Dafydd Crosby

Fork me on GitHub

Using i18n in Sinatra

Tweet ◁ 4

I've been playing a lot with Sinatra lately, and so far I'm pretty impressed with it. It works wonderfully for small operations, but there's a few creature comforts missing, and one of them is internationalization (i18n for short). Fortunately, it's possible to use the 'i18n' gem (which is used in Rails) to accomplish this.

Here's a small Sinatra app that demonstrates how to do this:

```ruby
require 'sinatra'
require 'i18n'

# We're going to load the paths to locale files,
I18n.load_path += Dir[File.join(File.dirname(__FILE__), 'locales', '*.yml').to_s]

helpers do
  def get_locale
    # Pulls the browser's language
    @env["HTTP_ACCEPT_LANGUAGE"][0,2]
  end

  def t(*args)
    # Just a simple alias
    I18n.t(*args)
  end
end

get '/' do
  t 'message', locale: get_locale
end
```

Then just create your locale files in YAML like you would for Rails:

locales/en.yml

```yaml
en:
  message: "Hello world"
```

locales/fr.yml

```yaml
fr:
  message: "Bonjour tout le monde!"
```

blogroll   social

Proudly powered by [Pelican](#), which takes great advantage of [Python](#).

The theme is by [Smashing Magazine](#), thanks!