



Sinatra

Fork me on GitHub

[README](#)
[DOCUMENTATION](#)
[BLOG](#)
[CONTRIBUTE](#)
[CREW](#)
[CODE](#)
[ABOUT](#)
[Gittip](#)

Using Extensions

Extensions provide helper or class methods for Sinatra applications. These methods are customarily listed and described on extensions home pages, many of which are listed below.

Using an extension is usually as simple as installing a gem or library and requiring a file. Consult these steps if you run into problems:

1. Install the gem or vendor the library with your project: `gem install sinatra-prawn`
2. Require the extension in your application: `require 'sinatra/prawn'`
3. If your application is “classic” (i.e., you require 'sinatra' and define the application in the main/top-level context), you're done. The extension methods should be available to your application.
4. If your application subclasses `Sinatra::Base`, you have to register the extension in your subclass: `register Sinatra::Prawn`

Helper Extensions

These extensions add helper methods to the request context:

1. [sinatra-prawn](#) adds support for PDF rendering with Prawn templates.
2. [sinatra-markaby](#) enables rendering of HTML files using markaby templates.
3. [sinatra-maruku](#) provides Maruku templates for a Sinatra application.
4. [sinatra-rdiscount](#) provides RDiscount templates for a Sinatra application.
5. [sinatra-effigy](#) provides Effigy templates and views for a Sinatra application.
6. [sinatra-content-for](#) provides `content_for` helper similar to Rails one.
7. [sinatra-url-for](#) construct absolute paths and full URLs to actions in a Sinatra application
8. [sinatra-static-assets](#) implements `image_tag`, `stylesheet_link_tag`, `javascript_script_tag` and `link_tag` helpers. These helpers construct correct absolute paths for applications dispatched to sub URI.
9. [sinatra-mapping](#) implements `map` in the DSL syntax commands which creates dynamically `mapname_path` method.
10. [sinatra-more](#) Library with agnostic generators, form builders, named route mappings, easy mailer support among other functionality.
11. [sinatra-authorization](#) HTTP auth helpers
12. [sinatra-simple-navigation](#) enables creating navigations using the simple-navigation gem.

DSL Extensions

These extensions add methods to Sinatra's application DSL:

1. [snap](#) provides support for named routes and helper methods for building URLs for use in links and redirects.

Extensions which handle setup and configuration

1. [sinatra-mongoid](#) sets up a MongoDB connection, provides Mongoid to your app, and provides options for configuration.

Add more! See [Writing Extensions](#) for more information about creating your own.