Jinja Tokens

`{% ... %}`	Control statements
`{{ ... }}`	Replacement expressions
`{# ... #}`	Comments

Whitespace control

+	Disable strip behavior to include extra whitespace
–	Remove whitespace around blocks

These modifiers can be added to the beginning or the end of a Jinja control token. `{%-` or `-%}`. In the case of `+`, the position controls where the trim is disabled. They can be applied to any control token `{%- %}`, `{{- }}`, or `{#- #}`.

Control blocks

```
{% if <statement> %}
...
{% elif <statement> %}
...
{% else %}
...
{% endif %}

{% for <item> in <list> [recur-
sive] %}
...
{% endfor %}

{% block <name> [scoped]
[required] %}
...
{% endblock %}

{% raw %}
...
{% endraw %}

{% macro <name>(<args...>) %}
...
{% endmacro %}

{% set <name> %}
...
{% endset %}
```

Filters

Built-in filters

abs	forcee scape	map
attr	format	max
batch	groupby	min
capitalize	indent	pprint
center	int	random
default	items	reject
dictsort	join	rejectattr
escape	last	replace
filesi- zeformat	length	reverse
first	list	round
float	lower	safe

(additional filters overlapping: select, selectattr, slice, sort, string, striptags, sum, title, tojson, trim, truncate, unique, upper, urlencode, urlize, wordcount, wordwrap, xmlattr)

Assignments and macros

Variable assignment

Simple variables can be assigned by using the `set` block: `{% set variab-le='value' %}`

Block assignments

Templated blocks can be assigned to variable names by using a `set` block.
```
{% set reply %}
    Thanks for your message:
    {{ message }}.
{% endset %}
```

Loop special variables

`loop.index`

The current iteration of the loop. (1 indexed)

`loop.index0`

The current iteration of the loop. (0 indexed)

`loop.revindex`

The number of iterations from the end of the loop (1 indexed)

Loop special variables (cont)

`loop.revindex0`

The number of iterations from the end of the loop (0 indexed)

`loop.first`

True if first iteration.

`loop.last`

True if last iteration.

`loop.length`

The number of items in the sequence.

`loop.cycle`

A helper function to cycle between a list of sequences.

`loop.depth`

Indicates how deep in a recursive loop the rendering currently is. Starts at level 1

`loop.depth0`

Indicates how deep in a recursive loop the rendering currently is. Starts at level 0

`loop.previtem`

The item from the previous iteration of the loop. Undefined during the first iteration.

`loop.nextitem`

The item from the following iteration of the loop. Undefined during the last iteration.

`loop.changed(*val)`

True if previously called with a different value (or not called at all).

Variables can be modified with filters. To apply a filter use a pipe, '|'. For example, `{{ name|upper }}`.

Filters can be chained together. For example, `{{ name|striptags|title }}` will strip all HTML tags from `name` then convert it to Title case.

Some filters allow arguments. See the Jinja documentation for details.