

JUMP START RAILS

BY ANDY HAWTHORNE

Jump Start Rails

by Andy Hawthorne

Copyright © 2013 SitePoint Pty. Ltd.

Product Manager: Simon Mackie

English Editor: Paul Fitzpatrick

Technical Editor: Glenn Goodrich

Cover Designer: Alex Walker

Notice of Rights

All rights reserved. No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical articles or reviews.

Notice of Liability

The author and publisher have made every effort to ensure the accuracy of the information herein. However, the information contained in this book is sold without warranty, either express or implied. Neither the authors and SitePoint Pty. Ltd., nor its dealers or distributors will be held liable for any damages to be caused either directly or indirectly by the instructions contained in this book, or by the software or hardware products described herein.

Trademark Notice

Rather than indicating every occurrence of a trademarked name as such, this book uses the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.



Published by SitePoint Pty. Ltd.

48 Cambridge Street Collingwood

VIC Australia 3066

Web: www.sitepoint.com

Email: business@sitepoint.com

ISBN 978-0-9874674-2-3 (print)

ISBN 978-0-9874674-3-0 (ebook)

Printed and bound in the United States of America

About Andy Hawthorne

Andy is a freelance writer and web developer from Coventry, England. He has spent 12 years as a web developer, and still likes trying new web coding technologies.

About SitePoint

SitePoint specializes in publishing fun, practical, and easy-to-understand content for web professionals. Visit <http://www.sitepoint.com/> to access our blogs, books, newsletters, articles, and community forums. You'll find a stack of information on JavaScript, PHP, Ruby, mobile development, design, and more.

About Jump Start

Jump Start books provide you with a rapid and practical introduction to web development languages and technologies. Typically around 150 pages in length, they can be read in a weekend, giving you a solid grounding in the topic and the confidence to experiment on your own.

*To my wife Mary— without her
never-failing love and support I'd
never get anything finished. And
to my Dad, who inspired me to
write in the first place.*

Table of Contents

| | |
|---|------|
| Preface | xi |
| Who Should Read This Book | xi |
| Conventions Used | xi |
| Code Samples | xi |
| Tips, Notes, and Warnings | xiii |
| Supplementary Materials | xiii |
| Do you want to keep learning? | xiv |
| | |
| Chapter 1 Getting on Rails | 1 |
| Do I need to know Ruby? | 2 |
| What You'll Need | 2 |
| The Rails Stack | 2 |
| Rails and MVC | 3 |
| Installing Rails | 4 |
| Installing Rails on Windows | 4 |
| Installing on GNU Linux | 6 |
| Installing Rails on Mac OS X | 10 |
| Some Other Options | 12 |
| rbenv | 12 |
| Databases | 13 |
| Installing Ruby Gems | 13 |
| A Word about Editors | 14 |
| Summary | 15 |
| | |
| Chapter 2 Starting an App | 17 |
| Data First? | 17 |

| | |
|---|-----------|
| Hello, World | 18 |
| A Simple App | 18 |
| The Project Folders | 21 |
| Generators | 23 |
| Generating a Controller | 24 |
| Some Notes About the Controller | 28 |
| Layouts | 28 |
| Template Data | 29 |
| Setting the Default Page | 30 |
| Creating a Route | 30 |
| Some Linking | 31 |
| Some Styling | 32 |
| Putting up Some Scaffolding | 33 |
| Generating a Scaffold | 33 |
| Active Record Validations | 39 |
| Test Driven Development | 41 |
| The Environments | 43 |
| Tests Rails Has Already Generated | 43 |
| Fixtures | 44 |
| Unit Testing Models | 45 |
| Creating a Layout with Bootstrap | 49 |
| Install the Gem | 49 |
| Connect the Bootstrap Assets | 50 |
| Refining the Layout | 52 |
| Some Notes about the Rails Asset Pipeline | 53 |
| Summary | 54 |
| | |
| Chapter 3 Working with Data | 57 |
| Starting the App | 58 |
| A Little Bit of Planning | 62 |

| | |
|--------------------------------|----|
| Generating a Model | 62 |
| Adding Some Validation | 65 |
| Adding a Controller | 70 |
| Adding Another Resource | 71 |
| Creating an Association | 71 |
| Some Rails Routing Notes | 72 |
| Adding a Test for Routes | 74 |
| Installing ActiveAdmin | 76 |
| Summary | 81 |

Chapter 4 **Creating an ActiveAdmin**

| | |
|---|-----|
| Resource | 83 |
| Creating Admin Functionality Manually | 83 |
| Creating a New Page | 86 |
| Showing Pages | 87 |
| Creating Resources with ActiveAdmin | 89 |
| Back to the Dashboard | 94 |
| Editing the Pages Listing | 95 |
| A New Resource | 97 |
| An Additional Controller | 100 |
| Adding Recent Posts to the Dashboard | 100 |
| Controlling Fields on the Forms | 103 |
| Extra Functionality with Scopes | 106 |
| Summary | 108 |

Chapter 5 **Adding More Features**

| | |
|--------------------------|-----|
| The FriendlyId Gem | 110 |
| Managing Metadata | 113 |
| Uploads | 116 |

| | |
|--|-----|
| Installing the Gem | 117 |
| Using the Generator | 118 |
| Creating an Uploads Folder | 118 |
| Updating the Post Model | 118 |
| Uploading | 119 |
| Displaying Images | 121 |
| Creating Thumbnail Images on the Fly | 122 |
| Uploading to S3 | 125 |
| Creating a Layout with Bootstrap | 128 |
| A Custom Helper | 131 |
| Changing the Posts Layout | 133 |
| Updating the Navigation | 136 |
| Controlling Images in the Layout | 137 |
| Summary | 166 |

Chapter 6 **Deploying to Heroku**

| | |
|---|-----|
| Using Partials to Add a Twitter Feed | 141 |
| A Dash of CoffeeScript | 144 |
| Watch out for Those Turbolinks | 150 |
| Adding the Portfolio (or Other Content) Section | 151 |
| Deployment | 154 |
| Hello, Heroku | 156 |
| Deployment | 158 |
| Working with Data | 160 |
| Adding a Custom Domain Name | 166 |
| Finally... .. | 166 |

Preface

Ruby on Rails was created in 2003 by David Heinemeier Hansson. Since then it has been extended by more than 21,000 contributors.

Rails was always intended to make web development a much slicker process than was previously available with other technologies. It doesn't require thousands of lines of code to get common functionality built into your apps. Rails uses the concept of "convention over configuration", meaning that many of the common tasks we do when developing web applications are covered quickly and easily.

It is true to say that Rails has a steeper learner curve than, say, your average PHP framework. However, the effort to learn it is certainly worth it. I doubt that you will ever fully go back to choosing other technologies over Rails where it makes sense for the app you are building.

The Ruby programming language is a delight to work with, too. It's what Rails is built on, and it offers a powerful set of features for all sorts of programming tasks, not just those for the Web.

This is a short book, designed to give you a "jump start" with Rails. I've based it on my own experiences of building a production Rails app for the first time. Hopefully, like me, you will come to enjoy the slick, efficient web development experience that Rails provides.

Who Should Read This Book

Developers seeking a rapid introduction to Rails. You'll need to know HTML and CSS, and experience with other programming languages would be useful.

Conventions Used

You'll notice that we've used certain typographic and layout styles throughout this book to signify different types of information. Look out for the following items.

Code Samples

Code in this book will be displayed using a fixed-width font, like so:

```
<h1>A Perfect Summer's Day</h1>
<p>It was a lovely day for a walk in the park. The birds
were singing and the kids were all back at school.</p>
```

If the code is to be found in the book's code archive, the name of the file will appear at the top of the program listing, like this:

```
example.css
```

```
.footer {
  background-color: #CCC;
  border-top: 1px solid #333;
}
```

If only part of the file is displayed, this is indicated by the word *excerpt*:

```
example.css (excerpt)
```

```
border-top: 1px solid #333;
```

If additional code is to be inserted into an existing example, the new code will be displayed in bold:

```
function animate() {
  new_variable = "Hello";
}
```

Also, where existing code is required for context, rather than repeat all the code, a `:` will be displayed:

```
function animate() {
  :
  return new_variable;
}
```

Some lines of code are intended to be entered on one line, but we've had to wrap them because of page constraints. A `↵` indicates a line break that exists for formatting purposes only, and should be ignored.

```
URL.open("http://www.sitepoint.com/responsive-web-design-real-user-  
testing/?responsive1");
```

Tips, Notes, and Warnings



Hey, You!

Tips will give you helpful little pointers.



Ahem, Excuse Me ...

Notes are useful asides that are related—but not critical—to the topic at hand. Think of them as extra tidbits of information.



Make Sure You Always ...

... pay attention to these important points.



Watch Out!

Warnings will highlight any gotchas that are likely to trip you up along the way.

Supplementary Materials

<http://www.sitepoint.com/books/jsrails1/>

The book's website, containing links, updates, resources, and more.

<http://www.sitepoint.com/books/jsrails1/code.php>

The downloadable code archive for this book.

<http://www.sitepoint.com/forums/forumdisplay.php?227-Ruby-amp-Rails>

SitePoint's forums, for help on any tricky web problems.

books@sitepoint.com

Our email address, should you need to contact us for support, to report a problem, or for any other reason.

Do you want to keep learning?

You can now get unlimited access to courses and ALL SitePoint books at Learnable for one low price. Enroll now and start learning today! Join Learnable and you'll stay ahead of the newest technology trends: <http://www.learnable.com>.

Once you've mastered the principles of Rails, challenge yourself with our online quiz. Can you achieve a perfect score? Head on over to <http://quizpoint.com/#categories/RUBY>.

Chapter 1

Getting on Rails

Welcome to Jump Start Rails! If you've come to Rails from another server-side coding technology such as PHP, you are in for a treat. Rails offers a slick and efficient coding experience for web developers, and was created with built-in solutions to many of the common web development headaches.

Rails is open source and free to use, which means that you don't need to spend a lot to get developing with it. In fact, the biggest outlay will probably be purchasing an editor. We'll look at the options a little later in this chapter.

Rails was created in 2003 by David Heinemeier Hansson of 37 Signals¹ fame. Since then, it has seen rapid development by the Rails core team², with over 2,000 contributors. Rails runs on the Ruby³ general purpose programming language, created by Yukihiro “Matz” Matsumoto, in 1995.

¹ <http://37signals.com/>

² <http://rubyonrails.org/core>

³ <http://www.ruby-lang.org/en/>

Do I need to know Ruby?

You can certainly build simple Rails apps with a *limited* knowledge of Ruby. Many developers tend to learn Ruby as they learn Rails. And as your knowledge and confidence with Rails increases, you'll want to do more with it.

I've found that you can do this in incremental steps; it's entirely possible to build a Rails app while you are still learning Ruby. The good news is that learning Rails is a great experience, but learning Ruby is equally rewarding — especially if you have come from another language like PHP, for example. Ruby is described on the Ruby website⁴ as "a programmer's best friend" for a reason. So to really get into Rails a good knowledge of Ruby will be required — eventually.

Ruby seems to be built for learning on the go—whenever you come across an obstacle, the answer is never far away. Ruby Docs⁵ will help enormously with this.

What You'll Need

Ruby on Rails⁶, like many other web coding technologies, requires some setting up on your system first. It's not too scary, though, and since this book covers Rails 4.0, we only have to be concerned with setting up to use the latest versions.

Rails isn't Ruby. It's built *using* Ruby, and you use Ruby to build Rails applications. As such, it needs to be present on your system for Rails to run. Happily Ruby is available to run pretty much everywhere.

The Rails Stack

There are several components that make up the Rails stack. Obviously Ruby is one component, the other main one being a database of some kind.

During the process of guiding you through installing Rails in this chapter, I'll mention PostgreSQL⁷ and Ruby Version Manager (RVM)⁸. Technically neither are *essential* requirements; it's just that they are common tools used in creating a Rails stack.

⁴ <http://www.ruby-lang.org/en/>

⁵ <http://ruby-doc.org/core-1.9.3/>

⁶ <http://rubyonrails.org/>

⁷ <http://www.postgresql.org/>

⁸ <http://rvm.io>

RVM is a sandboxed way to install numerous versions of Ruby on your system, all without affecting any system configuration files. It's available for Unix-based systems, and as part of an installer⁹ for Windows.

If you create a Rails project without specifying a database, one will be created anyway. It'll be a SQLite database¹⁰, and will serve very well for your initial Rails investigation. However, in Chapter 6, we'll be deploying to Heroku¹¹, and that requires a PostgreSQL database¹². As such, we'll be making PostgreSQL part of our Rails stack too.

Rails and MVC

The Rails framework is based on the Model View Controller (MVC) design pattern. No doubt you'll have heard of it if you've already spent time around web development. The truth is that, with Rails, there are real advantages to be had from MVC.

A few of these advantages are:

- the ability to keep application logic (or business logic, if you prefer) separate from the user interface
- Don't Repeat Yourself (DRY) capability. The term DRY also applies in all forms of web and software development. It's a concept where the objective is to only write one piece of code to perform a particular task. You'll see this in action as we begin to build our main app
- a clear pattern for where each type of code should be stored within the application

Rails uses MVC like this:

- **Models** are used mostly for setting the rules for interaction with database tables. Normally, you would have one model per database table.

⁹ <http://railsinstaller.org/>

¹⁰ <http://www.sqlite.org/>

¹¹ <http://www.heroku.com/>

¹² <http://www.postgresql.org/>

- **Views** are HTML files with Ruby embedded to perform tasks for the presentation of data. Views are the user interface—the part of your app with which the user interacts.
- **Controllers** are the components that decide how to respond to user requests. They are responsible for coordinating responses too. You can think of them as traffic police directing requests and responses around the application. It's important to understand that controllers are the only components that can speak to models and views, as well as to our user's browser.

Installing Rails

Let's run through the basic process of getting Rails installed on Windows, Mac, and Linux.

Installing Rails on Windows

I'm going to stick my neck out here: If you intend to work seriously with Rails, then you might want to consider switching to a Unix-based operating system. The reason is a practical one: You'll be spending a lot of time on the command line with Rails. You will also need to keep your Ruby gems up to date. This is all done via the command line. The fact is, it's far easier to manage this stuff on a Unix system such as Linux or Mac OS X. You could always run a Virtual Machine for your Rails-coding projects, and I'll explain how to do that later in this chapter.

There are options for Windows users, and it's worth mentioning that huge efforts are being made to make Ruby easier to work with on Windows.

If you are running Windows, there is now an easy solution for getting going with Rails. The RailsInstaller¹³ has been provided by the team at Engine Yard, and it takes the pain out of configuring Ruby and Rails manually. Simply download the installer, run it, and away you go.

The installer includes all of the required gems and dependencies so that you can start using Rails immediately. It even includes Git, the version control system, widely used in the Ruby/Rails community.

¹³ <http://railsinstaller.org/>

PostgreSQL Database

You can download the required software from the PostgreSQL¹⁴. It includes the excellent pgAdmin tool—a graphical user interface for the database. You just need to download and install the software and leave it at that. There is very little else to do, as you will work with the database mostly via Rails.

When we get to deployment, I'll explain how you can import and export data between your local PostgreSQL installation and the server running at Heroku.

Another Option for Windows Users

There is another way of setting up for Rails development on Windows—a virtual machine (VM). It's a more involved process, but if you want to try Linux, this is one way to do it.

For example, there is an excellent open source package available for Windows, called VirtualBox¹⁵. VirtualBox provides you with the required base in which you're free to create as many virtual machines as your system can handle. The idea is that you install whichever flavor of Linux takes your fancy, install Ruby/Rails and the associated requirements, and do your development in the VM, rather than on your base system.

The advantage of this approach is that you don't have to install any of the required components on your base system so, should anything go awry, you can simply delete the VM and start again. Your base system is not affected in any way.

So, how do we do it? First, you'll have some downloading to do, and it certainly helps if you have installed a Linux distribution (distro for short) before. Here are the steps:

1. Download the ISO for your chosen Linux distro. I always store mine in a folder called **iso** in my **Home** directory—it makes it easier to find when you create your VM.
2. Download VirtualBox, and install it. This should prove to be just a normal installation, like any other software.

¹⁴ <http://www.postgresql.org/download/windows/>

¹⁵ <https://www.virtualbox.org/wiki/Downloads>