

# RUBY ON RAILS - INSTALLATION

---

<http://www.tutorialspoint.com/ruby-on-rails/rails-installation.htm>

Copyright © tutorialspoint.com

To develop a web application using Ruby on Rails Framework, you need to install the following software –

- Ruby
- The Rails Framework
- A Web Server
- A Database System

We assume that you already have installed a Web Server and a Database System on your computer. You can use the WEBrick Web Server, which comes with Ruby. Most websites however use Apache or lightTPD web servers in production.

Rails works with many database systems, including MySQL, PostgreSQL, SQLite, Oracle, DB2 and SQL Server. Please refer to a corresponding Database System Setup manual to set up your database.

Let's look at the installation instructions for Rails on Windows and Linux.

## Rails Installation on Windows

Follow the steps given below for installing Ruby on Rails.

### Step 1: Check Ruby Version

First, check if you already have Ruby installed. Open the command prompt and type **ruby -v**. If Ruby responds, and if it shows a version number at or above 2.2.2, then type **gem --version**. If you don't get an error, skip **Install Ruby** step. Otherwise, we'll install a fresh Ruby.

### Step 2: Install Ruby

If Ruby is not installed, then download an installation package from [rubyinstaller.org](http://rubyinstaller.org). Follow the **download** link, and run the resulting installer. This is an exe file **rubyinstaller-2.2.2.x.exe** and will be installed in a single click. It's a very small package, and you'll get RubyGems as well along with this package. Please check the **Release Notes** for more detail.


[About](#) [Download](#) [Help](#) [Contribute](#)

## The easy way to install Ruby on Windows

This is a **self-contained Windows-based installer** that includes the **Ruby language**, an execution environment, important **documentation**, and more.

[Download](#)
[Add-ons](#)


### Latest News

#### RubyInstaller 2.0.0-p645, 2.1.6 and 2.2.2 released

These new releases of Ruby address a security issue (CVE-2015-1855). 2.1.6 and 2.2.2 also address some bugs and fixes. Upgrading to those versions is recommended. You can find the links to those archives in the download section.

April 16, 2015 [Read full article](#)

#### RubyInstaller 2.2.1 released

The RubyInstaller distribution of Ruby 2.2.1 was just released. Check out the downloads section for the installers and binary packages.

March 06, 2015 [Read full article](#)

### Extras

#### Online Ruby Programming Course

If you're new to Ruby, check out this online course from The Pragmatic Studio to learn all the fundamentals of object-oriented programming with Ruby.

#### Online Rails Programming Course

If you're looking to create Ruby on Rails web apps, you'll learn how to build a complete Rails 4 app step-by-step in this online course also from The Pragmatic Studio.

## Step 3: Install Rails

**Install Rails** – With Rubygems loaded, you can install all of Rails and its dependencies using the following command through the command line –

```
C:\> gem install rails
```

```
C:\Windows\system32\cmd.exe

C:\>gem install rails
Fetching: thread_safe-0.3.5.gem (100%)
Successfully installed thread_safe-0.3.5
Fetching: tzinfo-1.2.2.gem (100%)
Successfully installed tzinfo-1.2.2
Fetching: i18n-0.7.0.gem (100%)
Successfully installed i18n-0.7.0
Fetching: activesupport-4.2.3.gem (100%)
Successfully installed activesupport-4.2.3
Fetching: rails-deprecated_sanitizer-1.0.3.gem (100%)
Successfully installed rails-deprecated_sanitizer-1.0.3
Fetching: mini_portile-0.6.2.gem (100%)
Successfully installed mini_portile-0.6.2
Fetching: nokogiri-1.6.6.2-x64-mingw32.gem (100%)
Nokogiri is built with the packaged libraries: libxml2-2.9.2, libxslt-1.1.28, zlib-1.2.8, libiconv-1.14.
Successfully installed nokogiri-1.6.6.2-x64-mingw32
Fetching: rails-dom-testing-1.0.6.gem (100%)
Successfully installed rails-dom-testing-1.0.6
Fetching: loofah-2.0.2.gem (100%)
Successfully installed loofah-2.0.2
Fetching: rails-html-sanitizer-1.0.2.gem (100%)
Successfully installed rails-html-sanitizer-1.0.2
Fetching: erubis-2.7.0.gem (100%)
```

**Note** – The above command may take some time to install all dependencies. Make sure you are connected to the internet while installing gems dependencies.

## Step 4: Check Rails Version

Use the following command to check the rails version.

```
C:\> rails -v
```

### Output

```
Rails 4.2.4
```

Congratulations! You are now on Rails over Windows.

## Rails Installation on Linux

We are installing Ruby On Rails on Linux using **rbenv**. It is a lightweight Ruby Version Management Tool. The **rbenv** provides an easy installation procedure to manage various versions of Ruby, and a solid environment for developing Ruby on Rails applications.

Follow the steps given below to install Ruby on Rails using rbenv tool.

### Step 1: Install Prerequisite Dependencies

First of all, we have to install **git - core** and some ruby dependences that help to install Ruby on Rails. Use the following command for installing Rails dependencies using **yum**.

```
tp> sudo yum install -y git-core zlib zlib-devel gcc-c++ patch readline  
readline-devel libyaml-devel libffi-devel openssl-devel make bzip2 autoconf  
automake libtool bison curl sqlite-devel
```

### Step 2: Install rbenv

Now we will install rbenv and set the appropriate environment variables. Use the following set of commands to get rbenv for git repository.

```
tp> git clone git://github.com/sstephenson/rbenv.git .rbenv  
tp> echo 'export PATH = "$HOME/.rbenv/bin:$PATH"' >> ~/.bash_profile  
tp> echo 'eval "$(rbenv init -)"' >> ~/.bash_profile  
tp> exec $SHELL  
  
tp> git clone git://github.com/sstephenson/ruby-build.git ~/.rbenv/plugins/ruby-  
build  
tp> echo 'export PATH = "$HOME/.rbenv/plugins/ruby-build/bin:$PATH"' <<  
~/.bash_profile  
tp> exec $SHELL
```

### Step 3: Install Ruby

Before installing Ruby, determine which version of Ruby you want to install. We will install Ruby 2.2.3. Use the following command for installing Ruby.

```
tp> rbenv install -v 2.2.3
```

Use the following command for setting up the current Ruby version as default.

```
tp> rbenv global 2.2.3
```

Use the following command to verify the Ruby version.

```
tp> ruby -v
```

### Output

```
ruby 2.2.3p173 (2015-08-18 revivion 51636) [X86_64-linux]
```

Ruby provides a keyword **gem** for installing the supported dependencies; we call them **gems**. If you don't want to install the documentation for Ruby-gems, then use the following command.

```
tp> echo "gem: --no-document" > ~/.gemrc
```

Thereafter, it is better to install the Bundler gem, because it helps to manage your application dependencies. Use the following command to install bundler gem.

```
tp> gem install bundler
```

## Step 4: Install Rails

Use the following command for installing Rails version 4.2.4.

```
tp> install rails -v 4.2.4
```

Use the following command to make Rails executable available.

```
tp> rbenv rehash
```

Use the following command for checking the rails version.

```
tp> rails -v
```

### Output

```
tp> Rails 4.2.4
```

Ruby on Rails framework requires JavaScript Runtime Environment *Node.js* to manage the features of Rails. Next, we will see how we can use Node.js to manage Asset Pipeline which is a Rails feature.

## Step 5: Install JavaScript Runtime

Let us install Node.js from the Yum repository. We will take Node.js from EPEL yum repository. Use the following command to add the EPEL package to the yum repository.

```
tp> sudo yum -y install epel-release
```

Use the following command for installing the Node.js package.

```
tp> sudo yum install nodejs
```

Congratulations! You are now on Rails over Linux.

## Step 6: Install Database

By default, Rails uses sqlite3, but you may want to install MySQL, PostgreSQL, or other RDBMS. This is optional; if you have the database installed, then you may skip this step and it is not mandatory that you have a database installed to start the rails server. For this tutorial, we are using PostgreSQL database. Therefore use the following commands to install PostgreSQL.

```
tp> sudo yum install postgresql-server postgresql-contrib
```

Accept the prompt, by responding with a y. Use the following command to create a PostgreSQL database cluster.

```
tp> sudo postgresql-setup initdb
```

Use the following command to start and enable PostgreSQL.

```
tp> sudo systemctl start postgresql  
tp> sudo systemctl enable postgresql
```

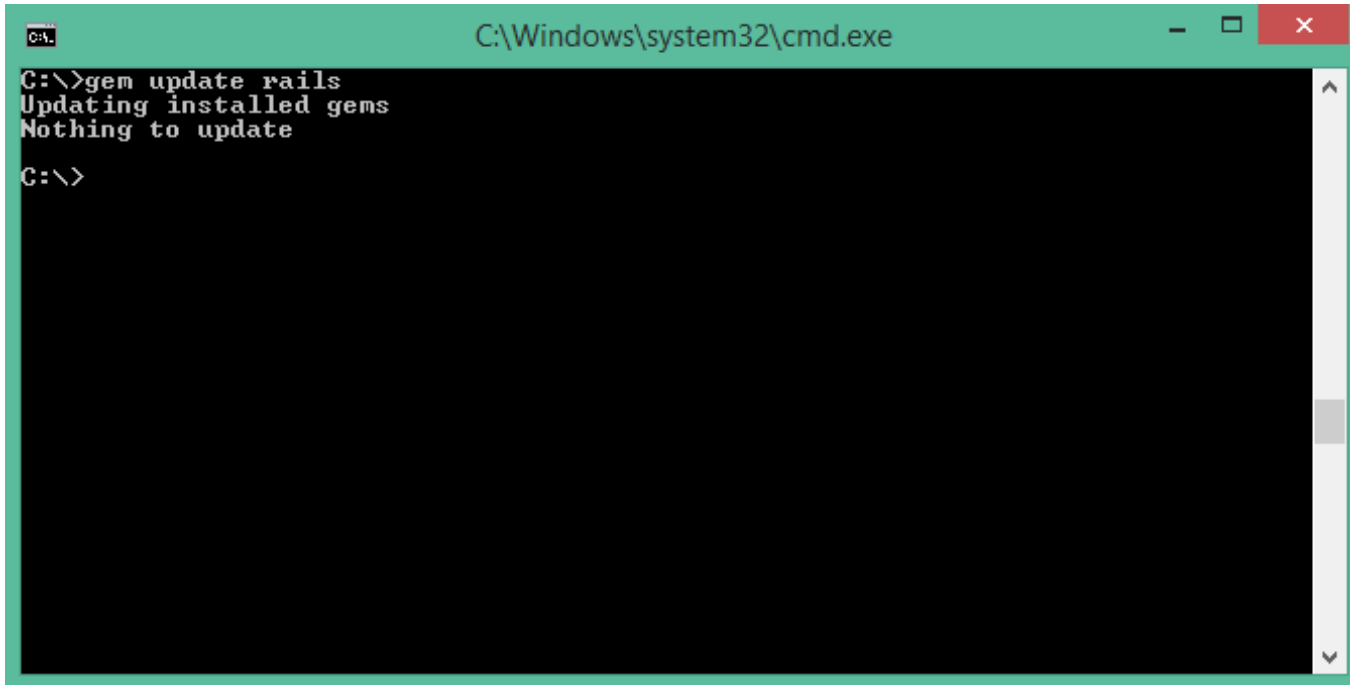
## Keeping Rails Up-to-Date

Assuming you have installed Rails using RubyGems, keeping it up-to-date is relatively easy. We can use the same command in both Windows and Linux platform. Use the following command –

```
tp> gem update rails
```

## Output

The following screenshot shows a Windows command prompt. The Linux terminal also provides the same output.



```
C:\Windows\system32\cmd.exe
C:\>gem update rails
Updating installed gems
Nothing to update
C:\>
```

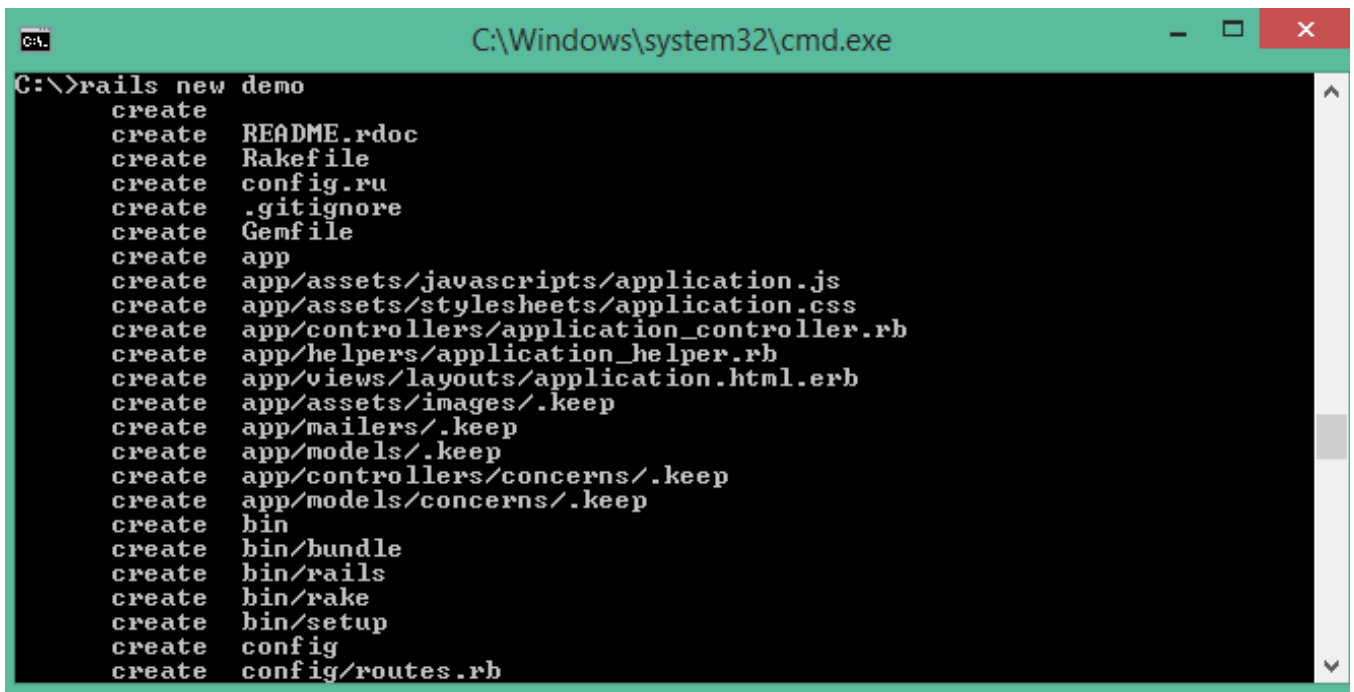
This will automatically update your Rails installation. The next time you restart your application, it will pick up this latest version of Rails. While using this command, make sure you are connected to the internet.

## Installation Verification

You can verify if everything is set up according to your requirements or not. Use the following command to create a demo project.

```
tp> rails new demo
```

## Output



```
C:\Windows\system32\cmd.exe
C:\>rails new demo
create
create  README.rdoc
create  Rakefile
create  config.ru
create  .gitignore
create  Gemfile
create  app
create  app/assets/javascripts/application.js
create  app/assets/stylesheets/application.css
create  app/controllers/application_controller.rb
create  app/helpers/application_helper.rb
create  app/views/layouts/application.html.erb
create  app/assets/images/.keep
create  app/mailers/.keep
create  app/models/.keep
create  app/controllers/concerns/.keep
create  app/models/concerns/.keep
create  bin
create  bin/bundle
create  bin/rails
create  bin/rake
create  bin/setup
create  config
create  config/routes.rb
```

It will generate a demo rail project; we will discuss about it later. Currently we have to check if the environment is set up or not. Next, use the following command to run WEBrick web server on your machine.

```
tp> cd demo
tp> rails server
```

It will generate auto-code to start the server

```
$ rails server
DL is deprecated, please use Fiddle
=> Booting WEBrick
=> Rails 4.2.3 application starting in development on http://localhost:3000
=> Run `rails server -h` for more startup options
=> Ctrl-C to shutdown server
_
```

Now open your browser and type the following –

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

It should display a message, something like, "Welcome aboard" or "Congratulations".

