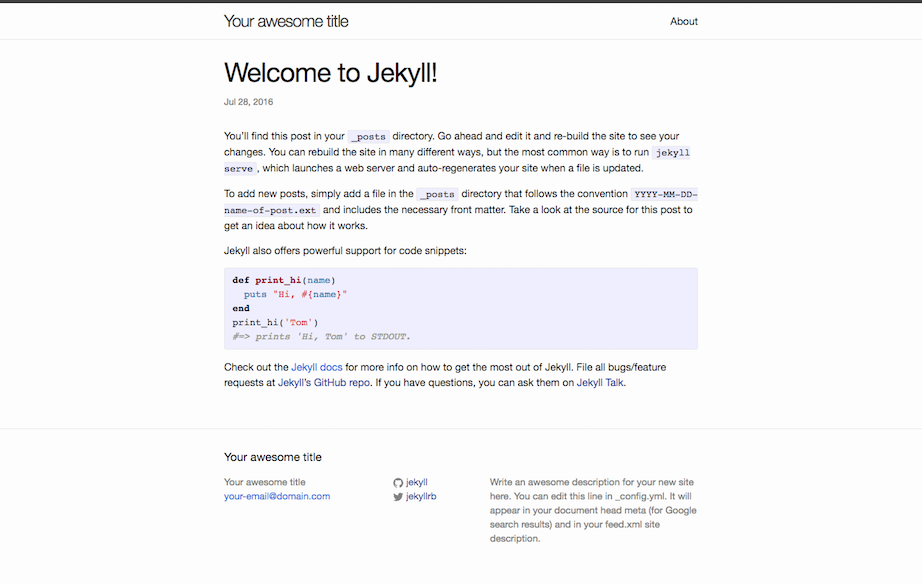


**minima**

*Minima is a one-size-fits-all Jekyll theme for writers*. It's Jekyll's default (and first) theme. It's what you get when you run jekyll new.

[](https://github.com/jekyll/minima/blob/master/screenshot.png)

**Installation**

Add this line to your Jekyll site's Gemfile:

gem "minima"

And add this line to your Jekyll site:

theme: minima

And then execute:

$ bundle

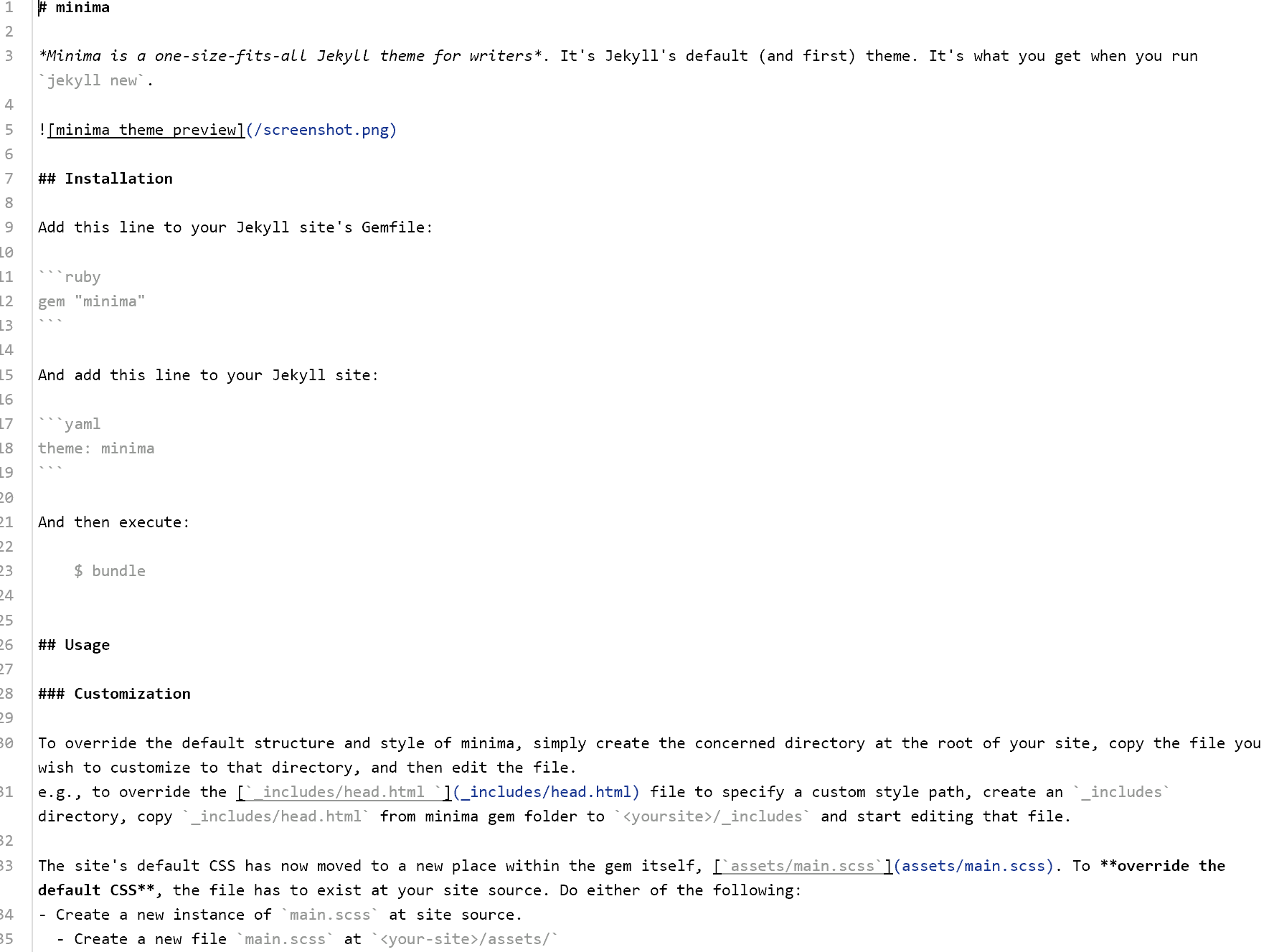
**Usage**

**Customization**

To override the default structure and style of minima, simply create the concerned directory at the root of your site, copy the file you wish to customize to that directory, and then edit the file. e.g., to override the [\_includes/head.html](https://github.com/jekyll/minima/blob/master/_includes/head.html) file to specify a custom style path, create an \_includes directory, copy \_includes/head.html from minima gem folder to <yoursite>/\_includes and start editing that file.

The site's default CSS has now moved to a new place within the gem itself, [assets/main.scss](https://github.com/jekyll/minima/blob/master/assets/main.scss). To **override the default CSS**, the file has to exist at your site source. Do either of the following:

* Create a new instance of main.scss at site source.
  + Create a new file main.scss at <your-site>/assets/
  + Add the frontmatter dashes, and
  + Add @import "minima";, to <your-site>/assets/main.scss
  + Add your custom CSS.



---

layout: post

title: "Welcome to Jekyll!"

date: 2016-10-13 11:39:43 -0400

categories: jekyll update

---

You’ll find this post in your `\_posts` directory. Go ahead and edit it and re-build the site to see your changes. You can rebuild the site in many different ways, but the most common way is to run `jekyll serve`, which launches a web server and auto-regenerates your site when a file is updated.

To add new posts, simply add a file in the `\_posts` directory that follows the convention `YYYY-MM-DD-name-of-post.ext` and includes the necessary front matter. Take a look at the source for this post to get an idea about how it works.

Jekyll also offers powerful support for code snippets:

{% highlight ruby %}

def print\_hi(name)

puts "Hi, #{name}"

end

print\_hi('Tom')

#=> prints 'Hi, Tom' to STDOUT.

{% endhighlight %}

Check out the [Jekyll docs][jekyll-docs] for more info on how to get the most out of Jekyll. File all bugs/feature requests at [Jekyll’s GitHub repo][jekyll-gh]. If you have questions, you can ask them on [Jekyll Talk][jekyll-talk].

[jekyll-docs]: http://jekyllrb.com/docs/home

[jekyll-gh]: https://github.com/jekyll/jekyll

[jekyll-talk]: <https://talk.jekyllrb.com/>

