

What should I know about input and output?



Written by Henrik Hofmeister
Updated over a week ago

What should I know about input and output?

Input

Often, your robot will require human-supplied input data in order to iterate through a list of URLs, perform search queries, or fill forms. To facilitate this, there are two tools available: in-editor input fields and run-specific input fields. The in-editor tools allow specification of input fields and testing-specific input values, while run-specific input fields allow bulk uploading of input data for specific executions.

Field name	Field type	Default value	Test value
search_query	Text		test
category	Text		shoes

Output

Output is the results of a robot's work. How a robot handles extracted information is defined in the robot editor with *output fields*, while actual, usable results are accessed from the execution page.

How do I use output fields?

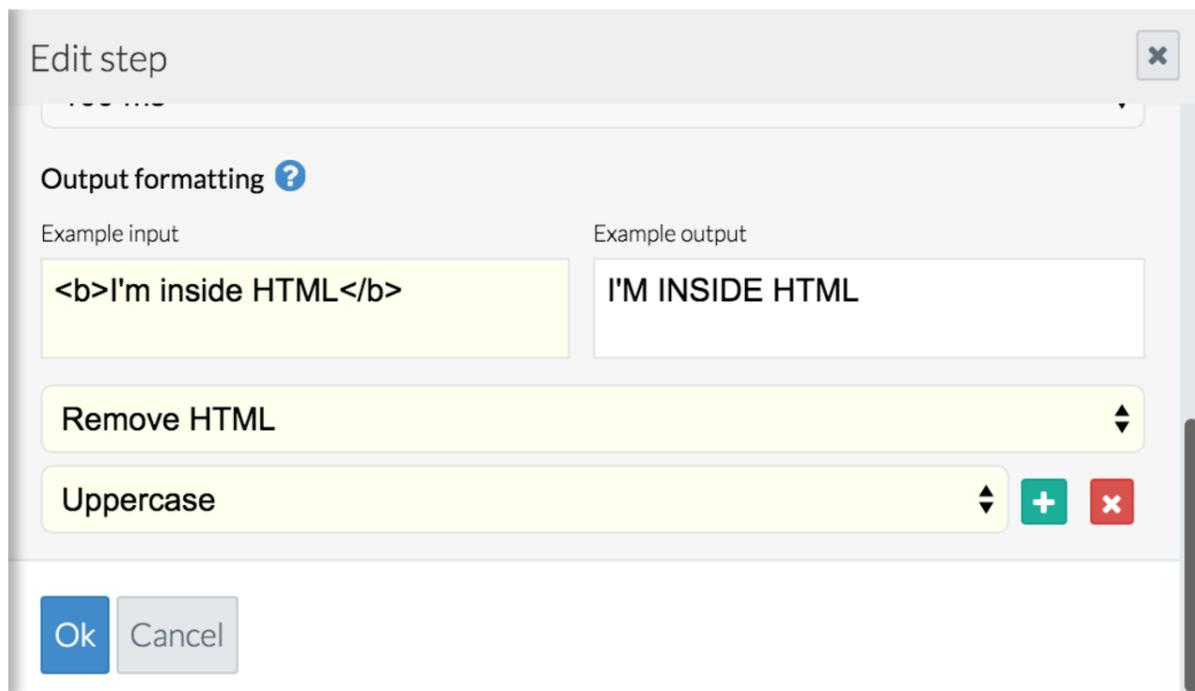
Output fields are the definition of what data your robot generates as results. These are defined in the *Outputs* tab. Once output fields have been defined, your robot can store results during execution.

Field name	Field type	Default value
text	Text	
author	Text	

+ Add field

How do I format my output?

Often, when extracting information from a Web site, the output contains garbled characters, HTML mark-up, or unwanted characters. To address this, every text extraction step type includes an *Output formatting* feature.



Each output formatter has a certain effect on the original output, which you can see live in the *Example output* text area. To test your output formatters with other input, simply edit the *Example input* text area.

Note that you can have as many output formatters as you'd like. If you have to match complex patterns, it is often easier to use multiple *Replace* and *Match* formatters to get the output you want.

Currently available output formatters include:

Number Ceiling: Only works for numbers - rounds a decimal number up to nearest integer.

Number Floor: Only works for numbers - rounds a decimal number down to nearest integer.

Number Round: Only works for numbers - rounds a decimal number to its nearest integer.

Format: Wraps the text in a format - replacing %1 with the existing text. E.g. *my format %1* becomes *my format rocks* for the output value *rocks*.

If Matches: If the output matches the given regular expression - then replace it with the given text. Optionally you can add %1 to the resulting text to insert the existing matched text.

If Not Matches: If the output does not match the given regular expression - then replace it with the given text. Optionally you can add %1 to the resulting text to insert the existing unmatched text.

Lowercase: Make all characters in the output lowercase.

Uppercase: Make all characters in the output uppercase.

Trim: Removes whitespace (line breaks and spaces) from the beginning and end of output.

Remove HTML: Removes all HTML tags and encoding. E.g. *Rock & Roll* becomes *Rock & Roll*.

Match: Remove everything but the first match in the output of the given regular expression. E.g. *USD 1.000* becomes *1.000* when matched with the regular expression *[1-9]\.*

Replace: Replaces one text with another. Uses regular expressions.

Prefix: Adds the given prefix to the text

Suffix: Adds the given suffix to the text

Date Format: Interprets and reformats a date according to the input and output formats. The format used is from Moment.js - which you can read more about [here](#).

Further reading

Regular Expressions examples:

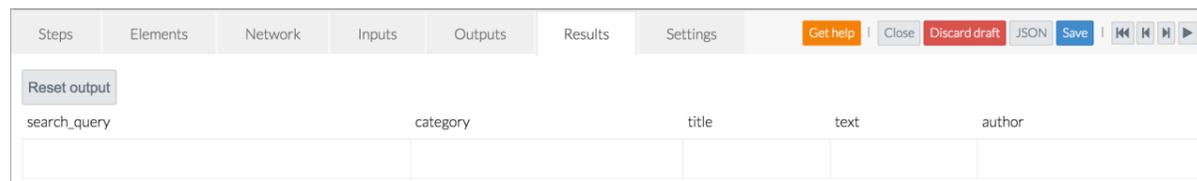
<http://www.regular-expressions.info/examples.html>

Online Testing tool for Regular Expressions:

<https://regex101.com/#javascript>

Where are my results?

As you test your robot in the editor, you can see the results under the Results tab. Note that the results in the editor are only intended for robot development and testing. When satisfied with your robot, create a run for it to upload real-world, "working" input data, execute your robot, and collect the resulting data.



search_query	category	title	text	author

Did this answer your question?



 We run on Intercom