

QuicklySign Syntax Definitions

This document outlines Tag Syntax and its various uses.

What is 'Tag Syntax'?

'Tag Syntax' is positional arguments separated by a pipe (| ← this "vertical bar" is also known as a 'pipe').

Why do I need it?

'Tag Syntax' simplifies the upload and setup of Microsoft Word generated documents on QuicklySign, by allowing you to use syntax to 'tag' where fields need to be placed on the document.

Some examples of where you may need it:

- You generate a letter that needs a signature on it; you place the tags in your letter template and when you upload the letter to QuicklySign, you click a 'wand' to automatically assign the fields.
- You have a document generated in another software application; if that application allows you to modify their templates, you can place QuicklySign syntax in those templates to indicate where the users must sign. These documents could be sent via API or directly uploaded on the frontend.

How does it work?

On upload, QuicklySign automatically detects the syntax and generates the corresponding fields when the trigger 'wand' that appears is clicked.

Depending on the syntax argument, the corresponding field that's generated simply overlays the syntax text on the document, in line with the configuration dictated by the syntax argument.

`{{ tx|1|1 }}`

This creates a text field that is required and assigned to signer 1

Transforms to



This creates a text field that is required and assigned to signer 1

The overall size of the resultant field is influenced by two things: the size of the font used to construct the argument and the distance of the curly brackets from the text.

- **Font size:** The bigger the argument font size, the bigger the resultant field

{{tx|1|1}} This creates a bigger field that is required and assigned to signer 1

Transforms to

* This creates a bigger field that is required and assigned to signer 1

- **Curly bracket distance:** The further away the curly brackets from the text, the wider the resultant field

{{ tx|1|1 }} This creates a wider field that is required and assigned to signer 1

Transforms to

* This creates a wider field that is required and assigned to signer 1

ARGUMENT PARAMETERS

Using tag syntax requires you to set up your positional argument such that when the document is uploaded the required field is rendered. The parameters for that argument comprise eight positions:

{{ type¹ | signer² | is the field required³ | field name (custom id)⁴ | help or info⁵ | field group⁶ | field specific options⁷ | is the field selection mutually exclusive⁸ }}

FIELD TYPES

Each field type has a corresponding syntax for use in the argument(s).

Field Type	Syntax
Text	tx
Signature	sig
Initials	in
Date	dt
Date (auto complete on signature)	dta
Checkbox	cb
File	fl
Dropdown	cho
Link	lk
Approve	apr
Decline	dec
Image	im

BASIC USAGE

Basic syntax focuses on the first three positions; the field 'type', the 'signer' the field is assigned to and whether the field is 'required'.

{{ type¹ | signer² | is the field required³ }}

Position 1 = Sets the field type

Position 2 = Indicates the signer the field has been assigned to by either numerical value, starting at 1, or a blank

- 1, 2, 3, etc. = either the first, second or third signer
- Blank = a field not assigned to any signer, for completion before sending the document out

Position 3 = Indicates whether a field is required; by either a 1, 0, or a blank

- 1 = required
- Blank = not required, except for signature, initial, and approve fields
- 0 = not required for signature and initial fields

ARGUMENTS

Basic syntax arguments can be constructed as follows:

For a **required text field assigned to signer 1**, your syntax argument would look as follows:

`{{tx|1|1}}`

- 'tx' represents a 'text' field, the first '1' represents signer-1, the second '1' indicates that the field is 'required'.

For a **non-required signature field assigned to signer 2**, your syntax argument would look as follows:

`{{sig|2|0}}`

- 'sig' represents a 'signature' field, the '2' represents signer-2, the '0' indicates that the field is not 'required'.

For a **required initial field assigned to Me (now) i.e., a field to be completed before sending the document out**, your syntax argument would look as follows:

`{{in||1}}`

- 'in' represents an 'initial' field, the absence of a signer indicates that the field is not assigned to a signer, the '1' indicates that the field is 'required'.

BASIC USAGE SUMMARY

Field Type	Syntax	Required	Not required
Text	tx	<code>{{tx 1 1}}</code>	<code>{{tx 1}}</code>
Signature	sig	<code>{{sig 1 1}}</code>	<code>{{sig 1 0}}</code>
Initials	in	<code>{{in 1 1}}</code>	<code>{{in 1 0}}</code>
Date	dt	<code>{{dt 1 1}}</code>	<code>{{dt 1}}</code>
Date (auto complete on signature)	dta	<code>{{dta 1}}</code>	
Checkbox	cb	<code>{{cb 1 1}}</code>	<code>{{cb 1}}</code>

File	fl	*Advanced usage	
Dropdown	cho	<code>{{cho.options(yes,no) 1 1}}</code>	<code>{{cho.options(yes,no) 1}}</code>
Link	lk	*Advanced usage	
Approve	apr	<code>{{apr 1 1}}</code>	<code>{{apr 1 0}}</code>
Decline	dec	*cannot be a required field	<code>{{dec 1}}</code>
Image	im	*Advanced usage	

DETAILED USAGE

Detailed syntax introduces enhancements to the first three positions: pre-set values, variables, and field dimensions.

PRE-SET VALUES

You can pre-set values on field types using the **‘.val’** function as follows:

For a **required textbox assigned to signer 1, that's been prepopulated with QuicklySign**, your syntax argument would look as follows:

`{{tx.val("QuicklySign")|1|1}}`

*

For a **required checkbox assigned to signer 1, that's been pre-checked**, your syntax argument would look as follows:

`{{cb.val(checked)|1|1}}`

*

For a **required dropdown assigned to signer 1, with options yes & no, but yes preselected**, your syntax argument would look as follows:

`{{cho.options(yes,no).val("yes")|1|1}}`

*

PRE-SET VALUES SUMMARY

Field Type	Syntax	Basic	Pre-set Value
Text	tx	{{tx 1 1}}	{{tx.val("QuicklySign") 1 1}}
Signature	sig		
Initials	in		
Date	dt		
Date (auto complete on signature)	dta		
Checkbox	cb	{{cb 1 1}}	{{cb.val(checked) 1 1}}
File	fl	*Advanced usage	
Dropdown	cho	{{cho.options(yes,no) 1 1}}	{{cho.options(yes,no).val("yes") 1 1}}
Link	lk	*Advanced usage	
Approve	apr		
Decline	dec		
Image	im	*Advanced usage	

VARIABLES

Sometimes the length of the syntax argument can result in a field that's longer than the space provided in the document; so, to resolve that we use variables.

Using variables allows you to reference the argument without putting it directly where it needs to go, by assigning a 'variable' to the argument, the variable referencing the argument is placed in the sentence instead of the argument.

Example:

The argument:

```
{{tx.val("QuicklySign")|1|1}}
```

QuicklySign *

The sentence:

By signing this contract <we need a long argument with limited space> and its users accept and agree to comply with the terms and conditions.

The variable:

\$a


- 'a' is the variable, but the variable MUST be prefixed with '\$'
- The variable can be a single text string (no space separation), not just limited to a single character

Defining the variable using the argument:

{{\$a=tx.val("QuicklySign")|1|1}}

The sentence using the variable to reference the argument:

By signing this contract {{\$a}} and its users accept and agree to comply with the Terms and Policies.

By signing this contract * and its users accept and agree to comply with the Terms and Policies.

FIELD DIMENSIONS

By default, the width and height of the field is determined by the font size and curly bracket placement, however, you can overwrite these by prescribing a field height and width.

Example:

{{tx|1|1}}



With dimensions:

{{tx.w(500).h(50)|1|1}}



- Setting field dimensions should be done in conjunction with the use of variables, to avoid long syntax arguments
 - The height and width would then be set when defining the variable

STANDARD QUICKLYSIGN DIMENSIONS

Field Type	Syntax	Width	Height	Basic
Text	tx	150	22	{{tx.w(150).h(22) 1 1}}
Signature	sig	150	40	{{sig.w(150).h(40) 1 1}}
Initials	in	50	40	{{in.w(50).h(40) 1 1}}
Date	dt	150	22	{{dt.w(150).h(22) 1 1}}
Date (auto complete on signature)	dta	150	22	{{dta.w(150).h(22) 1}}
Checkbox	cb	20	20	{{cb.w(20).h(20) 1 1}}
File	fl	150	50	*{{fl.w(150).h(50) 1 1}}
Dropdown	cho	150	22	{{cho.options(yes,no).w(150).h(22) 1 1}}
Link	lk	150	22	*{{lk.w(150).h(22)}}
Approve	apr	150	50	{{apr.w(150).h(50) 1 1}}
Decline	dec	150	50	{{dec.w(150).h(50) 1}}
Image	im	400	300	*{{im.w(400).h(300) 1 1}}

ADVANCED USAGE

Advanced syntax introduces the remaining five positions to the first three: custom field names, help text, field group names, field specific options, and mutual exclusivity.

{{ type⁴ | signer² | is the field required³ | field name (custom id)⁴ | help or info⁵ | field group⁶ | field specific options⁷ | is the field selection mutually exclusive⁸ }}

Position 4 = Sets a custom field name (custom id)

Position 5 = Dictates help text

Position 6 = Assigns a name to a group of fields

Position 7 = Sets any field specific options

Position 8 = Indicates whether a field is mutually exclusive by either a 1 or a blank

- 1 = mutually exclusive
- Blank = not mutually exclusive

FIELD NAME (CUSTOM ID)

You can set a custom field name to be able to reference a field at a later stage.

You can either **just populate** the **type¹** and **field name (custom id)⁴** positions:

{{tx||Full_Name}}

Assigned to

Required

Type

Help or Info

Detail of what to fill out here

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Use field replication [?](#)

Field Group

OR

You can **populate everything** from **type¹** to **field name (custom id)⁴**.

{{tx|1|Full_Name}}

Assigned to

Required

Type

Help or Info

Detail of what to fill out here

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Use field replication [?](#)

Field Group

- The chosen name should be unique to each field to avoid [‘field replication’](#).

HELP TEXT

You can insert help text to help the signer understand what they are required to do in each field.

You can either **just populate** the **type**¹ and **help or info**⁵ positions:

{{tx||||Insert full name}}

Assigned to

Required

Type

Help or Info

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

OR

You can **populate everything** from **type**¹ to **help or info**⁵:

{{tx|1|1|Full_Name|Insert full name}}

Assigned to

Required

Type

Help or Info

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Use field replication 

Field Group

While help text is optional, two fields require it to function effectively: namely, 'file' and 'link'.

For the **file and image upload**¹, it's important to **communicate the file you want them to upload**⁵. You can then also indicate who the field is assigned to², whether the upload is required³ as well as insert a custom field name⁴.

{{fl|||Copy of ID}} OR **{{fl|1|1|ID_Upload|Copy of ID}}**

Assigned to

Required

Type

File to upload

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

Assigned to

Required

Type

File to upload

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

For the **link**, the **help text works together with a pre-set value**. The pre-set value inserts the destination address, while the help text sets the display text to prompt them to click on the link.

{{lk.val("https://www.quicklysign.com/")|||For more information please click here}}

Type

Link

Link text to display

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

Unlike the other fields, you cannot assign a link to a specific signer, nor can you make the link required, the most you can do is insert a custom field name.

`{{lk.val("https://www.quicklysign.com/")|||Link_Address|For more information please click here}}`

Type

Link

Link text to display

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

- To ensure the destination address is safe for the signer to click on, the full 'https:' address must be inserted.

FIELD GROUPS

You can give fields in a particular section of a document a group identifier by assigning a field group name.

You can either **just populate** the **type**¹ and **field group**⁶ positions:

`{{tx|||Particulars}}`

Assigned to

Required

Type

Help or Info

Detail of what to fill out here

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

Particulars

OR

You can **populate everything from type¹ to field group⁶**:

{{tx|1|1|Full_Name|Insert full name|Particulars}}

Assigned to

Required

Type

Help or Info


Insert full name

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Full_Name

Use field replication 

Field Group

Particulars

FIELD SPECIFIC OPTIONS

In addition to being part of a section in a document, checkboxes can be part of a group so to accommodate a checkbox group (which is only specific to checkboxes), we use the field specific option.

You can either **just populate** the **type¹** and **field specific options⁷** positions:

{{cb|||||Marital_Status}}

{{cb|||||Marital_Status}}

Assigned to

Required

Type

Field Group

Checkbox group

Only 1 value can be selected in the checkbox group.

OR

You can **populate everything from type¹ to field specific options⁷**:

{{cb|1|1|Married_Checkbox|Indicate your marital status|Particulars|Marital_Status}}

{{cb|1|1|Single_Checkbox|Indicate your marital status|Particulars|Marital_Status}}

Assigned to

Required

Type

Help or Info

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Use field replication for checkboxes

Field Group

Checkbox group

Only 1 value can be selected in the checkbox group.

MUTUAL EXCLUSIVITY

For **mutually exclusive checkboxes**, the field specific option i.e., **the checkbox group name, works together with the mutual exclusivity indicator**. The checkbox group name identifies the group of options, while the mutual exclusivity indicator limits the selection to just one option in the group.

You can either **just populate the type¹, field specific options⁷, and mutual exclusivity⁸** positions:

{{cb|||||Marital_Status|1}}

{{cb|||||Marital_Status|1}}

Assigned to

Required

Type

Field Group

Checkbox group

Only 1 value can be selected in the checkbox group.

OR

You can **populate everything from type¹ to mutual exclusivity⁸**:

{{cb|1|1|Married_Checkbox|Indicate your marital status|Particulars|Marital_Status|1}}

{{cb|1|1|Single_Checkbox|Indicate your marital status|Particulars|Marital_Status|1}}

Assigned to

Required

Type

Help or Info

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Use field replication for checkboxes [?](#)

Field Group

Checkbox group

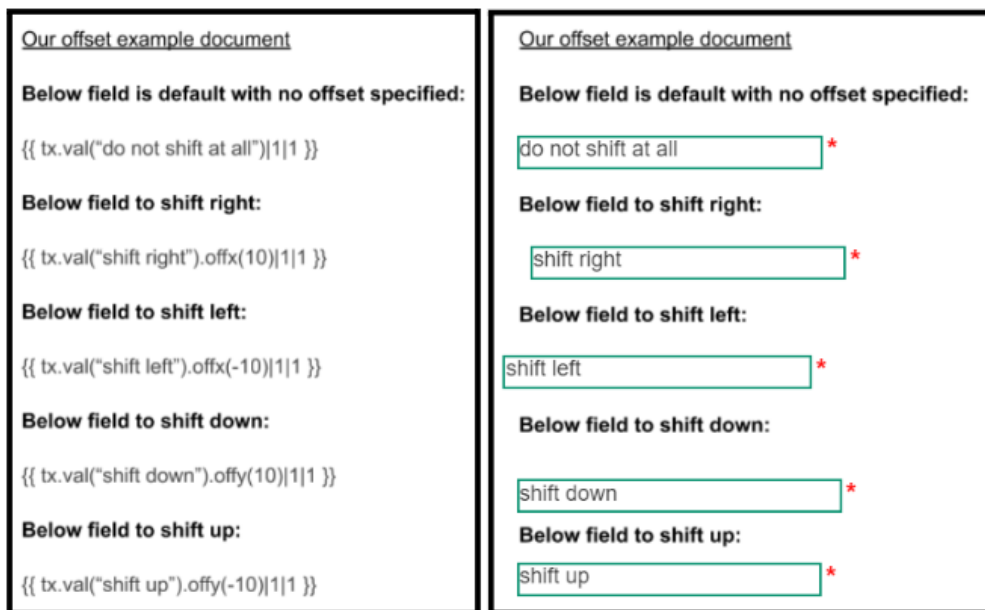
Only 1 value can be selected in the checkbox group.

ALIGNMENT ADJUSTMENT

In case your syntax field requires minor layout adjustments we have introduced the offset functionality. This will allow you to shift your detected fields location in any direction by an amount of pixels that you specify. Below is a representation of how to use this:

Your specified **offx** amount will allow you to adjust your field to the left or to the right. If you specify a positive amount `.offx(10)` the field will then shift to the right by 10 pixels. If you specify a negative amount `.offx(-10)` the field will then shift to the left by 10 pixels.

Your specified **offy** amount will allow you to adjust your field up or down. If you specify a positive amount `.offy(10)` the field will then shift down by 10 pixels. If you specify a negative amount `.offy(-10)` the field will then shift up by 10 pixels. To see this in action, see below before and after result:



VERBOSE SYNTAX

While the positional syntax works well for most cases, it can get confusing when you have more than three positions filled, so to make it easier, we have introduced verbose syntax.

Verbose syntax ignores the argument order required by positional syntax and uses full stops instead of pipes, as separators. This can make the tag syntax a lot easier to use as you don't have to try and cater for the tag field size when working between text as well as see an explicit naming convention for the information you're adding to your field.

Lets see an example:

Our scenario is that we want to create a text field, it needs to be assigned a signee, given a custom ID of “employee_first_name”, make it required, give it a description of “Please enter your first name” as well as pre-populate the field with the text “Joe” and lastly, give it a field grouping. This can get confusing when using the pipe driven format as we now require multiple fields while keeping our formatting clean.

A positional syntax argument for text field would look as follows:

Format:

```
{{ field_type|field_signee|mandatory_flag|your_custom_id|field_description|field_group }}
```

Using pipe format in practice for our scenario:

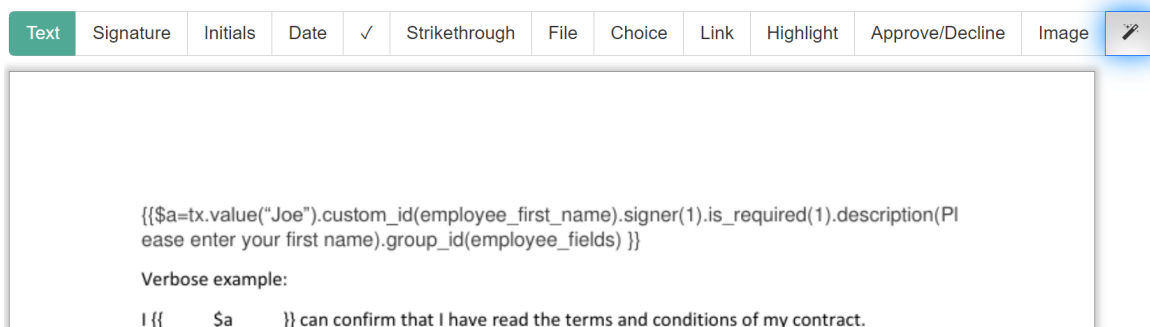
```
{{ tx|1|employee_first_name|"Please enter your first name"|employee_fields }} - as you can see when creating multiple fields with multiple pipes it can get confusing.
```

Verbose syntax now allows us to work with the fields more freely within the document and we can explicitly see which fields we are populating, eliminating the confusion of pipe positioning.

```
{{ $a=tx.value("Joe").custom_id("employee_first_name").signer(1).is_required(1).description("Please enter your first name").group_id(employee_fields) }}
```

 - you'll notice it is now a lot easier to use, we can now explicitly see what we are changing within the field as well as use it within a sentence without ruining formatting. [See below front-end sample for reference:](#)

Set up signature spaces on this document



Text Signature Initials Date ✓ Strikethrough File Choice Link Highlight Approve/Decline Image

```
{{ $a=tx.value("Joe").custom_id(employee_first_name).signer(1).is_required(1).description(Please enter your first name).group_id(employee_fields) }}
```

Verbose example:

I {{ \$a }} can confirm that I have read the terms and conditions of my contract.

NB: Verbose syntax should be used in conjunction with the use of variables, to avoid long syntax arguments

Result (you'll notice that our front end will automatically hide your verbose variable):

Set up signature spaces on this document

Text	Signature	Initials	Date	✓	Strikethrough	File	Choice	Link	Highlight	Approve/Decline	Image
------	-----------	----------	------	---	---------------	------	--------	------	-----------	-----------------	-------

Verbose example:

I can confirm that I have read the terms and conditions of my contract.

Text Hide

Please enter your first name

Assigned to

Required

Type

Help or Info

Please enter your first name

[Hide Advanced Setup](#)

Field Details

Field name (custom id)

Field Group

Please sign this document QuicklySign

Locate the fields assigned to you and click "Submit" when complete.

Verbose example:

I can confirm that I have read the terms and conditions of my contract.

Please enter your first name

Field Type ¹ (abbreviation in brackets)	Signer ² [.signer()]	Is the field required ³ [.is_required()]	Field Name ⁴ (custom id) [.custom_id()]	Help or info ⁵ [.description()]	Field Group ⁶ [.group_id()]	Field specific options ⁷ (see examples in line)	Mutual exclusivity ⁸ [.is_mutually_exclusive()]
Text (tx)	Allows you to indicate the signer a field must be assigned to, by numerical value, starting at 1. Can be left blank , however, if left blank, it's assumed to be a field none of the signers should be able to edit.	Allows you to indicate whether the completion of a field is required or not, by either 1 or 0. Can be left blank , however, if left blank, it's assumed the field is not required. The signature, initial, and approve field are by default required, so to indicate 'not required', place a zero (0).	Allows you to set a custom field name to be able to reference a field at a later stage. Can be left blank , however, if included, the chosen name should be unique to each field to avoid 'field replication'.	Allows you to insert help text to help the signer understand what they are required to do in each field.	Allows you to give fields in a particular section of a document a group identifier. Can be left blank.		
Signature (sig)							
Initials (in)							
Date (dt, dta)							
Checkbox (cb)							
File (fl)				Allows you to indicate the file you want them to upload.		The field specific option i.e., the checkbox group name, works together with the mutual exclusivity indicator. The checkbox group name identifies the group of options, while the mutual exclusivity indicator limits the selection to just one option in the group.	
Dropdown (cho)				Help text			
Link (lk)				Works together with a pre-set value. The pre-set value inserts the destination address, while the help text sets the display text to prompt them to click on the link.			
Approve (apr)				Help text		Allows you to set the button display text e.g. accept/reject [.button_text()]	
Decline (dec)							

Image (im)				Allows you to indicate the image you would like them to upload.			
------------	--	--	--	---	--	--	--