



# Sinatra

[Fork me on GitHub](#)

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

## Sinatra::Contrib

Starting with 1.3.0, every Sinatra release will be followed by a Sinatra::Contrib release which will include collection of common Sinatra extensions. Feel free to fork the [source code](#) and start contributing.

This extensions are organized in **common** and **custom extensions**. *Common extensions* will not add significant overhead or change any behavior of already existing APIs, and do not add any dependencies not already installed with Sinatra::Contrib. *Custom extensions*, on the other hand, may add additional dependencies and enhance the behavior of the existing APIs.

## Installation

All Sinatra::Contrib extensions are bundled in the `sinatra-contrib` gem.

```
gem install sinatra-contrib
```

## Usage

### In Classic Style Applications

A single extension (example: `sinatra-content-for`):

```
require 'sinatra'
require 'sinatra/content_for'
```

Common extensions:

```
require 'sinatra'
require 'sinatra/contrib'
```

All extensions:

```
require 'sinatra'
require 'sinatra/contrib/all'
```

## In Modular Style Applications

A single extension (example: `sinatra-content-for` and `sinatra-namespace`):

```
require 'sinatra/base'
require 'sinatra/content_for'
require 'sinatra/namespace'

class MyApp < Sinatra::Base
  # Note: Some modules are extensions, some helpers, see the specific
  # documentation or the source
  helpers Sinatra::ContentFor
  register Sinatra::Namespace
end
```

Common extensions:

```
require 'sinatra/base'
require 'sinatra/contrib'

class MyApp < Sinatra::Base
  register Sinatra::Contrib
end
```

All extensions:

```
require 'sinatra/base'
require 'sinatra/contrib/all'

class MyApp < Sinatra::Base
  register Sinatra::Contrib
end
```

## What Is Included?

### Common Extensions

- [Sinatra::ConfigFile](#): Load your application's configuration from YAML files.
- [Sinatra::ContentFor](#): Adds Rails-style *content\_for* helpers to Erb, Erubis, Haml and Slim.
- [Sinatra::Cookies](#): Provides the *cookies* helper for reading and writing cookies.
- [Sinatra::JSON](#): Easily return JSON documents.
- [Sinatra::LinkHeader](#): Helpers for generating link HTML tags and corresponding Link HTTP headers.
- [Sinatra::MultiRoute](#): Create multiple routes with just one statement.
- [Sinatra::Namespace](#): Adds namespaces to Sinatra.

- Sinatra::RespondWith: Choose action and/or template depending on the incoming request.
- Sinatra::Streaming: Improves the streaming API by making the *stream* object immitate an IO object.

## Custom Extensions

- Sinatra::Decompile: Recreates path patterns from Sinatra's internal data structures (used by other extensions).
- Sinatra::Reloader: Automatically reloads Ruby files on code changes.

## Other Tools

- Sinatra::Extension: Mixin for writing your own Sinatra extensions.