/Time* str2 – Date/Time*	str1 -	Returns total number of milliseconds between str1 and str2.
FirstDayOfMonth	str – Date/Time*	str – Date/Time	Returns First Day Of Month for input Date

LastDayOfMonth	str – Date/Time*	str – Date/Time	Returns Last Day Of Month for input Date
FirstDayOfNextMonth	str – Date/Time*	str – Date/Time	Returns First Day Of Next Month for input Date
LastDayOfNextMonth	str – Date/Time*	str – Date/Time	Returns Last Day Of Next Month for input Date
FirstDayOfPreviousMonth	str – Date/Time*	str – Date/Time	Returns First Day Of Previous Month for input Date
LastDayOfPreviousMonth	str – Date/Time*	str – Date/Time	Returns Last Day Of Previous Month for input Date
FirstDayOfYear	str – Date/Time*	str – Date/Time	Returns First Day Of Year for input Date
LastDayOfYear	str – Date/Time*	str – Date/Time	Returns Last Day Of Year for input Date
FirstDayOfQuarter	str – Date/Time*	str – Date/Time	Returns First Day Of Quarter for input Date
LastDayOfQuarter	str – Date/Time*	str – Date/Time	Returns Last Day Of Quarter for input Date
MaxValue		str – Date/Time	Returns Max value for DateTime – 12/31/9999 11:59:59 PM
MinValue		str – Date/Time	Returns Min value for DateTime – 01/01/0001 12:00:00 AM
MaxSQLServerDateTime		str – Date/Time	Returns Max value for SQL Server DateTime Type – 12/31/9999 11:59:59 PM
MinSQLServerDateTime		str – Date/Time	Returns Min value for SQL Server DateTime Type – 01/01/1753 12:00:00 AM

2. Date/Time Standard Format Specifiers (uses local culture settings, shown here for en-US)

Format Specifier		DateTime	Result
d	Short Date	4/9/2017 5:05:02.125 PM	4/9/2017
D	Long Date	4/9/2017 5:05:02.125 PM	Sunday, April 9, 2017
f	Full Date/Short Time	4/9/2017 5:05:02.125 PM	Sunday, April 9, 2017 5:05 PM
F	Full Date/Long Time	4/9/2017 5:05:02.125 PM	Sunday, April 9, 2017 5:05:02 PM
g	General Date/Short Time	4/9/2017 5:05:02.125 PM	4/9/2017 5:05 PM
G	General Date/Long Time	4/9/2017 5:05:02.125 PM	4/9/2017 5:05:02 PM
M or m	Month Day	4/9/2017 5:05:02.125 PM	April 9
s	Sortable	4/9/2017 5:05:02.125 PM	2017-04-09T17:05:02
t	Short Time	4/9/2017 5:05:02.125 PM	5:05 PM
T	Long Time	4/9/2017 5:05:02.125 PM	5:05:02 PM
Y or y	Year Month	4/9/2017 5:05:02.125 PM	April, 2017

Date/Time Custom Format Specifiers

Format Specifier	Meaning	DateTime	Result
d	Day	4/9/2017 5:05:02.125 PM	9
dd	Day w/leading zero	4/9/2017 5:05:02.125 PM	09
ddd	Day name abbreviation	4/9/2017 5:05:02.125 PM	Sun
dddd	Day name	4/9/2017 5:05:02.125 PM	Sunday

M	Month	4/9/2017 5:05:02.125 PM	4
MM	Month w/leading zero	4/9/2017 5:05:02.125 PM	04
MMM	Month name abbreviation	4/9/2017 5:05:02.125 PM	Apr
MMMM	Month name	4/9/2017 5:05:02.125 PM	April
yy	Year – 2 digit	4/9/2017 5:05:02.125 PM	17
yyyy	Year – 4 digit	4/9/2017 5:05:02.125 PM	2017
h	Hour – 12 hour	4/9/2017 5:05:02.125 PM	5
hh	Hour – 12 hour w/leading zero	4/9/2017 5:05:02.125 PM	05
H	Hour – 24 hour	4/9/2017 5:05:02.125 PM	17
HH	Hour – 24 hour	4/9/2017 5:05:02.125 PM	17
m	Minute	4/9/2017 5:05:02.125 PM	5
mm	Minute w/leading zero	4/9/2017 5:05:02.125 PM	05
s	Second	4/9/2017 5:05:02.125 PM	2
ss	Second w/leading zero	4/9/2017 5:05:02.125 PM	02
f	Fractional seconds	4/9/2017 5:05:02.125 PM	1
ff	Fractional seconds	4/9/2017 5:05:02.125 PM	12
fff	Fractional seconds	4/9/2017 5:05:02.125 PM	125
ffff	Fractional seconds	4/9/2017 5:05:02.125 PM	1250
fffff	Fractional seconds	4/9/2017 5:05:02.125 PM	12500
ffffff	Fractional seconds	4/9/2017 5:05:02.125 PM	125000
fffffff	Fractional seconds	4/9/2017 5:05:02.125 PM	1250000
F	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	1
FF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	12
FFF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	125
FFFF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	125
FFFFF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	125
FFFFFF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	125
FFFFFFF	Fractional seconds if not zero	4/9/2017 5:05:02.125 PM	125

Sample custom format strings:

Date	Format String	Result
4/9/2017 5:05:02.125 PM	“The day is “dddd” abbreviated “ddd”.	The day is Sunday abbreviated Sun.
4/9/2017 5:05:02.125 PM	MMMM” is month number “m” of the year.”	April is month number 4 of the year.
4/9/2017 5:05:02.125 PM	“This is going to look odd: “h:m:s	This is going to look odd: 5:5:2
4/9/2017 5:05:02.125 PM	“This is going to look odd: “H:m:s	This is going to look odd: 17:5:2
4/9/2017 5:05:02.125 PM	“This is probably better: “H:mm:ss	This is probably better: 17:05:02