

# **Creating a Website With Jekyll**

# Contents

- What is Jekyll?.....3**
- Create a Configuration File.....3**
  - Configuration File Settings.....4
  - What are Variables?..... 4
  - Create Default Variables..... 5
- Write a Post.....5**
  - What is YAML Front Matter?..... 6
  - Override Variables.....6

# What is Jekyll?

---

Jekyll is a program that takes static content and layout files and merges them to create a website. Jekyll provides the ability to reuse website layouts, share data between files, and customize URLs without access to a formal website backend.

The layout of a Jekyll website is stored separately from the content of the website. This means that the layout can be updated from only one location, rather than needing to update many pages individually. Jekyll also allows for multiple layouts, meaning a website can have several designs to fit various types of content. If a layout needs to access data from content pages, such as a blog post title, that data can be stored in a variable and accessed from the layout.

Variables store customized data that can be reused throughout the website. By storing data in a variable, the data can be updated with one change to the variable, instead of having to update many individual pages on the website. Other site information is also automatically stored in non-customizable variables, such as the current date.

URLs are easily customizable with the help of Jekyll. Jekyll can link to posts via their filename. This allows for the URL for each post to be changed easily when desired: the filename isn't related to the URL. The pattern a website's URLs follow is also fully customizable.

For more information, the official Jekyll documentation is available at <http://jekyllrb.com/>.

## Create a Configuration File

---

The configuration file has settings that can affect many aspects of the website, including customization of URLs and declaring custom variables.

The configuration file is required for a Jekyll site to function. It can be customized so that the Jekyll site works as desired. Default variables, build settings, server settings and installable extensions are all configured in the configuration file.

To create a configuration file:

1. Create a `_config.yml` in your site's root directory.
2. Create a section to store your default Jekyll variables, as described in [Create Default Variables](#) on page 5

```
defaults:
  -
    scope:
      path: ""
      type: "pages"
    values:
      layout: "default"
```

3. Set any language processing variables, such as Github Flavored Markdown (GFM).

```
kramdown:
  input: GFM
```

These settings allow Jekyll to understand the format in which posts are written.

4. Add any other customized settings, such as extensions, into the file. These settings are available in [Configuration File Settings](#) on page 4

## Configuration File Settings

---

The configuration file can be customized so the Jekyll site works as desired. These settings can affect many aspects of the website, including the customization of URLs and the declaration of custom variables.

### Configuration Settings

Setting	Description	Syntax
source	Jekyll looks for files in this directory	<code>source: DIR</code>
destination	Jekyll creates the website in this directory	<code>destination: DIR</code>
exclude	Jekyll ignores these files or directories	<code>exclude: [DIR, FILE, ...]</code>
include	Jekyll includes these files or directories	<code>include: [DIR, FILE, ...]</code>
timezone	Overrides the timezone with a different <a href="#">IANA Time Zones</a>	<code>timezone: TIMEZONE</code>
host	Hostname of server	<code>host: HOSTNAME</code>
port	Website port	<code>port: PORT</code>
baseurl	URL of the website	<code>baseurl: URL</code>

### Related concepts

[What are Variables?](#) on page 4

Jekyll provides the ability to pass variables between static files. Variables can be used to pass customized data from a post, or be used to set a default value for the entire website.

### Related tasks

[Create Default Variables](#) on page 5

Default variables are declared in the configuration file and can be reused throughout the website.

## What are Variables?

---

Jekyll provides the ability to pass variables between static files. Variables can be used to pass customized data from a post, or be used to set a default value for the entire website.

Jekyll variables are stored in two ways: as a variable belonging to the site, or as a variable belonging to the page.

Site variables will be consistent across all posts. The title of the site would be an effective site variables: it doesn't change depending on the page being viewed. Site variables are declared in the configuration file.

Page variables do change based on an individual post. The title of a blog post would be an effective page variable. Page variables are declared in the YAML Front Matter.

### Related tasks

[Create Default Variables](#) on page 5

Default variables are declared in the configuration file and can be reused throughout the website.

### Related reference

[Configuration File Settings](#) on page 4

The configuration file can be customized so the Jekyll site works as desired. These settings can affect many aspects of the website, including the customization of URLs and the declaration of custom variables.

## Create Default Variables

---

Default variables are declared in the configuration file and can be reused throughout the website.

Items such as the title of the website, the author, or layout settings make good default variables. Default variables can have their values overwritten by individual posts. If the post does not override the default variable, the default variable's value will be used for that post.

To create a default variable:

1. Pick a setting that will be reused on your website.
2. Decide the appropriate value to be the default of this setting.  
This should be a value that will not look out of place if a post does not have an override. This value may be shown on the website, even if you intend to override it.
3. Open the Jekyll configuration file. On a new line, enter the name of the variable and set it equal to the desired default value.



**Attention:** Be sure the variable is not nested under other variables, or it may cause the site to build incorrectly.

An variable named `backgroundColor` whose default value is red would look like this in the configuration file:

```
backgroundColor: red
```

The variable should now be accessible from the website's posts and layout.

### Related concepts

[What are Variables?](#) on page 4

Jekyll provides the ability to pass variables between static files. Variables can be used to pass customized data from a post, or be used to set a default value for the entire website.

### Related reference

[Configuration File Settings](#) on page 4

The configuration file can be customized so the Jekyll site works as desired. These settings can affect many aspects of the website, including the customization of URLs and the declaration of custom variables.

## Write a Post

---

A post is a file filled with site content. Posts are inserted into the layout by Jekyll when it creates the website.

Posts can be a variety of file types. This means the content can be in whatever format suits it the best. Markdown and HTML are the most common file types. Jekyll will take these posts and convert them into a web-friendly format.

To create a post:

1. Create a file with a filename fitting the required format and of an appropriate file type in the `_posts` folder.  
The filename of a post must follow a specific format. This format indicates the date the post was made, and the topic of the post. The following format for the filename is required: `YEAR-MONTH-DAY-title.extension`  
An example filename would be `2015-11-25-Jekyll.html`
2. Open the file in a text editor of your choice.
3. Create YAML Front Matter for your post. YAML Front Matter is described in [What is YAML Front Matter?](#) on page 6
4. Write content for your post and save it.

5. Open the post in your browser and double check that it looks correct.

## What is YAML Front Matter?

---

YAML Front Matter is used to tell Jekyll how to display and use a post. Front Matter is used to override default variables, and set the values of other properties.

Front Matter must be the first few lines in a file. Jekyll detects that a file starts with Front Matter and knows that the file needs to be processed. Jekyll then parses the YAML found in the Front Matter and handles the information as needed.

Front Matter is separated from the rest of the file by three dashes on their own line at the start and the conclusion of the YAML. There may only be one section of Front Matter in each file.

### Related tasks

[Override Variables](#) on page 6

Variables declared in the configuration file can be overridden a post's YAML Front Matter. This allows data to be customized based on the content the reader is viewing.

## Override Variables

---

Variables declared in the configuration file can be overridden a post's YAML Front Matter. This allows data to be customized based on the content the reader is viewing.

Overridden variables are only changed for the individual post. For an example, if a post is written by a new author, the author variable may need to be overridden with the new author's name. All other posts will retain the default author's name.

To override a variable:

1. Be sure you have your default variable declared, as described in [Create Default Variables](#) on page 5
2. Create your post, as described in [Write a Post](#) on page 5
3. In the YAML Front Matter of the post created in step 2, write the name of the variable. Set it equal to the value desired for this post.

A variable named `backgroundColor` set to be red would look like this in the YAML Front Matter:

```
---
backgroundColor: red
---
```

When viewing this post on the website, the post should reflect the changed setting. This changed setting will not be reflected on other posts.

### Related concepts

[What is YAML Front Matter?](#) on page 6

YAML Front Matter is used to tell Jekyll how to display and use a post. Front Matter is used to override default variables, and set the values of other properties.