**Formatting Dates**

Docmosis has a built-in function to format dates:

***dateFormat ( value [, outputFormat [, inputFormat] ])***

The parameters are:

***value*** *= the incoming Date to format*

***inputFormat (optional)*** *= a string that is use to decode the input date*

***outputFormat (optional)*** *= a string that represents the format to use for output*

The ***inputFormat***and ***outputFormat*** strings should be enclosed in single quotes.

**Example 1: inputFormat = not set**

If no *inputFormat* is specified Docmosis will try to decode the incoming *value* using a set of standard *inputFormats*.

**Date to format : <<dateOne>>**

|  |  |
| --- | --- |
| outputFormat | Output Result |
| dd/MM/yy | <<{dateFormat(dateOne, ‘dd/MM/yy’)}>> |

If the incoming *value* doesn’t match a known format (and Docmosis is in DEV mode) then an error will be injected in to the finished document.

**Date to format : <<dateTwo>>**

|  |  |
| --- | --- |
| outputFormat | Result |
| dd/MM/yy | <<{dateFormat(dateTwo, ‘dd/MM/yy’)}>> |

**Example 2: outputFormat = not set**

If *outputFormat* is not set Docmosis will use the default format ‘dd MMM yyyy’.

**Date to format : <<dateThree>>**

|  |  |
| --- | --- |
| outputFormat | Output Result |
| Not specified | <<{dateFormat(dateThree)}>> |

*Note: in the above example the Date to format is decoded using one of the known standard formats: ‘yyyy-MMdd'T'HH:mm:ss'Z'’. The list of possible standard inputFormats that will be used can be configured to suit your application.*

**Example 3: A selection of outputFormat’s**

**Date to format : <<dateOne>>**

|  |  |
| --- | --- |
| outputFormat | Output Result |
| dd/MM/yy | <<{dateFormat(dateOne, ‘dd/MM/yy’)}>> |
| MM-dd-yyyy | <<{dateFormat(dateOne, ‘MM-dd-yyyy’)}>> |
| MMMM dd, yyyy | <<{dateFormat(dateOne, ‘MMMM dd, yyyy’)}>> |
| hh:mmaa - MMMM dd | <<{dateFormat(dateOne, ‘hh:mmaa - MMMM dd’)}>> |
| EEEE, dd 'of' MMMM, yyyy | <<{dateFormat(dateOne, ‘EEEE, dd 'of' MMMM, yyyy’)}>> |

*Notice that in the last example above – the outputFormat string is enclosed in curly single quotes and the word 'of' is enclosed in straight quotes. This is a very subtle difference.*

*Straight quotes can be used to delimit characters in the formatting string that should pass straight through and not be interpreted as elements of the date. Characters not in straight quotes will be used to decode the values of day/month/year/time.***Example 4: Specifying inputFormat’s**

If an *inputFormat* is specified Docmosis will try to decode the incoming *value* using the *inputFormat* pattern.

**Date to format : <<dateTwo>>**

**inputFormat : ‘dd/MM/yyyy hh:mmaa’**

|  |  |
| --- | --- |
| outputFormat | Result |
| MM/dd/yy | <<{dateFormat(dateTwo, ‘MM/dd/yy’,‘dd/MM/yyyy hh:mmaa’)}>> |
| dd-MMM-yy | <<{dateFormat(dateTwo, ‘dd-MMM-yy’,‘dd/MM/yyyy hh:mmaa’)}>> |
| HH:mm | <<{dateFormat(dateTwo, ‘HH:mm’,‘dd/MM/yyyy hh:mmaa’)}>> |

**Date to format : <<dateFour>>**

**inputFormat : ‘MM-dd-yyyy 'at' HH:mm:ss’**

|  |  |
| --- | --- |
| outputFormat | Result |
| MM/dd/yy | <<{dateFormat(dateFour, ‘MM/dd/yy’,‘MM-dd-yyyy 'at' HH:mm:ss’)}>> |
| dd-MMM-yy | <<{dateFormat(dateFour, ‘dd-MMM-yy’,‘MM-dd-yyyy 'at' HH:mm:ss’)}>> |
| hh:mm aa | <<{dateFormat(dateFour, ‘hh:mm aa’,‘MM-dd-yyyy 'at' HH:mm:ss’)}>> |

**Date to format : <<dateFour>>**

**inputFormat : ‘MM-dd-yyyy’**

|  |  |
| --- | --- |
| outputFormat | Result |
| dd/MM/yy | <<{dateFormat(dateFour, ‘dd/MM/yy’,‘MM-dd-yyyy’)}>> |

*In DEV mode, if the incoming value doesn’t match the inputFormat then an error is injected.*