

Calling a keyword

```
{{ keyword <parameters> }}
```

Essential Keywords

var **“**name

Emit the contents of the variable name. If the corresponding variable isn't found, pop a value from the stack.

set **“**name **«**content

Set the contents of the variable name to content. If the variable does not exist, it will be created.

unset **“**name

Discard the contents of the variable name.

ifdef **“**name **{ { }**

Produce the contents of block if variable name is defined.

ifndef **“**name **{ { }**

Produce the contents of block if variable name is NOT defined.

include **“**file-path

Produce and include the contents of the HTML or Markdown file located at file-path (relative to project folder).

truncate **«**content **32**trim-at

Produce content, trimming the result if it passes the threshold at trim-at.

list_doc **≡** doc OR **≡** docs_in

var : **“**doc-path **“**template-path **{ { }**

Produce and list the Markdown document located at doc_path, using the template at template_path (both paths relative to the project folder). Import the variables set in the front matter into block, along with the document contents docdata and the path of the HTML file at path.

docs_in : **“**dir-path **“**template-path
“sort-by **“**order **{ { }**

Produce and list the Markdown documents located at dir-path, using the template at template-path (both paths relative to the project folder). Sort the order of entries via the variable sort-by, with order=reverse OR order=normal, corresponding to the alphabetical sorting method. Import the variables set in the front matter into block, along with the document contents docdata and the path of the HTML file at path.



Kinds of parameters

“ **Literal** - “Hello, world!”

« **Lit. (interpreted)** - Produces the result of any keywords within it.

32 **Integer** - An unsigned, 32-bit integer.

{ { } **Block content** - Begins with a newline, ends with the two closing brackets. Whatever is inside it is produced.

≡ **Sub-command** - A parameter which modifies the expected parameters relating to a keyword.

Project Hierarchy



The project folder

The command line

new - Create a new project in the current directory.

new_bare [name] - Create a new project in the name directory. If the directory does not exist, it will be created, too.

build - Build the current project to the out/ directory.

serve [portno] - Serve the site locally on port portno at localhost.

licenses - Print license information for Ploogly and the software it uses.

Blog Example

Project creation & setup

```
$ ploogly new_bare bloog
$ mkdir bloog/templates
$ mkdir bloog/posts
$ cd bloog
```

Edit file "project.ssg"

```
name: Burgerpants' Books
url: https://ballsfallart/
```

Create file "site/index.html"

```
<!DOCTYPE html>
<html>
  <head>
    <title>My bloog | {{ var name }}</title>
  </head>

  <body>
    <h1>Welcome to my bloog, {{ var name }}</h1>

    {{ list_doc docs_in posts templates/post.html title normal
      <h3><a href="{{ var path }}">{{ var title }}</h3>
      <p>{{ truncate "{{ var docdata }}" 140 }}</p>
    }}
  </body>
</html>
```

Create file "templates/post.html"

```
<!DOCTYPE html>
<html>
  <head>
    {{ ifdef title
      <title>{{ var title }}</title>
    }}
  </head>

  <body>
    {{ ifdef title
      <h1>{{ var title }}</h1>
    }}

    {{ var docdata }}
  </body>
</html>
```

File "posts/post1.md"

```
---
title: Twilight Review
---

Hello guys, today I'm gonna
tell you my opinions on why I
think Twilight is the most
dialectical book ever written,
surpassing even the original
developments made by Hegel via
its intricate use of prove to
convey a message which shakes
the current positivist (and
Popperian pseudo-positivist)
status quo to the core

... blah blah blah
```

Serving on 127.0.0.1:5000

```
$ ploogly build && ploogly serve 5000
```

This document is released under the Creative Commons Attribution-ShareAlike 4.0 International License. 

Ploogly © The Ploogly Contributors, 2025. All Rights Reserved.

Image credits

"A colored Emoji from Android project" by Google, under the Apache License 2.0, available at <https://www.apache.org/licenses/LICENSE-2.0>

"Curly Brackets" by Ederporto, under the Public Domain (CC0 1.0)

"Кавычки иконка" by Рытикова Людмила, under the CC-BY-SA 4.0

"Folder icon aem" by Aemiliusx, under the CC-BY-SA 4.0

"Guillemet français ouvrant" by Antimuonium, under the Public Domain (simple geometry)

The canary in the Ploogly logo is modified from an illustration in "the 1684 edition of Uccelliera" by Giovanni Olina, under the Public Domain

All images sourced from Wikimedia Commons.