

Data Extraction and Format

-

Functions description

DOUCMENT REVISION:

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Name | Description |
| 19/04/2019 | V1.0 | Romain Alexandre | First draft |

Contsents

[Introduction 2](#_Toc6562750)

[Notes 2](#_Toc6562751)

[Assets / Logics 3](#_Toc6562752)

# Introduction

Data Extraction and Format metabot can be used to extract and format data from a source text value.

Mainly used with IQ Bot extraction it can be call from any taskbot dealing with text inputs and outputs. If some of the functionalities can be performed using multiple RPA commands one of the targets of this metabot is to reduce number of RPA commands and make a task bot simpler and easier to maintain.

# Notes

1. No prerequisite for this metabot, it should be self-contained
2. Using certain features of this metabot require knowledge of Regular Expression, what a Regular Expression Pattern is and what Regular Expression Matching Groups are. A great place to start exploring Regular Expressions (RegEx for short) is regex101.com
3. Find all formats available [here](https://docs.microsoft.com/en-us/dotnet/standard/base-types/formatting-types).

# Assets / Logics

Data Extraction and Format metabot contains one library: FormatUtility.dll, source code can be found here.

|  |  |  |  |
| --- | --- | --- | --- |
| Logic Name | Inputs | Outputs | Comments |
| Get Extract and Format Value | String – SourceValue  String – RegexToExtract  String – ValuesToReplace  Int – TrimValue  String – ValueType  String – ValueFormat  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  RegexToExtract: Regex pattern to extract from original value  ValuesToReplace: use "," to separate old and new value and ";" to separate occurrences (e.g old1","new1";"old2","new2)  TrimValue: 1=yes; 0=no  ValueType: numeric ; date ; text  ValueFormat: e.g. dd/MM/yyyy ; #,0.00 ; ...  ValueIfError: value returned if error in logic  This logic is the combination of all logics following in document |
| Get Extracted Text Match Regex | String – SourceValue  String – Regex  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  RegexToExtract: Regex pattern to extract from original value  ValueIfError: value returned if error in logic |
| Get Formatted Date | String – SourceValue  String – TargetFormat  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  TargetFormat: Date format e.g. dd/MM/yyyy  ValueIfError: value returned if error in logic |

|  |  |  |  |
| --- | --- | --- | --- |
| Logic Name | Inputs | Outputs | Comments |
| Get Formatted Date Last Day Of Month | String – SourceValue  String – TargetFormat  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  TargetFormat: Date format e.g. dd/MM/yyyy  ValueIfError: value returned if error in logic  Same as previous logic but returns date with last day of the month |
| Get Formatted Date With Date Operation | String – SourceValue  String – TargetFormat  String – ValueIfError  Int – AddDayNumber  Int – AddMonthNumber  Int – AddYearNumber  Int – AddHourNumber  Int – AddMinuteNumber  Int – AddSecondsNumber | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  TargetFormat: Date format e.g. dd/MM/yyyy  ValueIfError: value returned if error in logic  AddDayNumber: number of days to add (e.g. 3; -2)  AddMonthNumber: same logic for month  AddYearNumber: same logic for year  AddHourNumber: same logic for hour  AddMinuteNumber: same logic for minute  AddSecondsNumber: same logic for seconds |
| Get Formatted Number | String – SourceValue  String – TargetFormat  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  TargetFormat: Numeric format e.g. #,0.00  ValueIfError: value returned if error in logic |
| Get Text Replace Values | String – SourceValue  String – ValuesToReplace  String – ValueIfError | String – ResultValue | SourceValue: Original value (e.g. value extracted by IQ Bot)  ValuesToReplace: use "," to separate old and new value and ";" to separate occurrences (e.g old1","new1";"old2","new2)  ValueIfError: value returned if error in logic |