LEARNING

ruby-on-rails-3

#ruby-on-

rails-3

Table of Contents

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: ruby-on-rails-3

It is an unofficial and free ruby-on-rails-3 ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official ruby-on-rails-3.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with ruby-on-rails-3

Remarks

This section provides an overview of what ruby-on-rails-3 is, and why a developer might want to use it.

It should also mention any large subjects within ruby-on-rails-3, and link out to the related topics. Since the Documentation for ruby-on-rails-3 is new, you may need to create initial versions of those related topics.

Examples

Installating Rails on mac.

You would need to install `ruby` before you can install rails.

Mac already comes with `ruby` installed based on how recent your `macOS` is? Depending on what `ruby` version you want for your development, the best way to install `Ruby` is to use RVM. In your terminal, type the command below listed in steps:

1. Install `rvm`

```
curl -sSL https://get.rvm.io | bash -s stable --ruby
```

2. For `Rails 3`, best version to install is `ruby 1.9.3`

```
rvm install 1.9.3
ruby -v #=> 1.9.3
```

3. Set your `Ruby` version

```
rvm use 1.9.3 --default
```

4. Install Rails *(this rails version requires ruby-version >=1.9.3)*

```
gem install rails -v 4.2.7.1
rails -v #=> 4.2.7.1
```

5. Install `rails app`

```
rails new my_first_app #(this will install the app for you.)
cd my_first_app
rails s #(run the server)
```

6. Open the browser and type below in your URL.

```
http://localhost:3000
```

Message saying 'Welcome to rails' will be displayed or similar.

Hello World in Rails

1. **Say "Hello", Rails**

   To get Rails saying "Hello", you need to create at minimum a controller and a view.

   A controller's purpose is to receive specific requests for the application. Routing decides which controller receives which requests. Often, there is more than one route to each controller, and different routes can be served by different actions. Each action's purpose is to collect information to provide it to a view.

   A view's purpose is to display this information in a human readable format. An important distinction to make is that it is the controller, not the view, where information is collected. The view should just display that information. By default, view templates are written in a language called eRuby (Embedded Ruby) which is processed by the request cycle in Rails before being sent to the user.

   To create a new controller, you will need to run the "controller" generator and tell it you want a controller called "Welcome" with an action called "index", just like this:

   ```
   $ bin/rails generate controller Welcome index
   ```

   Rails will create several files and a route for you.

   ```
   create    app/controllers/welcome_controller.rb
    route    get 'welcome/index'
   invoke    erb
   create      app/views/welcome
   create      app/views/welcome/index.html.erb
   invoke    test_unit
   create      test/controllers/welcome_controller_test.rb
   invoke    helper
   create      app/helpers/welcome_helper.rb
   invoke    assets
   invoke      coffee
   create        app/assets/javascripts/welcome.coffee
   invoke      scss
   create        app/assets/stylesheets/welcome.scss
   ```

2. Most important of these are of course the controller, located at `app/controllers/welcome_controller.rb` and the view, located at `app/views/welcome/index.html.erb`.

   Open the `app/views/welcome/index.html.erb` file in your text editor. Delete all of the existing code in the file, and replace it with the following single line of code:

```
<h1>Hello, Rails!</h1>
```

3. Now that we have made the controller and view, we need to tell Rails when we want "Hello, Rails!" to show up. In our case, we want it to show up when we navigate to the root URL of our site, `http://localhost:3000`.

   Next, you have to tell Rails where your actual home page is located.Edit the file by adding the line of code root 'welcome#index'. It should look something like the following:

```
Rails.application.routes.draw do
  get 'welcome/index'

  root 'welcome#index'
end
```

4. root `welcome#index` tells Rails to map requests to the root of the application to the welcome controller's index action and get `welcome/index` tells Rails to map requests to `http://localhost:3000/welcome/index` to the `welcome controller's index action`. This was created earlier when you ran the controller generator (bin/rails generate controller Welcome index).

5. Yay, now the moment of truth. Launch web server after restarting your `rails server` and navigate to `http://localhost:3000` in your browser. You'll see the **"Hello, Rails!"** message you put into `app/views/welcome/index.html.erb`, indicating that this new route is indeed going to WelcomeController's index action and is rendering the view correctly.

This Guide is from guides.rubyonrails.org. Happy Hacking!

Read Getting started with ruby-on-rails-3 online: https://riptutorial.com/ruby-on-rails-3/topic/9066/getting-started-with-ruby-on-rails-3

Credits

S. No	Chapters	Contributors
1	Getting started with ruby-on-rails-3	Community, TheMouseMan