



eBook Gratuit

APPRENEZ

ruby-on-rails-5

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

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

Table des matières

À propos.....	1
Chapitre 1: Démarrer avec ruby-on-rails-5.....	2
Remarques.....	2
Exemples.....	2
Installation ou configuration.....	2
Création de votre première page de salut.....	3
Installation de Ruby On Rails sur Ubuntu 17.04 Zesty Zapus.....	4
Installation de Ruby.....	4
Installation avec rbenv :.....	4
Installation avec rvm :.....	5
Installation avec source:.....	5
Installation de rails.....	5
Configurer PostgreSQL.....	6
Crédits.....	7

À propos

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

Chapitre 1: Démarrer avec ruby-on-rails-5

Remarques

Cette section fournit une vue d'ensemble de ce qu'est ruby-on-rails-5 et pourquoi un développeur peut vouloir l'utiliser.

Il devrait également mentionner tous les grands sujets dans ruby-on-rails-5, et établir un lien avec les sujets connexes. La documentation de ruby-on-rails-5 étant nouvelle, vous devrez peut-être créer des versions initiales de ces rubriques connexes.

Exemples

Installation ou configuration

Dans Windows Platform

Étape 1: Installation de Ruby

Si vous avez déjà installé Ruby sur votre PC, vous pouvez ignorer cette étape.

- Allez sur la [page de téléchargement](#) de [rubyinstaller.org](#) et téléchargez l'une des versions stables de ruby correspondant à l'architecture de votre plate-forme Windows.
- Encore une fois, téléchargez le kit de développement pour votre [formulaire de version ruby ici](#) .
- Maintenant installez ruby.
- C'est le moment d'améliorer votre installation de rubis avec le kit de développement Ruby. Ceci est requis par certaines des gemmes à compiler dans votre PC.
- Extrayez votre fichier de kit de développement `C:\Ruby-**` dans un dossier proche de votre chemin d'installation `C:\Ruby-**` . Par exemple `C:\rubyDevkit` .
- Ouvrez maintenant `cmd.exe` et déplacez-vous dans le répertoire où vous avez extrait votre devkit.
- Ensuite, lancez cette commande `ruby dk.rb init` . Cela initialisera votre installation. Ensuite, lancez `ruby dk.rb install` .

Étape 2: Installez les rails

- Après l'installation réussie du rubis. L'étape suivante consiste à installer des rails. Avant d'installer les rails, installez bundler gem exécutant un `gem install bundler` depuis votre fenêtre d'invite de commandes.
- Après avoir installé Bundler, lancez maintenant `gem install rails -v version_of_rails` par exemple `-v 5.0.0.1` .
- Si tous les processus sont terminés sans erreur, vous avez installé les rails avec succès.

Création de votre première page de salut

Créez une nouvelle application rails `hello-world` partir de la commande sous Windows et Terminal sous Linux.

```
rails new hello-world
```

Passez maintenant au nouveau répertoire de l'application

```
cd hello-world
```

Maintenant, générez un contrôleur

```
rails generate controller hello_world index
```

Ici, `index` est le nom de la méthode dans le contrôleur `hello_world`. Vous pouvez le vérifier en ouvrant le fichier `app/controllers/hello_world_controller.rb` dans le `app/controllers/hello_world_controller.rb` de votre application. Code ressemble à ceci:

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

Un `route` est automatiquement ajouté dans votre fichier `config/routes.rb` qui pointe vers votre méthode. Voir le code dans votre fichier `routes.rb`.

```
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
```

Maintenant, ouvrez le fichier `app/views/hello_world/index.html.erb` Effacer tout le contenu et écrire

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

Serveur de rails de démarrage:

```
rails server
```

Ouvrez cette URL dans votre navigateur:

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

Tu devrais voir:

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

Faites votre nouvelle page, votre page d'accueil. Dans le fichier routes.rb du dossier config, supprimez la ligne `get 'hello_world/index'` et ajoutez:

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

Maintenant ouvert: <http://localhost:3000/> Vous verrez: Hello, this is my first rails Vous avez terminé.

Installation de Ruby On Rails sur Ubuntu 17.04 Zesty Zapus

Cela prendra environ 30 minutes. Nous allons configurer l'environnement de développement Ruby on Rails sur Ubuntu 16.10 Yakkety Yak.

Vous voudrez télécharger la dernière version de Desktop ici: <http://releases.ubuntu.com/17.04/>

Ouvrez votre terminal en utilisant `Ctrl + Alt + T`.

Installation de Ruby

La première étape consiste à installer les dépendances pour 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
```

Nous allons utiliser Ruby version 2.4.0 et c'est recommandé. Installez Ruby en utilisant l'une des trois méthodes. Chacun a ses propres avantages, la plupart des gens préfèrent rbenv mais si vous êtes familier avec rvm alors allez-y.

Remarque: seuls les utilisateurs de `rbenv` doivent exécuter `rbenv rehash` après avoir installé `bundler`.

Installation avec `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
```

Ensuite, installez Bundler:

```
gem install bundler
```

Utilisez `rbenv rehash` après avoir installé bundler.

Installation avec `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
```

Ensuite, installez Bundler:

```
gem install bundler
```

Installation avec 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
```

La dernière étape consiste à installer Bundler:

```
gem install bundler
```

Installation de rails

Rails est livré avec tellement de dépendances ces jours-ci, nous allons devoir installer un runtime JavaScript comme NodeJS. Cela vous permet d'utiliser Coffee-script et le [pipeline Asset](#) dans Rails, qui combine et réduit votre JavaScript pour fournir un environnement de production plus rapide.

Installez NodeJS en utilisant le dépôt officiel:

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

Maintenant, installez Rails 5.0.1 :

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

Si vous utilisez `rbenv` exécutez la commande suivante pour rendre les rails exécutables disponibles:

```
rbenv rehash
```

Maintenant, Rails est installé, exécutez `rails -v` pour vous assurer que les rails installés correctement:

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

Si vous obtenez un résultat différent pour une raison quelconque, cela signifie que votre environnement peut ne pas être configuré correctement.

Configurer PostgreSQL

Rails est livré avec `sqlite3` comme base de données par défaut. Les chances sont que vous ne voudrez pas l'utiliser parce qu'il est stocké comme un simple fichier sur le disque. Vous voudrez probablement quelque chose de plus robuste comme MySQL ou PostgreSQL.

Il y a beaucoup de documentation sur les deux, vous pouvez donc en choisir un qui vous semble plus à l'aise.

```
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
```

L'installation de PostgreSQL ne configure pas d'utilisateur pour vous. Vous devez donc suivre ces étapes pour créer un utilisateur autorisé à créer des bases de données. N'hésitez pas à remplacer `Hizqeel` par votre nom d'utilisateur.

```
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
```

Lire Démarrer avec ruby-on-rails-5 en ligne: <https://riptutorial.com/fr/ruby-on-rails-5/topic/9625/demarrer-avec-ruby-on-rails-5>

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec ruby-on-rails-5	Community , Hizqeel , Nick Beaujean , Pradeep Sapkota