

Put your site development on rails with Ruby!

Ruby on Rails™

FOR DUMMIES®

Use Ruby to
develop Web apps,
blogs, and more

**A Reference
for the
Rest of Us!®**

FREE eTips at dummies.com®

Barry Burd

Author of Java™ 2 For Dummies



Put your site development on rails with Ruby!

Ruby on Rails™

FOR DUMMIES®

*Use Ruby to
develop Web apps,
blogs, and more*

*A Reference
for the
Rest of Us!®*

FREE eTips at dummies.com®

Barry Burd
Author of Java™ 2 For Dummies



***Ruby on Rails™ For
Dummies®***

by Barry Burd



WILEY

Wiley Publishing, Inc.

Ruby on Rails™ For Dummies®

Published by

Wiley Publishing, Inc.

111 River St.

Hoboken, NJ 07030-5774

www.wiley.com

Copyright © 2007 by Wiley Publishing, Inc.,
Indianapolis, Indiana

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Legal Department, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, (317) 572-3447, fax (317)

572-4355, or online at

<http://www.wiley.com/go/permissions>.

Trademarks: Wiley, the Wiley Publishing logo, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies Daily, The Fun and Easy Way, Dummies.com, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. Ruby on Rails is a trademark of David Heinemeier Hansson. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The

fact that an organization or Website is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Website may provide or recommendations it may make. Further, readers should be aware that Internet Websites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 800-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

For technical support, please visit www.wiley.com/techsupport.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Control Number: 2006936826

ISBN: 978-0-470-08120-4

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

10/RZ/RS/QW/IN



About the Author

Dr. Barry Burd received an M.S. degree in Computer Science at Rutgers University and a Ph.D. in Mathematics at the University of Illinois. As a teaching assistant in Champaign-Urbana, Illinois, he was elected to the university-wide “List of Teachers Ranked as Excellent by Their Students” five times.

Since 1980, Dr. Burd has been a professor in the Department of Mathematics and Computer Science at Drew University in Madison, New Jersey. When he’s not lecturing at Drew University, Dr. Burd leads training courses for professional programmers in business and industry. He has lectured at conferences in the United States, Europe, Australia, and Asia. He is the author of several articles and books, including *Java For Dummies*, 4th Edition, and *JSP: JavaServer Pages*, both from Wiley Publishing, Inc.

Dr. Burd lives in Madison, New Jersey, with his wife and two children. In his spare time, he enjoys being a workaholic.

Dedication

for Harriet, Sam and Jennie, Sam and Ruth, Abram and
Katie, Benjamin and Jennie

Author's Acknowledgments

Many thanks to Paul Levesque who worked so closely with me on this project, and thanks to Katie Feltman who headed up the project at Wiley. And to Andy Cummings who steers the *For Dummies* series, thanks. And, yes, thanks to copy editors Mary Lagu and Virginia Sanders. Also, thanks to Laura Lewin, agent at StudioB. Thanks, and thanks again to Jay Zimmerman and the speakers in the No Fluff, Just Stuff Symposium for opening my eyes to Ruby on Rails. And to Charles Nutter and Thomas Enebo, who bridge the gap between Ruby and Java, thanks. Of course, Matt Kent, Kyle Shank, and Marc Baumbach, thanks for the use of RadRails, both inside and outside of this book. I extend thanks to Stefan Reichert with his Wicked Shell. To Francis Hwang and the members of the Ruby-NYC group, I say thanks. Thanks indeed to Frank Greco and his New York Java Special Interest Group and to Mike Redlich and the gang at the Amateur Computer Group of New Jersey because without them I wouldn't know anything about object-relational mapping. Thanks. And special thanks to Sam and Jennie, and of course, to Harriet, thanks I say thanks I will Thanks.

Publisher's Acknowledgments

We're proud of this book; please send us your comments through our online registration form located at www.dummies.com/register/.

Some of the people who helped bring this book to market include the following:

Acquisitions, Editorial, and Media Development

Senior Project Editor: Paul Levesque

Acquisitions Editor: Katie Feltman

Copy Editors: Mary Lagu, Virginia Sanders

Technical Editor: Charles Nutter

Editorial Manager: Leah Cameron

Media Development Specialists: Angela Denny, Kate Jenkins, Steven Kudirka, Kit Malone

Media Development Coordinator: Laura Atkinson

Media Project Supervisor: Laura Moss

Media Development Manager: Laura VanWinkle

Editorial Assistant: Amanda Foxworth

Sr. Editorial Assistant: Cherie Case

Cartoons: Rich Tennant (www.the5thwave.com)

Composition Services

Project Coordinator: Erin Smith

Layout and Graphics: Claudia Bell, Lavonne Cook, Denny Hager, Barbara Moore, Barry Offringa, Laura Pence, Heather Ryan

Proofreaders: Cynthia Fields, Jessica Kramer, Techbooks

Indexer: Techbooks

Anniversary Logo Design: Richard Pacifico

Publishing and Editorial for Technology Dummies

Richard Swadley, Vice President and Executive Group Publisher

Andy Cummings, Vice President and Publisher

Mary Bednarek, Executive Acquisitions Director

Mary C. Corder, Editorial Director

Publishing for Consumer Dummies

Diane Graves Steele, Vice President and Publisher

Joyce Pepple, Acquisitions Director

Composition Services

Gerry Fahey, Vice President of Production Services

Debbie Stailey, Director of Composition Services

Contents

[Title](#)

[Introduction](#)

[How to Use This Book](#)

[Conventions Used in This Book](#)

[What You Don't Have to Read](#)

[Foolish Assumptions](#)

[How This Book Is Organized](#)

[Icons Used in This Book](#)

[Where to Go from Here](#)

[Part I : Nuts and Bolts](#)

[Chapter 1: Welcome to the World of Ruby on Rails](#)

[The Software Development Process](#)

[Along Comes Ruby on Rails](#)

[Let's Get Going](#)

[Chapter 2: Installing the Software](#)

[Six Pieces of Software](#)

[Installing the Ruby Interpreter](#)

[Installing Rails](#)

[Installing Java](#)

[Installing RadRails](#)

[Installing MySQL](#)

[Installing MySQL Administrator](#)

[Chapter 3: Details on Rails](#)

[Creating a Database](#)

[Creating a New Ruby on Rails Project](#)

[Running Your New Rails Project \(Already!\)](#)

[Creating a Model](#)

[Creating a Database Table](#)

[Creating a Scaffold](#)

[Using the New Web Interface](#)

[Chapter 4: Using RadRails](#)

[Words, Words, Words](#)

[Some Common RadRails Tasks](#)

[Troubleshooting the Run of a Ruby Program](#)

[Part II : Creating Code](#)

[Chapter 5: Ruby One's Day](#)

[Hello, Again](#)

[Working with Values](#)

[Going with the Flow](#)

[Bunches of Things](#)

[Using Methods](#)

[Chapter 6: Ruby Two's Day](#)

[Objects and Classes](#)

[Objects Have Methods](#)

[Enhancing Classes](#)

[Creating a Module](#)

[Chapter 7: Weaving the Web](#)

[The Working of the Web](#)

[Your HTML Starter Kit](#)

[HTML Elements](#)

[Part III : Real Rails](#)

[Chapter 8: Action-Packed Adventures](#)

[Model/View/Controller](#)

[The Rails Way of Life](#)

[Writing What You Want Where You Want It](#)

[The Controller Shakes Hands with the View](#)

[Dividing the Work of the View](#)

[Chapter 9: Some Things You Can Do with Models](#)

[A Web Site for Photos](#)

[Programming with a Rails Model](#)

[Modifying a Database](#)

[More Rails Programming Tricks](#)

[Chapter 10: I've Been Working on the Rails Code](#)

[Displaying an Image](#)

[Importing Files](#)

[Chapter 11: Image Is Everything](#)

[Enhancing Your Project's Code](#)

[Understanding the Enhanced Code](#)

[Chapter 12: More Model Magic](#)

[Blogging Your Dreams](#)

[Validating the Visitor's Input](#)

[Adding Comments](#)

[Adding Keywords](#)

[Chapter 13: Cool Things on Rails](#)

[Using Ajax](#)

[Sending E-Mail](#)

[Creating and Consuming Web Services](#)

[Part IV : The Part of Tens](#)

[Chapter 14: Ten \(Times Two\) Great Web Sites](#)

[Ten Ruby Sites](#)

[Ten Rails Sites](#)

[Chapter 15: Ten Features That Set Ruby Apart](#)

[Hashes](#)

[Open Classes](#)

[Duck Typing](#)

[Modifiers](#)

[Blocks](#)

[Everything Is an Object](#)

[Objects Might Have Their Own Methods](#)

[Mixins](#)

[Built-In Unit Testing](#)

[Built-In Reflection](#)

[Chapter 16: Ten Pivotal Ruby on Rails Concepts](#)

[Don't Repeat Yourself \(DRY\)](#)

[Convention over Configuration](#)

[Model/View/Controller \(MVC\)](#)

[Agile Development](#)

[Dynamic Discovery of a Database's Characteristics](#)

[Object-Relational Mapping \(ORM\)](#)

[Using Generators](#)

[Create, Read, Update, and Delete \(CRUD\)](#)

[Using Migrations](#)

[Using Partial](#)

[Chapter 17: Ten Ways to Override Rails Defaults](#)

[Overriding the Database Name](#)

[Overriding a Database Table Name](#)

[Overriding a Controller Name](#)

[Overriding the Name of a Table's Primary Key](#)

[Using Singular Nouns](#)

[Creating Irregular Plurals](#)

[Overriding a Default Layout](#)

[Creating Additional Web Pages](#)

[Modifying the Meanings of URLs](#)

[Changing the Server Environment](#)

[: Further Reading](#)

Introduction

“Ruby on Rails? What’s that?” asks my uncle. “You write about this stuff for dummies? You mean those black and yellow books that everyone buys?”

“Yes, like the one I’m quoting you in,” I say. “Please check your spelling as you speak.”

“I will. But what’s Ruby on Rails? Is it the 6:05 train to Poughkeepsie? Is it the name of an old vaudeville act? Is it a pop singer? A rock band? Is it a rare stone from India? Is it the codename of an informer in a political scandal?”

“No.”

“Is it the name of an exotic cocktail? A species of bird? An animal act in a circus? A John D. MacDonald title?”

Finally, I interrupt. “Ruby on Rails is a computer thing.”

“What kind of computer thing?” he asks.

“It’s a framework for creating applications with Web interfaces to databases.”

“Oh, yeah?” he says. “Your nephew from Brooklyn, he read *Getting Ahead in Politics For Dummies*. He loved the book. Did you write that one?”

How to Use This Book

As a computer book author, I strive not to be full of myself. I have no illusions that you plan on reading this book from cover to cover. I read sections and chapters out of order when I buy a computer book. Why would I expect you to approach my book any differently? And even if I read something in Chapter 2, who says I remember it when I read Chapter 11?

I write each section with these thoughts in mind. In the middle of Chapter 12, I might want you to remember some nugget of knowledge that I introduce in Chapter 4. If I use that nugget over and over again in Chapters 5, 7, 8, and 9, I don't remind you about it in Chapter 12. But for other nuggets — ones that you don't read about repeatedly in this book — I provide cross references.

So in general, my advice is

- ✓ Read what interests you; skip what doesn't interest you.
- ✓ If you already know something, don't bother reading about it.
- ✓ If you're curious, don't be afraid to skip ahead. You can always sneak a peek at an earlier chapter if you really need to do so.

Conventions Used in This Book

Almost every technical book starts with a little typeface legend, and *Ruby on Rails For Dummies* is no exception. What follows is a brief explanation of the typefaces used in this book:

- ✓ New terms are set in *italics*.
- ✓ If you need to type something that's mixed in with the regular text, the characters you type appear in bold. For example: "Type **MyNewProject** in the text field."
- ✓ You also see this computerese font. I use computerese for Ruby code, filenames, Web page addresses (URLs), on-screen messages, and other such things. Also, if something you need to type is really long, it appears in computerese font on its own line (or lines).
- ✓ You need to change certain things when you type them. Words that you need to replace with your own words are set in italicized computerese. For instance, I might ask you to type

```
class Anyname
```

which means that you type **class** and then some name that you make up on your own.

What You Don't Have to Read

Pick the first chapter or section that has material you don't already know and start reading there. Of course, you might hate making decisions as much as I do. If so, here are some guidelines that you can follow:

- ✓ If you already know what kind of an animal Ruby on Rails is and you know that you want to use Ruby on Rails, skip Chapter 1 and go straight to Chapter 2. Believe me, I won't mind.
- ✓ If you already have Ruby on Rails, a database, and a Ruby program editor installed on your computer, skip Chapter 2 and go to Chapter 3.
- ✓ If you've seen one of the many Ruby on Rails demos or worked through a Ruby on Rails tutorial, move quickly through Chapter 3.

However, don't completely ignore Chapter 3. Some of the wording I use in Chapter 3 might be helpful, even if you've already been through a Rails demo or two.

- ✓ If you're a computer programmer, you might have already used Eclipse (for Java or for some other programming language). In that case, plan a quick excursion through Chapter 4. This book's examples use the RadRails integrated development environment, and RadRails is based on Eclipse.
- ✓ If you've never written programs in Ruby, Perl, or Smalltalk, set aside some time to read Chapters 5 and 6.

These chapters cover some Ruby concepts, but the chapters don't describe the entire Ruby language. In these chapters, I highlight Ruby concepts that appear frequently in Rails code. I also emphasize some unusual features of Ruby — features that you don't find in other language families (in Java and C++, for example).

If you want to skip the sidebars and the Technical Stuff icons, please do. But try not to skip too many of my jokes. (I tell my kids that I write jokes for a living. They don't believe me. But even so, I'd appreciate your help in perpetuating this myth.)

Foolish Assumptions

In this book, I make a few assumptions about you, the reader. If one of these assumptions is incorrect, you're probably okay. If all these assumptions are incorrect . . . well, buy the book anyway.

✓ **I assume that you have access to a computer.**
Here's the good news: You can run the code in this book on almost any computer. The only computers that you can't use to run this code are ancient things that are more than six years old (give or take a few years).

✓ **I assume that you can navigate through your computer's common menus and dialog boxes.**
You don't have to be a Windows, Linux, or Macintosh power user, but you should be able to start a

program, find a file, put a file into a certain directory . . . that sort of thing.

On those rare occasions when you need to drag and drop, cut and paste, or plug and play, I guide you carefully through the steps. But your computer might be configured in any of several billion ways, and my instructions might not quite fit your special situation. So, when you reach one of these platform-specific tasks, try following the steps in this book. If the steps don't quite fit, consult a book with instructions tailored to your system or visit one of this book's Web sites for helpful hints. The URLs are www.burdbrain.com/RubyOnRails and www.dummies.com/go/RonR1e.

✓ **I assume that you've written some computer programs.** I've tried to make the book interesting for experienced programmers, yet accessible to people with only a modest amount of programming experience. I don't assume that you've written programs in any particular language or that you've hacked from midnight until dawn on the latest UNIX system. I assume only that you can compose loops, if statements, and other such things. (Of course, if you have no computer programming experience, you can start with my *Beginning Programming with Java For Dummies* book. Remember, the more of my books that you buy, the less debt I'll have when my kids finish college.)

If you've written lots of programs in Visual Basic, Java, or C++, you'll discover some interesting plot twists in Ruby. The developer of Ruby took the best ideas in other programming languages, streamlined them, combined them, and reorganized them into a flexible, powerful new programming language. Ruby

has many new, thought-provoking features. As you find out about these features, many of them will seem very natural to you. One way or another, you'll feel good about using Ruby.

How This Book Is Organized

This book is divided into subsections, which are grouped into sections, which come together to make chapters, which are lumped finally into four parts. (When you write a book, you get to know your book's structure pretty well. After months of writing, you find yourself dreaming in sections and chapters when you go to bed at night.) The parts of the book are listed here.

Part I: Nuts and Bolts

This part is your executive briefing. It includes a chapter that answers the question “What is Ruby on Rails?” and a chapter with a complete set of instructions on installing and running the software. It also has a jump-start chapter and a chapter with details about the RadRails integrated development environment.

Part II: Creating Code

Chapters 5 through 7 cover Ruby and HTML. Some of the material in Part II might be familiar to you. If so, you can skip some sections or read this stuff quickly. But don't read too quickly. Ruby is a little different from some other programming languages, and you might stumble upon some exciting new ideas.

Part III: Real Rails

This third part cuts to the chase. Rails has three components — Action Controller, Action View, and Active Record. The controller controls things (of course), the view displays things, and Active Record maintains all the data. Chapters 8 through 13 cover these three components and describe some interesting applications along the way.