

The Ultimate Power Automate expressions cheat sheet

Working with string / text

Extract a piece of a string

Build a string

Modify a string

Find a character or a string in a string

Get string length or check if it's (not) empty

Use a special character

Working with numbers

Use mathematic operations

Get the lowest or highest number

Get a random number

Change number format (currency, percentage...)

Working with dates (and times)

Calculate a date / time

Calculate a difference between dates / times

Get a specific date / time

Format a date

Display date / time in the correct time zone

Identify a day in a week / month / year

Working with arrays

Get shared values from two arrays

Get unique values from array(s)

Check if array contains a specific value

Create an array

Extract data from an array

Get the currently processed item in a loop

Create array with a number sequence

Working with JSON objects

Add new property to an object

Remove property from an object

Remove property from an object

Extract data from an array of JSON objects (e.g.

SharePoint items)

Conditions and logical operations

Use IF condition

Use logical operations

Compare values

Data conversion

Convert attachment content into binary

Convert data types

Other useful expressions

Create a unique identifier

Get flow information

The Ultimate Power Automate expressions cheat sheet

Working with string / text

Extract a piece of a string

Expression	Description & combination examples	Usage	Example	Example output
split()	Split string by a specific delimiter, returns an array as a result.	<code>split(string, delimiter)</code>	<code>split('Power Automate is great', ' ')</code>	<code>["Power", "Automate", "is", "great"]</code>
	Combination: <code>[index]</code> - access a specific value		<code>split('Power Automate is great', ' ')[0]</code> <code>split('Power-Automate-is-great', '-')[1]</code> <code>split('Power.Automate.is.great', '.')[2]</code>	<code>"Power"</code> <code>"Automate"</code> <code>"is"</code>
	Combination: <code>first()</code> - take the first value		<code>first(split('Power Automate is great', ' '))</code>	<code>"Power"</code>
	Combination: <code>last()</code> - take the last value		<code>last(split('Power Automate is great', ' '))</code>	<code>"great"</code>
	Combination: <code>join()</code> - convert array back to string with defined separator		<code>join(split('Power Automate is great', ' '), '_')</code>	<code>"Power_Automate_is_great"</code>
	Combination: <code>skip()</code> - skip X values		<code>skip(split('Power Automate is great', ' '), 1)</code>	<code>["Automate", "is", "great"]</code>
	Combination: <code>skip()</code> and <code>join()</code> - skip first X values and turn it back into a string		<code>join(skip(split(string, ' '), 1), ' ')</code> <code>*string = 'Power Automate is great'</code>	<code>"Automate is great"</code>
	Combination: <code>take()</code> - take X values		<code>take(split('Power Automate is great', ' '), 2)</code>	<code>["Power", "Automate"]</code>
	Combination: <code>take()</code> and <code>join()</code> - take first X values and turn it back into a string		<code>join(take(split(string, ' '), 2), ' ')</code> <code>*string = 'Power Automate is great'</code>	<code>"Power Automate"</code>
	Combination: <code>decodeURIComponent()</code> - split string by a special character, e.g. new line		<code>split(string, decodeURIComponent('%0A'))</code> <code>*string = 'Power Automate is great'</code>	<code>["Power", "Automate", "is", "great"]</code>
substring()	Extract a piece of string using start position and number of characters. First position is 0.	<code>substring(string, startPosition, numberOfCharacters)</code>	<code>substring('Power Automate is great', 0, 5)</code> <code>substring('Power Automate is great', 10, 5)</code>	<code>"Power"</code> <code>"mate "</code>
	Combination: <code>indexOf()</code> - find the position of a specific character/string dynamically		<code>substring(string, indexOf(string, 'Auto'), 8)</code> <code>*string = 'Power Automate is great'</code>	<code>"Automate"</code>
	Combination: <code>lastIndexOf()</code> - find the last position of a specific character/string dynamically		<code>substring(string, lastIndexOf(string, ' '), 6)</code> <code>*string = 'Power Automate is great'</code>	<code>" great"</code>

The Ultimate Power Automate expressions cheat sheet

Expression	Description & combination examples	Usage	Example	Example output
slice()	Extract a piece of string using start position and end position. Unlike with substring() you don't need to know the number of characters to extract. First position is 0.	slice(string, startPosition, endPosition)	slice('Power Automate is great', 0, 5) slice('Power Automate is great', 10, 15)	"Power" "mate "
	Combination: indexOf() - find the first position of a specific character/string dynamically		slice(string, indexOf(string, 'P'),indexOf(string, ' ')) *string = 'Power Automate is great'	"Power"
	Combination: lastIndexOf() - find last position of a specific character/string dynamically		slice(string, 0, lastIndexOf(string, '.')) *string = 'Power Automate is great.pdf'	"Power Automate is great"
	Combination: indexOf() and lastIndexOf() - extract string between first and last position of character/string dynamically		slice(string, indexOf(string, '_'), lastIndexOf(string, '.')) *string = '001_Power Automate is great.pdf'	"_Power Automate is great"
	Combination: combinations above and add() - extract the string after the character/string you were looking for		slice(string, add(indexOf(string, ' '),1), lastIndexOf(string, '.')) *string = '001_Power Automate is great.pdf'	"Power Automate is great"