

Welcome to Introduction to Jekyll

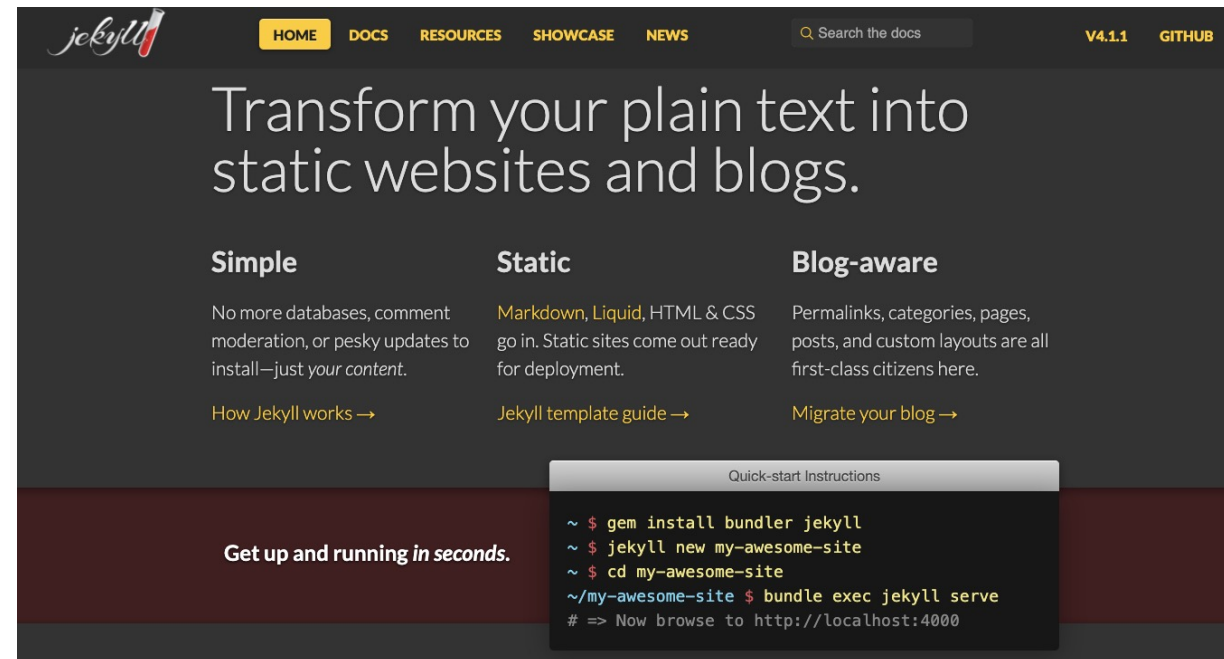
Make you own website in a day



Transform your plain text into
static websites and blogs.

What is Jekyll?

- Jekyll is a **Static Site Generator**
- **Static site** are the ones that are fixed and display the same **content for every user**
- Written in **Ruby**
- Best suited for **personal and organizational sites**
- Jekyll is the **most popular** static site generator (2017)
- Easy to deploy on **Github.io** and many other platforms



How to build my first site?

Step – 1 Install Ruby and Jekyll

- Mac user's **Open terminal**
- Install [Jekyll & Bundler](#)
- Both are called [Gems](#)
- Gems are **code that you can include** in your **Ruby** project
- Create a project using `> mkdir "dirname"`
- Run `> bundle init` and it creates a Gemfile
- **Gemfile** is a list of **Gems** used by the site

Installation

Jekyll is a Ruby program so you need to install Ruby on your machine to begin with. Head over to the [install guide](#) and follow the instructions for your operating system.

With Ruby setup you can install Jekyll by running the following in your terminal:

terminal

```
gem install jekyll bundler
```

To create a new **Gemfile** to list your project's dependencies run:

terminal

```
bundle init
```

Now edit the **Gemfile** and add jekyll as a dependency:

```
gem "jekyll"
```

Finally run **bundle** to install jekyll for your project.

Step – 2: Create an index.html file

- **Create a new directory** for your project
- **Initialize** your directory with a Git repository
- Create a file named **index.html** in the directory
- **Design** the webpage using **html**

Create a site

It's time to create a site! Create a new directory for your site, you can name it whatever you'd like. Through the rest of this tutorial we'll refer to this directory as "root".

If you're feeling adventurous, you can also initialize a Git repository here. One of the great things about Jekyll is there's no database. All content and site structure are files which a Git repository can version. Using a repository is completely optional but it's a great habit to get into. You can learn more about using Git by reading through the [Git Handbook](#).

Let's add your first file. Create `index.html` in the root with the following content:

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Home</title>
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

Step – 3: Build your site

- In the same directory, run > **Jekyll serve**
- It **automatically rebuilds** the site anytime the contents are changed
- Go to <http://localhost:4000> in your browser to check your newly created website

Build

Jekyll is a static site generator so we need Jekyll to build the site before we can view it. There are two commands you can run in the root of your site to build it:

- `jekyll build` - Builds the site and outputs a static site to a directory called `_site`.
- `jekyll serve` - Does the same thing except it rebuilds any time you make a change and runs a local web server at `http://localhost:4000`.

When you're developing a site you'll use `jekyll serve` as it updates with any changes you make.

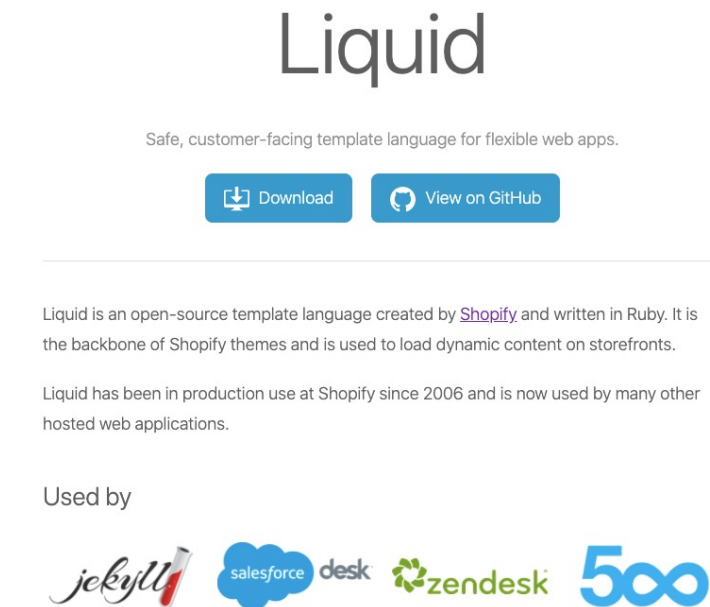
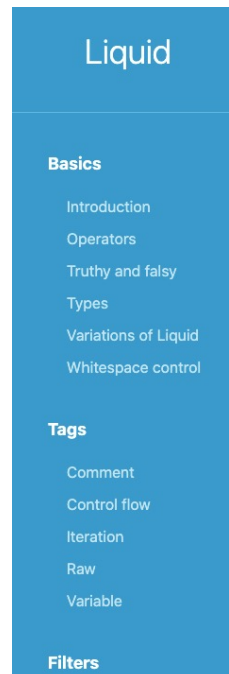
Run `jekyll serve` and go to `http://localhost:4000` in your browser. You should see "Hello World!"

Well, you might be thinking what's the point in this? Jekyll just copied an HTML file from one place to another. Well patience young grasshopper, there's still much to learn!

Liquid – Makes Jekyll Cool

Liquid

- [Open-source template](#) language written in Ruby
- Created by [Shopify](#)
- Helps to **load dynamic contents** on the website
- It is used by **many other platforms** and Shopify use it in their production systems since 2006



Main parts of Liquid

- **Objects** output the content of the variable
- **Tags** create the logic and control flow like any other if-else statement
- **Filters** are used to change the output of the Liquid objects

```
{{ page.title }}
```

Objects

Tags

```
{% if page.show_sidebar %}  
  <div class="sidebar">  
    sidebar content  
  </div>  
{% endif %}
```

```
{{ "hi" | capitalize }}
```

Filters

Front Matter

How does Jekyll know which file to parse?

- Front Matter is a snippet of **YAML**
- It sits between **two triple dashed lines** at the top of the file
- Used to **set variables** for the page
- **Every page should have Front Matter** if you want Jekyll to process it

Use front matter

Let's change the `<title>` on your site to populate using front matter:

```
---
title: Home
---
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>{{ page.title }}</title>
  </head>
  <body>
    <h1>{{ "Hello World!" | lowercase }}</h1>
  </body>
</html>
```

Layouts

How to have multiple pages?

- **Websites** typically have **multiple pages**
- Each page can have **different layout**
- **_layouts** directory allows you to create **multiple templates** for different pages
- **Templates** wrap around your content

Creating a layout

Using a layout is a much better choice. Layouts are templates that wrap around your content. They live in a directory called **_layouts**.

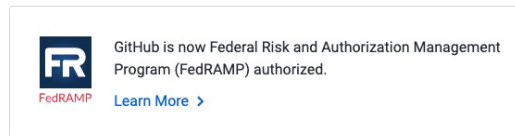
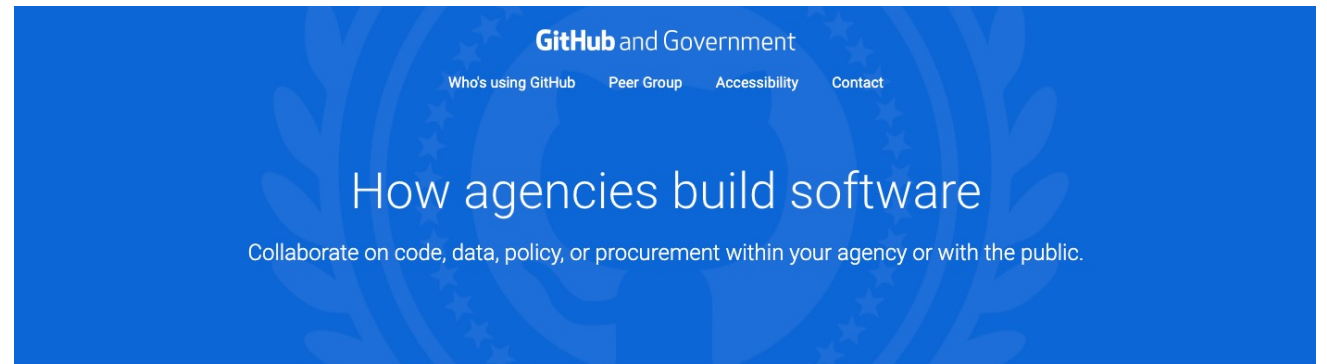
Create your first layout at **_layouts/default.html** with the following content:

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>{{ page.title }}</title>
  </head>
  <body>
    {{ content }}
  </body>
</html>
```

Pragmatic Approach

Steps to create a professional site in a day

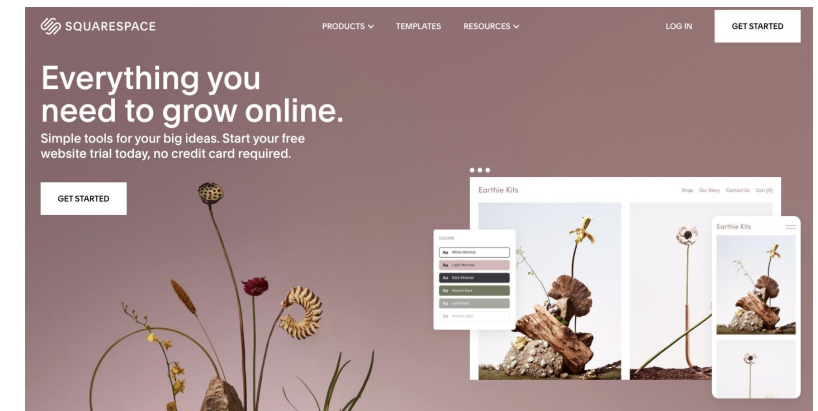
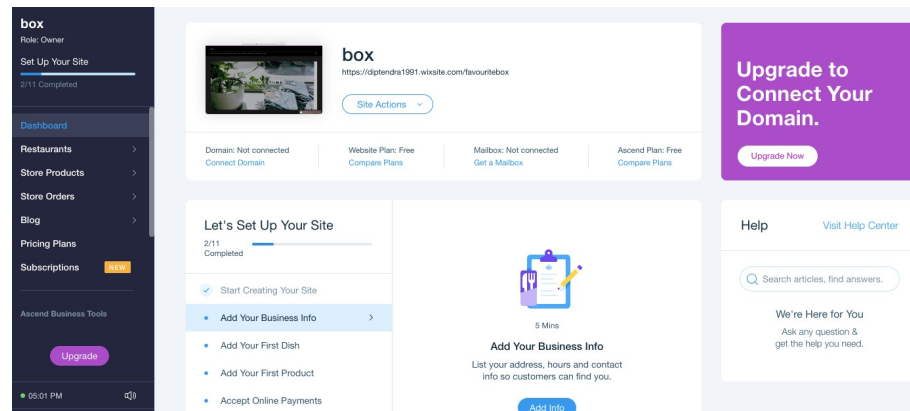
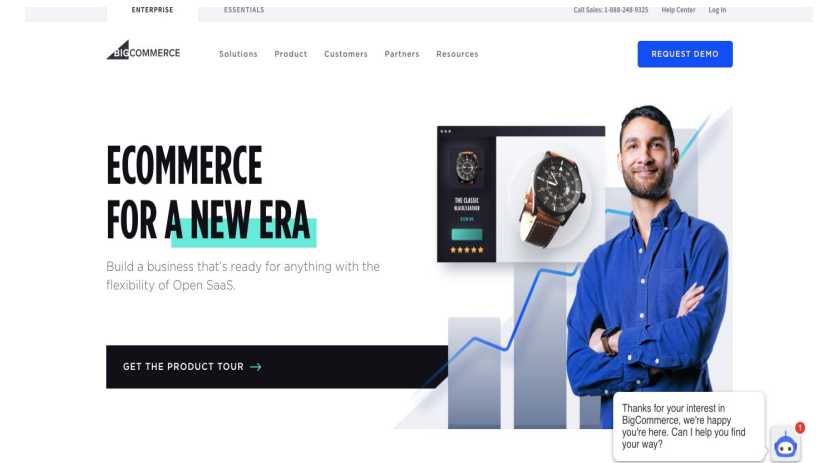
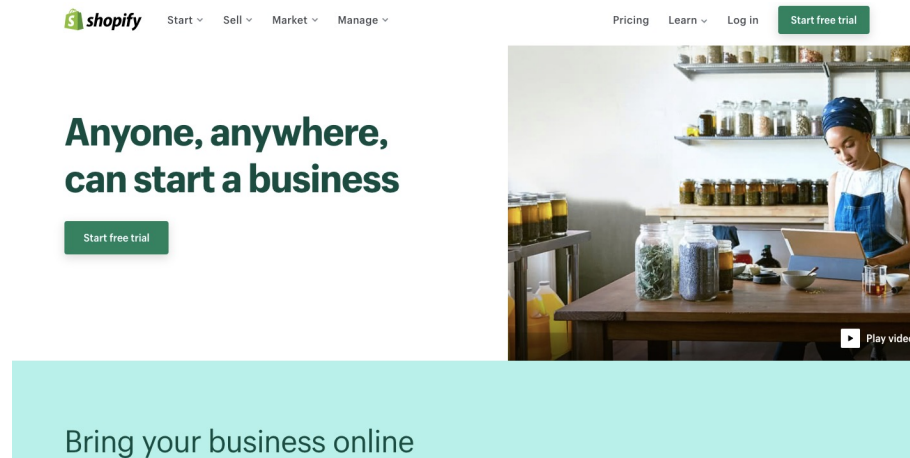
1. Find a **Jekyll website on Github** that suits your need both **functionally & aesthetically** - [Examples](#)
2. **Clone** the repository
3. Change the **necessary files**
4. Files to change - **_config.yml, _posts, about.md, _includes**



Other Options?

Different platforms to create a professional site

- [Wix](#)
- [WordPress.org](#)
- [BigCommerce](#)
- [Shopify](#)
- [Squarespace](#)



Resources

1. Installation - <https://jekyllrb.com/docs/installation/>
2. Tutorial - <https://jekyllrb.com/docs/step-by-step/01-setup/>
3. Liquid - <https://shopify.github.io/liquid/>
4. Exercise - <https://github.com/Diptendra/jekyll-workshop>
5. Blog Template - <https://github.com/Diptendra/diptendra.github.io>
6. Github Pages - <https://github.com/collections/github-pages-examples>