

Délimiteurs	Filtre Chaines (cont)	Filtre Liste
<code>{{ Variable }}</code>	<code>string(object)</code> <code>striptags(value)</code>	<code>dictsort(value, case_sensitive = 'key')</code>
<code>{{ Var.attrib }}</code> <code>{{ Var[attrib] }}</code>	<code>title(s)</code> <code>trim(value)</code>	<code>first(seq)</code> <code>last(seq)</code>
<code>{{ Variable filtre ... }}</code>	<code>urlencode(value)</code>	<code>groupby(value, attr)</code> <code>map()</code>
<code>{% Controle %}</code>	<code>urlize(value, trim_url_limit, nofollow, target)</code>	<code>join(value, d=' u', attribute =None)</code>
<code>{# Commentaires #}</code>	<code>wordcount(s)</code> <code>xmlattr(d, auto_spaces)</code>	<code>length(object)</code> <code>l1st()</code>
Controles	Filtre Nombres	Tests
<code>for else endfor</code> <code>loop.xxx</code>	<code>abs(number)</code> <code>random(seq)</code>	<code>callable(object)</code> <code>divisibleby(x, y)</code>
<code>if elif else endif</code>	<code>int(value, default=0)</code> <code>float(value, default=0.0)</code>	<code>escaped(value)</code> <code>mapping(value)</code>
<code>macro endmacro</code> <code>call</code>	<code>sum(iterable, attribute=None, start=0)</code>	<code>iterable(value)</code> <code>sequence(value)</code>
<code>filter endfilter</code>		<code>string(value)</code> <code>number(value)</code>
<code>set</code> <code>set block endset</code>		<code>lower(value)</code> <code>upper(value)</code>
<code>include</code> <code>ignore missing</code>		<code>defined(value)</code> <code>undefined(value)</code>
<code>import</code> <code>from import as</code>		<code>none(value)</code> <code>even(value)</code> <code>odd(value)</code>
Opérateurs		<code>equalto(value, other)</code> <code>sameas(value, other)</code>
<code>+</code> <code>-</code> <code>/</code> <code>//</code>		
<code>%</code> <code>*</code> <code>**</code>		
<code>==</code> <code>!=</code> <code>></code> <code>>=</code>		
<code><</code> <code><=</code>		
<code>and</code> <code>or</code> <code>not</code> <code>(expr)</code>		
<code>in</code> <code>is</code> <code> </code> <code>~</code>		
<code>./[]</code> <code>foo.bar</code> <code>foo[' bar']</code>		
Espaces		
<code>{%- %}</code> <code>supprime espaces avant ou après</code>		
<code>{%+ }</code>		
Filtre Chaines	Filtre Divers	
<code>capitalize(s)</code> <code>center(value, width)</code>	<code>attr(obj, name)</code> <code>reject attr()</code>	
<code>escape(s)</code> <code>forceescape(value)</code>	<code>default(value, default, boolean = False) d</code>	
<code>format(value, *args, **kwargs)</code>	<code>filesizeformat(value, binary =False)</code>	
<code>indent(s, width=4, indentfirst = False)</code>	<code>safe(value)</code>	
<code>lower(s)</code> <code>upper(s)</code>	<code>select()</code> <code>select attr()</code>	
<code>pprint(value, verbose=False)</code>		
<code>replace(s, old, new, count=None)</code>		
	Utilisé dans <code>{{ Variable filtre ... }}</code> ou dans les commandes	

