



FREE eBook

LEARNING

ruby-on-rails-5

Free unaffiliated eBook created from
Stack Overflow contributors.

**#ruby-on-
rails-5**

Table of Contents

| | |
|--|----------|
| About | 1 |
| Chapter 1: Getting started with ruby-on-rails-5 | 2 |
| Remarks..... | 2 |
| Examples..... | 2 |
| Installation or Setup..... | 2 |
| Creating your first hello world page..... | 2 |
| Setup Ruby On Rails on Ubuntu 17.04 Zesty Zapus..... | 4 |
| Installing Ruby | 4 |
| Installing with rbenv:..... | 4 |
| Installing with rvm:..... | 5 |
| Installing with source:..... | 5 |
| Installing Rails | 5 |
| Setting Up PostgreSQL..... | 6 |
| Credits | 7 |

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [ruby-on-rails-5](#)

It is an unofficial and free ruby-on-rails-5 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-5.

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-5

Remarks

This section provides an overview of what ruby-on-rails-5 is, and why a developer might want to use it.

It should also mention any large subjects within ruby-on-rails-5, and link out to the related topics. Since the Documentation for ruby-on-rails-5 is new, you may need to create initial versions of those related topics.

Examples

Installation or Setup

In windows Platform

Step 1: Installation of Ruby

If you already have installed ruby in your pc then you can skip this step.

- Go to rubyinstaller.org's [download page](#) and download one of the stable version of ruby corresponding to architecture of your windows platform.
- Again download development kit for your ruby version [form here](#).
- Now install ruby.
- This is the time to enhance your ruby installation with ruby development kit. This is required by some of the gems to compile in your pc.
- Extract your ruby development kit file in a folder near your ruby installation path `C:\Ruby-**. For example C:\rubyDevkit.`
- Now open `cmd.exe` and move to the directory where you extracted your devkit.
- Then run this command `ruby dk.rb init`. This will initialize your installation. Then run `ruby dk.rb install`.

Step 2: Install Rails

- After successful installation of ruby. Next step is to install rails. Before installing rails install bundler gem running `gem install bundler` from your command prompt window.
- After installing bundler now run `gem install rails -v version_of_rails` for example `-v 5.0.0.1`.
- If all process are completed without error then you installed rails successfully.

Creating your first hello world page

Create a new rails app `hello-world` from command in Windows and Terminal in Linux.

```
rails new hello-world
```

Now move to new app directory

```
cd hello-world
```

Now generate a controller

```
rails generate controller hello_world index
```

Here `index` is the name of method in `hello_world` controller. You can check it opening the file `app/controllers/hello_world_controller.rb` in your application directory. Code looks like this:

```
class HelloWorldController < ApplicationController
  def index
    end
end
```

A route is automatically added in your `config/routes.rb` file which points to your method. See the code in your `routes.rb` file.

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

  # For details on the DSL available within this file, see
  http://guides.rubyonrails.org/routing.html
end
```

Now open file `app/views/hello_world/index.html.erb` Clear all the content and write

```
Hello, this is my first rails page.
```

Start rails server:

```
rails server
```

Open this url in your browser:

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

You should see:

```
Hello, this is my first rails page
```

Make your new page, your home page. In `routes.rb` file in config folder remove the line `get 'hello_world/index'` and add:

```
root 'hello_world#index'
```

Now open: `http://localhost:3000/` You will see: `Hello, this is my first rails` You are done.

Setup Ruby On Rails on Ubuntu 17.04 Zesty Zapus

This will take about 30 minutes. We will be setting Ruby on Rails Development Environment on Ubuntu 16.10 Yakkety Yak.

You'll want to download the latest Desktop version here: <http://releases.ubuntu.com/17.04/>

Open up your terminal using `Ctrl + Alt + T`.

Installing Ruby

The First step is to install Dependencies For Ruby.

```
sudo apt-get update
sudo apt-get install git-core curl zlib1g-dev build-essential libssl-dev libreadline-dev
libyaml-dev libsqlite3-dev sqlite3 libxml2-dev libxslt1-dev libcurl4-openssl-dev python-
software-properties libffi-dev nodejs
```

We will be using Ruby version 2.4.0 and it is recommended. Install Ruby using one of the three methods. Each have their own benefits, mostly people prefer `rbenv` but if you are familiar with `rvm` then go for it.

Note: Only `rbenv` users need to run `rbenv rehash` after installing `bundler`.

Installing with `rbenv`:

```
cd
git clone https://github.com/rbenv/rbenv.git ~/.rbenv
echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashrc
echo 'eval "$(rbenv init -)"' >> ~/.bashrc
exec $SHELL

git clone https://github.com/rbenv/ruby-build.git ~/.rbenv/plugins/ruby-build
echo 'export PATH="$HOME/.rbenv/plugins/ruby-build/bin:$PATH"' >> ~/.bashrc
exec $SHELL

rbenv install 2.4.0
rbenv global 2.4.0
ruby -v
```

Then install Bundler:

```
gem install bundler
```

Use `rbenv rehash` after installing `bundler`.

Installing with `rvm`:

```
sudo apt-get install libgdbm-dev libncurses5-dev automake libtool bison libffi-dev
gpg --keyserver hkps://keys.gnupg.net --recv-keys 409B6B1796C275462A1703113804BB82D39DC0E3
curl -sSL https://get.rvm.io | bash -s stable
source ~/.rvm/scripts/rvm
rvm install 2.4.0
rvm use 2.4.0 --default
ruby -v
```

Then install Bundler:

```
gem install bundler
```

Installing with source:

```
cd
wget http://ftp.ruby-lang.org/pub/ruby/2.4/ruby-2.4.0.tar.gz
tar -xzf ruby-2.4.0.tar.gz
cd ruby-2.4.0/
./configure
make
sudo make install
ruby -v
```

Then Last step is to install Bundler:

```
gem install bundler
```

Installing Rails

Rails ships with so many dependencies these days, we're going to need to install a JavaScript run-time like NodeJS. This lets you use Coffee-script and the [Asset Pipeline](#) in Rails which combines and minifies your JavaScript to provide a faster production environment.

Install NodeJS using the official repository:

```
curl -sL https://deb.nodesource.com/setup_4.x | sudo -E bash -
sudo apt-get install -y nodejs
```

Now install Rails 5.0.1:

```
gem install rails -v 5.0.1
```

If you are using `rbenv` then run the following command to make rails executable available:

```
rbenv rehash
```

Now Rails is Installed, Run `rails -v` to make sure rails installed properly:

```
rails -v  
# Rails 5.0.1
```

If you are getting a different result for some reason, it means that your environment may not be setup properly.

Setting Up PostgreSQL

Rails ships with `sqlite3` as the default database. Chances are you won't want to use it because it's stored as a simple file on disk. You'll probably want something more robust like MySQL or PostgreSQL.

There is a lot of documentation on both, so you can just pick one that seems like you'll be more comfortable with.

```
sudo sh -c "echo 'deb http://apt.postgresql.org/pub/repos/apt/ xenial-pgdg main' >  
/etc/apt/sources.list.d/pgdg.list"  
wget --quiet -O - http://apt.postgresql.org/pub/repos/apt/ACCC4CF8.asc | sudo apt-key add -  
sudo apt-get update  
sudo apt-get install postgresql-common  
sudo apt-get install postgresql-9.5 libpq-dev
```

PostgreSQL installation doesn't setup a user for you, so you'll need to follow these steps to create a user with permission to create databases. Feel free to replace `Hizqeel` with your username.

```
sudo -u postgres createuser hizqeel -s  
  
# If you would like to set a password for the user, you can do the following  
sudo -u postgres psql  
postgres=# \password hizqeel
```

Read [Getting started with ruby-on-rails-5](https://riptutorial.com/ruby-on-rails-5/topic/9625/getting-started-with-ruby-on-rails-5) online: <https://riptutorial.com/ruby-on-rails-5/topic/9625/getting-started-with-ruby-on-rails-5>

Credits

| S. No | Chapters | Contributors |
|-------|--------------------------------------|---|
| 1 | Getting started with ruby-on-rails-5 | Community , Hizqeel , Nick Beaujean , Pradeep Sapkota |