

Token Engine

Token Engine

MadeToPrint provides a powerful token engine that helps to define rules to generate dynamic strings, e.g. for file names, target directories and a variety of status information for the job slug (infobar).

The tokens (variables) are accessible via pulldown menus under *file name*; *infobar* and other functionality areas of MadeToPrint. A description of each variable can be found in the token help.



If you have further questions regarding tokens, e.g. generation of custom token sequences, please contact: support@axaio.com

How it works

Each token starts with « and ends with ». The token can contain a function (e.g. `pagelabel()`), a number (e.g. `123`), a string (e.g. `"Text"`) and operators (e.g. `+`). Functions might contain parameters, that are explained in token help in the file name section. White spaces (space, new line, tabulator) inside of tokens are ignored, which can help to format complex tokens.

Example: numbers and strings

- To understand the token engine, we start with the most basic options: numbers and string.
 - «5» => 5
 - «"MadeToPrint"» => MadeToPrint
- As mentioned, white spaces inside the token are ignored inside the token, but not inside strings or function names:
 - «5 » => 5

- « 5 » => 5
- « 5 » => 5
- « "MadeToPrint Token" » => MadeToPrint Token

Example: basic mathematical operators

- The token engine supports the basic mathematical operators. They can be combined and respect order of operations.

- « 5 + 4 » => 9
- « 5 - 4 » => 1
- « 5 * 4 » => 20
- « 5 / 4 » => 1.25
- « 5 + 4 * 3 » => 17
- « (5 + 4) * 3 » => 27

Example: & operator

- Adds two strings, one after another
 - « "Made" & "To" & "Print" » => MadeToPrint
- Can be combined with functions
 - « RIGHT("000" & PAGELABEL(), 3) »

Example: % (modulo) operator

- Returns remainder of a division
 - « 2 % 2 » => 0
 - « 5 % 3 » => 2
- Can be combined with functions, e.g. for using with page label to identify "left page" / "right page"
 - « IF(PAGELABEL()%2, "Right Page", "Left Page") »

Example: functions

- MadeToPrint provides various functions, which can provide information about the environment, the document, or even read from files on the hard disk. Function names are case-insensitive.
 - « ABS(-2) » => 2
 - « Abs(-2) » => 2

Example: multiple tokens

- In input fields where tokens can be used, you can also use multiple token
 - « "Made" & "To" & "Print" » => MadeToPrint

- `«"Made"»«"To"»«"Print" » => MadeToPrint`
- Token can also be mixed with plain text
 - `«"Made"»To«"Print" » => MadeToPrint`

Rule-based Queries (Ask Token)

With the help of so-called ASK-Tokens, it is possible to create queries, for example, to enable users to enter marks that will appear on the Infobar while executing a Output Job. The ASK token follows this syntax:

```
«ASK( "Prompt", "Scope", "Format", "Error" )»
```

The text entered underneath “*Prompt*” will be displayed as enter prompt within the dialog window while executing a Output Job.

The details under “*Scope*”, “*Format*”, and “*Error*” are optional and mean:

The Area "Scope"

- "Scope" defines the time of the query. Values possible for “Scope” are “*Job*”, “*Session*” and “*Page*”.
- "Job" this query appears while executing a Output Job. This is a default setting and does not need to be entered.
- "Session" this query appears before each InDesign session, so you only need to confirm after InDesign has been restarted.
- "Page" this query appears on each page of the document to be output.

The Area "Format"

- "Format" is used to define rules (conditions) for the input, for example,
- "\\w" limits the input to alphanumeric values
- "\\d" limits the input to numerical values
- "\\d{6}" limits the input to a 6-digit number

The Area "Error"

If the rule defined in “*Format*” does not match this, the following default error message will appear: *This entry is not al-*

lowed in the job settings. It is possible to adapt this preset error message by specifying an entry in “Error”.

Examples:

The plain token «ASK("Enter your name:")» displays a dialog every time a Output Job is being executed. The entry is not limited, so the user may enter any amount of characters in the query dialog.

The advanced token

```
«ASK("Date of publication [DDMMJJ]:";"Job";"\d{6}";"Error, enter a date!")»
```

causes a query dialog appearing each time a Output Job is being executed. The entry is limited to six numerical values. Whenever the entry is incorrect, the message „Error, enter a date!“ will appear.

Modifying the token in the area “Format“ will even check the validation of the entered numerals while being entered.

```
«ASK("Date of publication [DDMMJJ]:";"Job";"([12][0-9]/3[01]/0[1-9])(0[1-9]|1[012])([0-9][0-9]);"Error, enter a date!")»
```

The regular expression

"([12][0-9]/3[01]/0[1-9])(0[1-9]|1[012])([0-9][0-9])" describes the rules to enter the day (01 to max. 31), month and year.

([12][0-9]/3[01]/0[1-9]) describes the day (01 to max. 31) and allows only 1 or 2 in the first position, followed by 0 to 9. If the first number is 3, it can only be followed by 0 or 1. If the first number is 0, then only numerals from 1 to 9 may follow.

These rules also apply for month and year.

Page Numbers in File Names

The token «PAGENUMBER()» inserts the current page number, for example, into the file name. The token «PAGELABEL()» may also be used here, instead of total page numbers, page marks will be used, which have been defined in the numbering and section options of Adobe InDesign. In connection with the token «DOCUMENTNAME()», it is to define the following name conventions:

«PAGENUMBER()»_«DOCUMENTNAME()»

The result will look like this:

- 1_Document.pdf
- 2_Document.pdf
- 3_Document.pdf ...

It is possible to create a three digit page number (starting with zero) using this token combination

«RIGHT("000"&PAGENUMBER(), 3)»_«DOCUMENT-NAME()»

The token «RIGHT("000"&PAGENUMBER(), 3)» calculates from a string of text the defined amount of characters, in this case, beginning on the right. The string of text consists of triple zero "000" and the respective page number, e.g. "0001". Then it takes the three rightmost characters, e.g. "001". The total result would be

- 001_Document.pdf
- 002_Document.pdf
- 003_Document.pdf ...