

Adding a theme to your GitHub Pages site using Jekyll

In this article

Adding a theme

Customizing your theme's CSS

Customizing your theme's HTML layout

Further reading


You can personalize your Jekyll site by adding and customizing a theme.

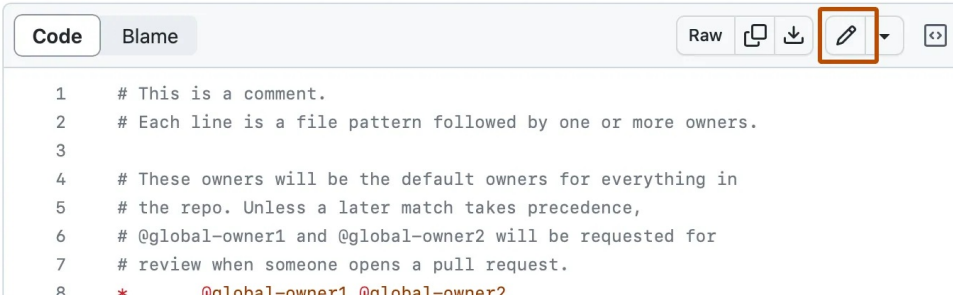
GitHub Pages is available in public repositories with GitHub Free and GitHub Free for organizations, and in public and private repositories with GitHub Pro, GitHub Team, GitHub Enterprise Cloud, and GitHub Enterprise Server. For more information, see "[GitHub's plans](#)."

People with write permissions for a repository can add a theme to a GitHub Pages site using Jekyll.

If you are publishing from a branch, changes to your site are published automatically when the changes are merged into your site's publishing source. If you are publishing from a custom GitHub Actions workflow, changes are published whenever your workflow is triggered (typically by a push to the default branch). If you want to preview your changes first, you can make the changes locally instead of on GitHub Enterprise Cloud. Then, test your site locally. For more information, see "[Testing your GitHub Pages site locally with Jekyll](#)."

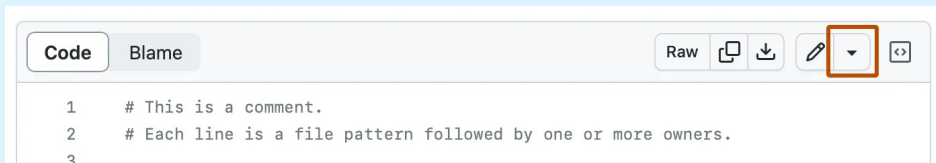
Adding a theme

- 1 On GitHub Enterprise Cloud, navigate to your site's repository.
- 2 Navigate to the publishing source for your site. For more information, see "[Configuring a publishing source for your GitHub Pages site](#)."
- 3 Navigate to `_config.yml`.
- 4 In the upper right corner of the file view, click  to open the file editor.



```
1  # This is a comment.
2  # Each line is a file pattern followed by one or more owners.
3
4  # These owners will be the default owners for everything in
5  # the repo. Unless a later match takes precedence,
6  # @global-owner1 and @global-owner2 will be requested for
7  # review when someone opens a pull request.
8  *           @global-owner1 @global-owner2
```

Note: Instead of editing and committing the file using the default file editor, you can optionally choose to use the [github.dev code editor](#) by selecting the ▼ dropdown menu and clicking **github.dev**. You can also clone the repository and edit the file locally via GitHub Desktop by clicking **GitHub Desktop**.



5 Add a new line to the file for the theme name.

- To use a supported theme, type `theme: THEME-NAME`, replacing *THEME-NAME* with the name of the theme as shown in the README of the theme's repository. For a list of supported themes, see "[Supported themes](#)" on the GitHub Pages site. For example, to select the Minima theme, type `theme: minima`.
- To use any other Jekyll theme hosted on GitHub, type `remote_theme: THEME-NAME`, replacing *THEME-NAME* with the name of the theme as shown in the README of the theme's repository.

6 Click **Commit changes...**

7 In the "Commit message" field, type a short, meaningful commit message that describes the change you made to the file. You can attribute the commit to more than one author in the commit message. For more information, see "[Creating a commit with multiple authors](#)."

8 If you have more than one email address associated with your account on GitHub.com, click the email address drop-down menu and select the email address to use as the Git author email address. Only verified email addresses appear in this drop-down menu. If you enabled email address privacy, then a no-reply will be the default commit author email address. For more information about the exact form the no-reply email address can take, see "[Setting your commit email address](#)."

9 Below the commit message fields, decide whether to add your commit to the current branch or to a new branch. If your current branch is the default branch, you should choose to create a new branch for your commit and then create a pull request. For more information, see "[Creating a pull request](#)."

☐ Commit directly to the `main` branch.

☒ Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

octocat-patch-1

Propose changes

Cancel

10 Click **Commit changes** or **Propose changes**.

Customizing your theme's CSS [🔗](#)

These instructions work best with themes that are officially supported by GitHub Pages. For a complete list of supported themes, see "[Supported themes](#)" on the GitHub Pages site.

Your theme's source repository may offer some help in customizing your theme. For example, see [Minima's README](#).

- 1 On GitHub Enterprise Cloud, navigate to your site's repository.
- 2 Navigate to the publishing source for your site. For more information, see "[Configuring a publishing source for your GitHub Pages site](#)."
- 3 Create a new file called `/assets/css/style.scss`.
- 4 Add the following content to the top of the file:

```
---  
---  
  
@import "{{ site.theme }}";
```

- 5 Add any custom CSS or Sass (including imports) you'd like immediately after the `@import` line.

Customizing your theme's HTML layout [🔗](#)

These instructions work best with themes that are officially supported by GitHub Pages. For a complete list of supported themes, see "[Supported themes](#)" on the GitHub Pages site.

Your theme's source repository may offer some help in customizing your theme. For example, see [Minima's README](#).

- 1 On GitHub, navigate to your theme's source repository. For example, the source repository for Minima is <https://github.com/jekyll/minima>.
- 2 In the `_layouts` folder, navigate to your theme's `default.html` file.
- 3 Copy the contents of the file.
- 4 On GitHub Enterprise Cloud, navigate to your site's repository.
- 5 Navigate to the publishing source for your site. For more information, see "[Configuring a publishing source for your GitHub Pages site](#)."

- 6 Create a file called `_layouts/default.html`.
- 7 Paste the default layout content you copied earlier.
- 8 Customize the layout as you'd like.

Further reading

- "[Creating new files](#)"

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)