

Addison-Wesley Professional Ruby Series



THE RAILS™ 5 WAY



OBIE FERNANDEZ

Foreword by GILES BOWKETT

Praise for *The Rails Way*

For intermediates and above, I strongly recommend adding this title to your technical bookshelf. There is simply no other Rails title on the market at this time that offers the technical depth of the framework than *The Rails™ 3 Way*.

—Mike Riley, *Dr. Dobbs's Journal*

I highly suggest you get this book. Software moves fast, especially the Rails API, but I feel this book has many core API and development concepts that will be useful for a while to come.

—Matt Polito, software engineer and member of Chicago Ruby User Group

This book should live on your desktop if you're a Rails developer. It's nearly perfect in my opinion.

—Luca Pette, developer

The Rails™ 3 Way is likely to take you from being a haphazard poke-a-stick-at-it programmer to a deliberate, skillful, productive, and confident RoR developer.

—Katrina Owen, JavaRanch

I can positively say that it's the single best Rails book ever published to date. By a long shot.

—Antonio Cangiano, software engineer and technical evangelist at IBM

The Rails 5 Way

Table of Contents

Cover

Title Page

Copyright Page

Contents

Foreword by Giles Bowkett

Foreword to the Previous Edition by Steve Klabnik

Foreword to the Previous Edition by David Heinemeier
Hansson

Foreword to the Previous Edition by Yehuda Katz

Introduction

Acknowledgments

About the Author

Chapter 1 Rails Configuration and Environments

1.1 Bundler

1.1.1 Gemfile

1.1.2 Installing Gems

1.1.3 Gem Locking

1.1.4 Packaging Gems

1.1.5 Binstubs

1.2 Startup Scripts

1.2.1 config/environment.rb

1.2.2 config/boot.rb

Table of Contents

1.2.3 config/application.rb

1.3 Default Initializers

1.3.1 config/initializers/application_controller_renderer.rb

1.3.2 config/initializers/assets.rb

1.3.3 config/initializers/backtrace_silencers.rb

1.3.4 config/initializers/cookies_serializer.rb

1.3.5 config/initializers/filter_parameter_logging.rb

1.3.6 config/initializers/inflections.rb

1.3.7 config/initializers/mime_types.rb

1.3.8 config/initializers/new_framework_defaults.rb

1.3.9 config/initializers/session_store.rb

1.3.10 config/initializers/wrap_parameters.rb

1.4 Other Common Initializers

1.4.1 Time Zones

1.4.2 Localization

1.4.3 Generator Default Settings

1.4.4 Load Path Modifications

1.4.5 Log-Level Override

1.4.6 Schema Dumper

1.4.7 Console

1.5 Spring Application Preloader

1.6 Development Mode

1.6.1 Automatic Class Reloading

1.6.2 Eager Load

1.6.3 Error Reports

1.6.4 Caching

1.6.5 Action Mailer Settings

1.6.6 Deprecation Notices

1.6.7 Pending Migrations Error Page

Table of Contents

1.6.8 Assets Debug Mode

1.6.9 Missing Translations

1.7 Test Mode

1.8 Production Mode

1.8.1 Assets

1.8.2 Asset Hosts

1.9 Configuring a Database

1.10 Configuring Application Secrets

1.11 Logging

1.11.1 Rails Log Files

1.11.2 Tagged Logging

1.11.3 Log File Analysis

1.11.4 Rails::Subscriber.colorize_logging

1.12 Conclusion

Chapter 2 Routing

2.1 The Two Purposes of Routing

2.2 The routes.rb File

2.2.1 Regular Routes

2.2.2 Constraining Request Methods

2.2.3 URL Patterns

2.2.4 Segment Keys

2.2.5 Spotlight on the :id Field

2.2.6 Optional Segment Keys

2.2.7 Defining Defaults

2.2.8 Redirect Routes

2.2.9 The Format Segment

2.2.10 Routes as Rack Endpoints

2.2.11 Accept Header

Table of Contents

2.2.12 Segment Key Constraints

2.2.13 The Root Route

2.3 Route Globbing

2.4 Named Routes

2.4.1 Creating a Named Route

2.4.2 name_path versus name_url

2.4.3 What to Name Your Routes

2.4.4 Argument Sugar

2.4.5 A Little More Sugar with Your Sugar?

2.5 Scoping Routing Rules

2.5.1 Controller

2.5.2 Path Prefix

2.5.3 Name Prefix

2.5.4 Namespaces

2.5.5 Bundling Constraints

2.5.6 Direct Routes

2.6 Listing Routes

2.7 Conclusion

Chapter 3 REST, Resources, and Rails

3.1 REST in a Rather Small Nutshell

3.2 Resources and Representations

3.3 REST in Rails

3.4 Routing and CRUD

3.4.1 REST Resources and Rails

3.4.2 From Named Routes to REST Support

3.4.3 Reenter the HTTP Verb

3.5 The Standard RESTful Controller Actions

3.5.1 PATCH versus PUT

Table of Contents

- 3.5.2 Singular and Plural RESTful Routes
- 3.5.3 The Special Pairs: new/create and edit/update
- 3.5.4 The PATCH and DELETE Cheat
- 3.5.5 Limiting Routes Generated
- 3.6 Singular Resource Routes
- 3.7 Nested Resources
 - 3.7.1 RESTful Controller Mappings
 - 3.7.2 Considerations
 - 3.7.3 Deep Nesting?
 - 3.7.4 Shallow Routes
- 3.8 Routing Concerns
- 3.9 RESTful Route Customizations
 - 3.9.1 Extra Member Routes
 - 3.9.2 Extra Collection Routes
 - 3.9.3 Custom Action Names
 - 3.9.4 Mapping to a Different Controller
 - 3.9.5 Routes for New Resources
 - 3.9.6 Considerations for Extra Routes
- 3.10 Controller-Only Resources
- 3.11 Different Representations of Resources
 - 3.11.1 The respond_to Method
 - 3.11.2 Responders Gem
 - 3.11.3 Formatted Named Routes
- 3.12 The RESTful Rails Action Set
 - 3.12.1 Index
 - 3.12.2 Show
 - 3.12.3 Destroy
 - 3.12.4 new and create

Table of Contents

3.12.5 edit and update

3.13 Conclusion

Chapter 4 Working with Controllers

4.1 Rack

4.1.1 Configuring Your Middleware Stack

4.2 Action Dispatch: Where It All Begins

4.2.1 Request Handling

4.2.2 Getting Intimate with the Dispatcher

4.3 Render unto View . . .

4.3.1 When in Doubt, Render

4.3.2 Explicit Rendering

4.3.3 Rendering Another Actions Template

4.3.4 Rendering a Different Template Altogether

4.3.5 Rendering a Partial Template

4.3.6 Rendering HTML

4.3.7 Rendering Inline Template Code

4.3.8 Accessing Helpers in the Controller

4.3.9 Rendering JavaScript

4.3.10 Rendering Text

4.3.11 Rendering Raw Body Output

4.3.12 Rendering Other Types of Structured Data

4.3.13 Default Rendering Policies

4.3.14 Rendering Nothing

4.3.15 Rendering Options

4.4 Additional Layout Options

4.5 Redirecting

4.5.1 The `redirect_to` Method

4.5.2 The `redirect_back` Method

Table of Contents

4.6 Controller/View Communication

4.7 Action Callbacks

- 4.7.1 Action Callback Inheritance
- 4.7.2 Action Callback Types
- 4.7.3 Action Callback Chain Ordering
- 4.7.4 Around Action Callbacks
- 4.7.5 Action Callback Chain Skipping
- 4.7.6 Action Callback Conditions
- 4.7.7 Action Callback Chain Halting

4.8 Streaming

- 4.8.1 ActionController::Live
- 4.8.2 Support for EventSource
- 4.8.3 Streaming Templates
- 4.8.4 Streaming Buffers and Files

4.9 Variants

4.10 Conclusion

Chapter 5 Working with Active Record

5.1 The Basics

5.2 Macro-Style Methods

- 5.2.1 Relationship Declarations
- 5.2.2 Convention over Configuration
- 5.2.3 Setting Names Manually
- 5.2.4 Legacy Naming Schemes

5.3 Defining Attributes

- 5.3.1 Default Attribute Values

5.4 CRUD: Creating, Reading, Updating, Deleting

- 5.4.1 Creating New Active Record Model Instances
- 5.4.2 Reading Active Record Objects

Table of Contents

5.4.3 Reading and Writing Attributes

5.4.4 Accessing and Manipulating Attributes Before They Are Typecast

5.4.5 Reloading

5.4.6 Cloning

5.4.7 The Query Cache

5.4.8 Updating

5.4.9 Updating by Condition

5.4.10 Updating a Particular Instance

5.4.11 Updating Specific Attributes

5.4.12 Saving without Updating Timestamp

5.4.13 Convenience Updaters

5.4.14 Touching Records

5.4.15 Readonly Attributes

5.4.16 Deleting and Destroying

5.5 Database Locking

5.5.1 Optimistic Locking

5.5.2 Pessimistic Locking

5.5.3 Considerations

5.6 Querying

5.6.1 where(*conditions)

5.6.2 order(*clauses)

5.6.3 take(number) and skip(number)

5.6.4 select(*clauses)

5.6.5 from(*tables)

5.6.6 group(*args)

5.6.7 distinct

5.6.8 having(*clauses)

5.6.9 includes(*associations)

5.6.10 eager_load(*associations)

Table of Contents

- 5.6.11 preloads(*associations)
- 5.6.12 references(*table_names)
- 5.6.13 joins(expression)
- 5.6.14 left_outer_join
- 5.6.15 find_or_create_by(attributes, &block)
- 5.6.16 find_or_initialize_by(attributes, &block)
- 5.6.17 new(attributes, &block)
- 5.6.18 create_with
- 5.6.19 reload
- 5.6.20 reset
- 5.6.21 explain
- 5.6.22 extending(*modules, &block)
- 5.6.23 exists?
- 5.6.24 any?
- 5.6.25 empty?
- 5.6.26 many?
- 5.6.27 one?
- 5.6.28 none
- 5.6.29 lock
- 5.6.30 readonly
- 5.6.31 reorder
- 5.6.32 reverse_order
- 5.6.33 rewhere(conditions)
- 5.6.34 scoping(&block)
- 5.6.35 unscope(*args)
- 5.6.36 merge(other)
- 5.6.37 only(*onlies)
- 5.6.38 except(*skips)
- 5.6.39 or(other)

Table of Contents

5.6.40 load

5.6.41 to_a

5.6.42 to_sql

5.6.43 to_json and friends

5.6.44 arel_table

5.6.45 cache_key

5.7 Ignoring Columns

5.8 Connections to Multiple Databases in Different Models

5.9 Using the Database Connection Directly

5.9.1 The DatabaseStatements Module

5.9.2 Other Connection Methods

5.10 Custom SQL Queries

5.11 Other Configuration Options

5.12 Conclusion

Chapter 6 Active Record Migrations

6.1 Creating Migrations

6.1.1 Generator Magic

6.1.2 Sequencing

6.1.3 The change Method

6.1.4 Rolling Back

6.1.5 Redo

6.1.6 Reversible Operations

6.1.7 Irreversible Operations

6.1.8 create_table(name, options, &block)

6.1.9 change_table(table_name, &block)

6.1.10 create_join_table(*table_names)

6.1.11 API Reference

6.2 Defining Columns

Table of Contents

- 6.2.1 Column Type Mappings
- 6.2.2 Column Options
- 6.2.3 Column Type Gotchas
- 6.2.4 Magic Timestamp Columns
- 6.2.5 More Command-Line Magic

6.3 Transactions

6.4 Data Migration

- 6.4.1 Using SQL
- 6.4.2 Migration Models
- 6.4.3 Database Adapter Helper Methods

6.5 Database Schema

6.6 Database Seeding

6.7 Database-Related Tasks

- 6.7.1 db:create and db:create:all
- 6.7.2 db:drop and db:drop:all
- 6.7.3 db:forward and db:rollback
- 6.7.4 db:migrate
- 6.7.5 db:migrate:down
- 6.7.6 db:migrate:up
- 6.7.7 db:migrate:redo
- 6.7.8 db:migrate:reset
- 6.7.9 db:migrate:status
- 6.7.10 db:reset and db:setup
- 6.7.11 db:schema:dump
- 6.7.12 db:schema:load
- 6.7.13 db:seed
- 6.7.14 db:structure:dump
- 6.7.15 db:test:prepare
- 6.7.16 db:version

Table of Contents

6.8 Conclusion

Chapter 7 Active Record Associations

7.1 The Association Hierarchy

7.2 One-to-Many Relationships

7.2.1 Adding Associated Objects to a Collection

7.3 Belongs to Associations

7.3.1 Methods

7.3.2 Options

7.3.3 Scopes

7.4 Has Many Associations

7.4.1 Methods

7.4.2 Options

7.4.3 Scoping

7.5 Many-to-Many Relationships

7.5.1 has_and_belongs_to_many

7.5.2 has_many :through

7.5.3 Options

7.5.4 Unique Association Objects

7.6 One-to-One Relationships

7.6.1 has_one

7.6.2 Using has_one together with has_many

7.6.3 Options

7.6.4 Scopes

7.7 Working with Unsaved Objects and Associations

7.7.1 One-to-One Associations

7.7.2 Collections

7.7.3 Deletion

7.8 Association Extensions

Table of Contents

7.9 The CollectionProxy Class

7.9.1 Owner, Reflection, and Target

7.9.2 reload and reset

7.10 Conclusion

Chapter 8 Validations

8.1 Finding Errors

8.2 The Simple Declarative Validations

8.2.1 validates_absence_of

8.2.2 validates_acceptance_of

8.2.3 validates_associated

8.2.4 validates_confirmation_of

8.2.5 validates_each

8.2.6 validates_format_of

8.2.7 validates_inclusion_of and validates_exclusion_of

8.2.8 validates_length_of

8.2.9 validates_numericality_of

8.2.10 validates_presence_of

8.2.11 validates_uniqueness_of

8.2.12 validates_with

8.2.13 RecordInvalid

8.3 Common Validation Options

8.3.1 allow_blank and allow_nil

8.3.2 if and unless

8.3.3 message

8.3.4 on

8.3.5 strict

8.4 Conditional Validation

8.4.1 Usage and Considerations

Table of Contents

8.4.2 Validation Contexts

8.5 Short-Form Validation

8.6 Custom Validation Techniques

8.6.1 Custom Validation Macros

8.6.2 Create a Custom Validator Class

8.6.3 Add a validate Method to Your Model

8.7 Skipping Validations

8.8 Working with the Errors Hash

8.8.1 Checking for Errors

8.9 Testing Validations with Shoulda

8.10 Conclusion

Chapter 9 Advanced Active Record

9.1 Scopes

9.1.1 Scope Parameters

9.1.2 Chaining Scopes

9.1.3 Scopes and has_many

9.1.4 Scopes and Joins

9.1.5 Scope Combinations

9.1.6 Default Scopes

9.1.7 Using Scopes for CRUD

9.2 Callbacks

9.2.1 One-Liners

9.2.2 Protected or Private

9.2.3 Matched before/after Callbacks

9.2.4 Halting Execution

9.2.5 Callback Usages

9.2.6 Special Callbacks: after_initialize and after_find

9.2.7 Callback Classes

Table of Contents

9.3 Attributes API

9.3.1 Type Coercion

9.3.2 The attribute Method

9.3.3 Custom Types

9.3.4 Registering New Attribute Types

9.4 Serialized Attributes

9.4.1 ActiveRecord::Store

9.4.2 Native Database Support for Serialized Attributes

9.5 Enums

9.5.1 Prefixes and Suffixes

9.5.2 Reflection

9.5.3 State Machine

9.6 Generating Secure Tokens

9.6.1 Migrations

9.6.2 Regenerating a Token

9.7 Calculation Methods

9.7.1 average(column_name, *options)

9.7.2 count(column_name, *options)

9.7.3 ids

9.7.4 maximum(column_name, *options)

9.7.5 minimum(column_name, *options)

9.7.6 pluck(*column_names)

9.7.7 sum(column_name, *options)

9.8 Batch Operations

9.8.1 Creating Many Records

9.8.2 Reading Many Records at Once

9.8.3 Updating Many Records at Once

9.8.4 Bulk Deletion

Table of Contents

9.9 Single-Table Inheritance (STI)

9.9.1 Mapping Inheritance to the Database

9.9.2 STI Considerations

9.9.3 STI and Associations

9.10 Abstract Base Model Classes

9.11 Polymorphic has_many Relationships

9.11.1 Modeling User Comments

9.12 Foreign-Key Constraints

9.12.1 Foreign Key Names and Indexes

9.12.2 Removing Foreign Keys

9.13 Modules for Reusing Common Behavior

9.13.1 A Review of Class Scope and Contexts

9.13.2 The included Callback

9.14 Value Objects

9.14.1 Immutability

9.15 Non-persisted Models

9.16 Modifying Active Record Classes at Runtime

9.16.1 Considerations

9.16.2 Ruby and Domain-Specific Languages

9.17 PostgreSQL

9.17.1 Array Type

9.17.2 Network Address Types

9.17.3 UUID Type

9.17.4 Range Types

9.17.5 JSON Type

9.17.6 Index Types

9.18 Conclusion

Chapter 10 Action View

Table of Contents

10.1 Layouts and Templates

- 10.1.1 Template Filename Conventions
- 10.1.2 Layouts
- 10.1.3 Yielding Content
- 10.1.4 Conditional Output
- 10.1.5 Decent Exposure
- 10.1.6 Standard Instance Variables
- 10.1.7 Displaying flash Messages

10.2 Partial

- 10.2.1 Simple Use Cases
- 10.2.2 Reuse of Partial
- 10.2.3 Shared Partial
- 10.2.4 Passing Variables to Partial
- 10.2.5 Rendering an Object
- 10.2.6 Rendering Collections
- 10.2.7 Logging

10.3 Conclusion

Chapter 11 All about Helpers

11.1 ActiveModelHelper

11.2 AssetTagHelper

- 11.2.1 Head Helpers
- 11.2.2 Asset Helpers
- 11.2.3 For Plugins Only

11.3 AssetUrlHelper

- 11.3.1 `asset_path(source, options = {})`
- 11.3.2 `asset_url(source, options = {})`
- 11.3.3 `audio_path(source, options = {})`
- 11.3.4 `audio_url(source, options = {})`

Table of Contents

- 11.3.5 font_path(source, options = {})
- 11.3.6 font_url(source, options = {})
- 11.3.7 image_path(source)
- 11.3.8 image_url(source, options = {})
- 11.3.9 javascript_path(source, options = {})
- 11.3.10 javascript_url(source, options = {})
- 11.3.11 stylesheet_path(source, options = {})
- 11.3.12 stylesheet_url(source, options = {})
- 11.3.13 video_path
- 11.3.14 video_url(source, options = {})
- 11.3.15 Using Asset Hosts
- 11.4 AtomFeedHelper
- 11.5 CacheHelper
- 11.6 CaptureHelper
 - 11.6.1 capture(&block)
 - 11.6.2 content_for(name, &block)
 - 11.6.3 content_for?(name)
 - 11.6.4 provide(name, content = nil, &block)
- 11.7 ControllerHelper
- 11.8 CsrfHelper
- 11.9 DateHelper
 - 11.9.1 The Date and Time Selection Helpers
 - 11.9.2 The Individual Date and Time Select Helpers
 - 11.9.3 Common Options for Date Selection Helpers
 - 11.9.4 distance_in_time Methods with Complex Descriptive Names
 - 11.9.5 time_tag(date_or_time, *args, &block)
- 11.10 DebugHelper
- 11.11 FormHelper

Table of Contents

- 11.11.1 Creating Forms for Models
- 11.11.2 How Form Helpers Get Their Values
- 11.11.3 Integrating Additional Objects in One Form
- 11.11.4 Customized Form Builders
- 11.11.5 Form Inputs

11.12 FormOptionsHelper

- 11.12.1 Select Helpers
- 11.12.2 Checkbox/Radio Helpers
- 11.12.3 Option Helpers

11.13 FormTagHelper

- 11.13.1 `button_tag(content_or_options = nil, options = nil, &block)`
- 11.13.2 `check_box_tag(name, value = "1", checked = false, options = {})`
- 11.13.3 `color_field_tag(name, value = nil, options = {})`
- 11.13.4 `date_field_tag(name, value = nil, options = {})`
- 11.13.5 `datetime_field_tag(name, value = nil, options = {})`
- 11.13.6 `datetime_local_field_tag(name, value = nil, options = {})`
- 11.13.7 `email_field_tag(name, value = nil, options = {})`
- 11.13.8 `field_set_tag(legend = nil, options = nil, &block)`
- 11.13.9 `file_field_tag(name, options = {})`
- 11.13.10 `form_tag(url_for_options = {}, options = {}, &block)`
- 11.13.11 `hidden_field_tag(name, value = nil, options = {})`
- 11.13.12 `image_submit_tag(source, options = {})`
- 11.13.13 `label_tag(name = nil, content_or_options = nil, options = nil, &block)`
- 11.13.14 `month_field_tag(name, value = nil, options = {})`
- 11.13.15 `number_field_tag(name, value = nil, options = {})`
- 11.13.16 `password_field_tag(name = "password", value = nil, options = {})`
- 11.13.17 `radio_button_tag(name, value, checked = false, options = {})`
- 11.13.18 `range_field_tag(name, value = nil, options = {})`
- 11.13.19 `search_field_tag(name, value = nil, options = {})`

Table of Contents

- 11.13.20 `select_tag(name, option_tags = nil, options = {})`
- 11.13.21 `submit_tag(value = "Save changes", options = {})`
- 11.13.22 `telephone_field_tag(name, value = nil, options = {})`
- 11.13.23 `text_area_tag(name, content = nil, options = {})`
- 11.13.24 `text_field_tag(name, value = nil, options = {})`
- 11.13.25 `time_field_tag(name, value = nil, options = {})`
- 11.13.26 `url_field_tag(name, value = nil, options = {})`
- 11.13.27 `utf8_enforcer_tag()`
- 11.13.28 `week_field_tag(name, value = nil, options = {})`

11.14 JavaScriptHelper

- 11.14.1 `escape_javascript(javascript)`
- 11.14.2 `javascript_tag(content_or_options_ with_block = nil, html_options = {}, &block)`

11.15 NumberHelper

- 11.15.1 `number_to_currency(number, options = {})`
- 11.15.2 `number_to_human_size(number, options = {})`
- 11.15.3 `number_to_human_size(number, options = {})`
- 11.15.4 `number_to_percentage(number, options = {})`
- 11.15.5 `number_to_phone(number, options = {})`
- 11.15.6 `number_with_delimiter(number, options = {})`
- 11.15.7 `number_with_precision(number, options = {})`

11.16 OutputSafetyHelper

- 11.16.1 `raw(stringish)`
- 11.16.2 `safe_join(array, sep=$,)`
- 11.16.3 `to_sentence(array, options = {})`

11.17 RecordTagHelper

- 11.17.1 `content_tag_for(tag_name, single_or_ multiple_records, prefix = nil, options = nil, &block)`
- 11.17.2 `div_for(record, *args, &block)`

Table of Contents

11.18 RenderingHelper

11.19 SanitizeHelper

11.19.1 `sanitize(html, options = {})`

11.19.2 `sanitize_css(style)`

11.19.3 `strip_links(html)`

11.19.4 `strip_tags(html)`

11.20 TagHelper

11.20.1 `CDATA_section(content)`

11.20.2 `content_tag(name, content_or_options_ with_block = nil, options = nil, escape = true, &block)`

11.20.3 `escape_once(html)`

11.20.4 `tag(name, options = nil, open = false, escape = true)`

11.21 TextHelper

11.21.1 `concat(string)`

11.21.2 `current_cycle(name = "default")`

11.21.3 `cycle(first_value, *values)`

11.21.4 `xcerpt(text, phrase, options = {})`

11.21.5 `highlight(text, phrases, options = {})`

11.21.6 `pluralize(count, singular, plural = nil)`

11.21.7 `reset_cycle(name = "default")`

11.21.8 `simple_format(text, html_options = {}, options = {})`

11.21.9 `truncate(text, options = {}, &block)`

11.21.10 `word_wrap(text, options = {})`

11.22 TranslationHelper and the I18n API

11.22.1 Localized Views

11.22.2 TranslationHelper Methods

11.22.3 I18n Setup

11.22.4 Setting and Passing the Locale

11.22.5 Internationalizing Your Application

Table of Contents

11.22.6 Organization of Locale Files

11.22.7 Looking Up Translations

11.22.8 How to Store Your Custom Translations

11.22.9 Overview of Other Built-In Methods That Provide I18n Support

11.22.10 Exception Handling

11.23 UrlHelper

11.23.1 button_to(name = nil, options = nil, html_options = nil, &block)

11.23.2 current_page?(options)

11.23.3 link_to(name = nil, options = nil, html_options = nil, &block)

11.23.4 link_to_if(condition, name, options = {}, html_options = {}, &block)

11.23.5 link_to_unless(condition, name, options = {}, html_options = {}, &block)

11.23.6 link_to_unless_current(name, options = {}, html_options = {}, &block)

11.23.7 mail_to(email_address, name = nil, html_options = {}, &block)

11.24 Writing Your Own View Helpers

11.24.1 Small Optimizations: The title Helper

11.24.2 Encapsulating View Logic: The photo_for Helper

11.24.3 Smart View: The breadcrumbs Helper

11.25 Wrapping and Generalizing Partial

11.25.1 A tiles Helper

11.25.2 Generalizing Partial

11.26 Conclusion

Chapter 12 Haml

12.1 Getting Started

12.2 The Basics

12.2.1 Creating an Element

12.2.2 Attributes

12.2.3 Classes and IDs

12.2.4 Implicit Divs

Table of Contents

12.2.5 Empty Tags

12.3 Doctype

12.4 Comments

12.4.1 HTML Comments

12.4.2 Haml Comments

12.5 Evaluating Ruby Code

12.5.1 Interpolation

12.5.2 Escaping/Unescaping HTML

12.5.3 Escaping the First Character of a Line

12.5.4 Multiline Declarations

12.6 Helpers

12.6.1 Object Reference []

12.6.2 page_class

12.6.3 list_of(enum, opts = {}) { |item| ... }

12.7 Filters

12.8 Haml and Content

12.9 Configuration Options

12.9.1 autoclose

12.9.2 cdata

12.9.3 compiler_class

12.9.4 encoding

12.9.5 escape_attrs

12.9.6 escape_html

12.9.7 format

12.9.8 hyphenate_data_attrs

12.9.9 mime_type

12.9.10 parser_class

12.9.11 preserve

Table of Contents

12.9.12 remove_whitespace

12.9.13 ugly

12.10 Conclusion

Chapter 13 Session Management

13.1 What to Store in the Session

13.1.1 The Current User

13.1.2 Session Use Guidelines

13.2 Storage Mechanisms

13.2.1 Active Record Session Store

13.2.2 Memcached Session Storage

13.2.3 The Controversial CookieStore

13.3 Cookies

13.3.1 Reading and Writing Cookies

13.4 Conclusion

Chapter 14 Authentication and Authorization

14.1 Warden

14.1.1 Middleware

14.1.2 Strategies

14.1.3 Scopes

14.2 Devise

14.2.1 Getting Started

14.2.2 Modules

14.2.3 Models

14.2.4 Mappings / Scopes

14.2.5 Controllers

14.2.6 Routing

14.2.7 Views

14.2.8 Configuration

Table of Contents

- 14.2.9 Strong Parameters
- 14.2.10 Extensions
- 14.2.11 Testing with Devise
- 14.2.12 Summary

14.3 has_secure_password

- 14.3.1 Getting Started
- 14.3.2 Creating the Models
- 14.3.3 Setting Up the Controllers
- 14.3.4 Controller, Limiting Access to Actions
- 14.3.5 Summary

14.4 Pundit

- 14.4.1 Getting Started
- 14.4.2 Creating a Policy
- 14.4.3 Controller Integration
- 14.4.4 Policy Scopes
- 14.4.5 Strong Parameters
- 14.4.6 Testing Policies

14.5 Conclusion

Chapter 15 Security

15.1 Password Management

- 15.1.1 BCrypt

15.2 Log Masking

15.3 SSL (Secure Sockets Layer)

15.4 Model Mass-Assignment Attributes Protection

15.5 SQL Injection

- 15.5.1 What Is a SQL Injection?

15.6 Cross-Site Scripting (XSS)

- 15.6.1 HTML Escaping

Table of Contents

15.6.2 HTML Sanitization

15.6.3 Input versus Output Escaping

15.7 XSRF (Cross-Site Request Forgery)

15.7.1 Restricting HTTP Method for Actions with Side-Effects

15.7.2 Require Security Token for Protected Requests

15.7.3 Client-Side Security Token Handling

15.8 Session Fixation Attacks

15.9 Keeping Secrets

15.10 Conclusion

Chapter 16 Action Mailer

16.1 Mailer Models

16.1.1 Preparing Outbound Email Messages

16.1.2 Decent Exposure

16.1.3 Mailer Views

16.1.4 HTML Email Messages

16.1.5 The render Method

16.1.6 Multipart Messages

16.1.7 Attachments

16.1.8 Generating URLs

16.1.9 Mailer Layouts

16.1.10 Sending an Email

16.1.11 Asynchronous Delivery

16.1.12 Callbacks

16.1.13 Interceptors

16.2 Previews

16.3 Receiving Emails

16.3.1 The receive Method

16.3.2 Handling Incoming Attachments

Table of Contents

16.4 Testing Email Content

16.5 Sending via API

16.6 Configuration

16.7 Conclusion

Chapter 17 Caching and Performance

17.1 View Caching

17.1.1 Page Caching

17.1.2 Action Caching

17.1.3 Fragment Caching

17.1.4 Russian-Doll Caching

17.1.5 Collection Caching

17.1.6 Conditional Caching

17.1.7 Expiration of Cached Content

17.1.8 Automatic Cache Expiry with Sweepers

17.1.9 Avoiding Extra Database Activity

17.1.10 Cache Logging

17.1.11 Cache Storage

17.2 Data Caching

17.2.1 Eliminating Extra Database Lookups

17.2.2 Initializing New Caches

17.2.3 fetch Options

17.3 Control of Web Caching

17.3.1 expires_in(seconds, options =)

17.3.2 expires_now

17.4 ETags

17.4.1 fresh_when(options)

17.4.2 stale?(options)

17.5 Conclusion

Table of Contents

Chapter 18 Background Processing

18.1 Active Job

- 18.1.1 Creating a Job
- 18.1.2 Job Generator
- 18.1.3 Enqueueing a Job
- 18.1.4 Delayed Execution
- 18.1.5 Immediate Execution
- 18.1.6 Callbacks / Hooks
- 18.1.7 Declarative Error Handling
- 18.1.8 Integration with Action Mailer
- 18.1.9 GlobalID

18.2 Queueing Backends

- 18.2.1 Sucker Punch
- 18.2.2 Sidekiq
- 18.2.3 Resque
- 18.2.4 Que
- 18.2.5 Delayed Job

18.3 Rails Runner

- 18.3.1 Getting Started
- 18.3.2 Usage Notes
- 18.3.3 Considerations
- 18.3.4 Summary

18.4 Conclusion

Chapter 19 Asset Pipeline

19.1 Introduction to Asset Management

19.2 Organization. Where Does Everything Go?

19.3 Manifest Files

- 19.3.1 Manifest Directives

Table of Contents

19.3.2 Search Path

19.3.3 Gemified Assets

19.3.4 Index Files

19.3.5 Format Handlers

19.4 Custom Format Handlers

19.5 Post-Processing

19.5.1 Stylesheets

19.5.2 JavaScripts

19.5.3 Custom Compressor

19.6 Helpers

19.6.1 Images

19.6.2 Getting the URL of an Asset File

19.6.3 Built-in Sass Asset Path Helpers

19.6.4 Data URLs

19.7 Fingerprinting

19.8 Serving the Files

19.8.1 Configuration Settings

19.8.2 Who Delivers the Assets?

19.8.3 GZip Compression

19.9 Rake Tasks

19.10 Yarn

19.11 Webpack

19.11.1 Webpack Executables

19.11.2 Linking to Webpack Processed Files in Your Application

19.12 Conclusion

Chapter 20 Ajax on Rails

20.1 Unobtrusive JavaScript

20.1.1 Helpers

Table of Contents

20.1.2 UJS Custom Events

20.2 Ajax and JSON

20.2.1 Ajax link_to

20.3 Ajax and HTML

20.4 JSONP Requests

20.5 Conclusion

Chapter 21 Turbolinks

21.1 Turbolinks Usage

21.1.1 Each Navigation Is a Visit

21.1.2 Application Visits

21.1.3 Restoration Visits

21.1.4 Canceling Visits Before They Start

21.1.5 Disabling Turbolinks on Specific Links

21.2 Building Your Turbolinks Application

21.2.1 Running JavaScript When a Page Loads

21.2.2 Working with Script Elements

21.3 Understanding Turbolinks Caching

21.3.1 Preparing the Page to Be Cached

21.3.2 Detecting When a Preview Is Visible

21.3.3 Opting Out of Caching

21.4 Making Transformations Idempotent

21.5 Responding to Page Updates

21.6 Persisting Elements across Page Loads

21.7 Advanced Turbolinks

21.7.1 Displaying Progress

21.7.2 Reloading When Certain Things Change

21.7.3 Setting a Root Location

21.7.4 Following Redirects

Table of Contents

21.7.5 Redirecting after a Form Submission

21.7.6 Setting Custom HTTP Headers

21.8 Turbolinks API Reference

21.8.1 Turbolinks.visit

21.8.2 Turbolinks.clearCache

21.8.3 Turbolinks.supported

21.9 Turbolinks Events

21.10 Conclusion

Chapter 22 Action Cable

22.1 Web Sockets

22.2 Publish-Subscribe Pattern

22.3 Connections

22.4 Channels

22.5 Subscriptions

22.5.1 Subscribers

22.6 Streams

22.6.1 Broadcasting

22.7 Subscriptions Revisited (Browser-Side)

22.8 Rebroadcasting

22.9 Channel Actions

22.10 Configuration

22.10.1 Subscription Adapter

22.10.2 Allowed Request Origins

22.10.3 Consumer Configuration

22.10.4 Worker Count

22.11 Running Stand-Alone Cable Servers

22.12 Generator

Table of Contents

22.13 Conclusion

Chapter 23 RSpec

23.1 Introduction

23.2 Behavior-Driven Development

23.2.1 Up and Down the Stack . . .

23.3 Basic Syntax and API

23.3.1 Contexts (aka Example Groups)

23.3.2 Shared Variables

23.3.3 Examples

23.3.4 Expectations

23.3.5 Metadata

23.3.6 Hooks

23.3.7 Implicit Subject

23.3.8 Explicit Subject

23.4 Custom Expectation Matchers

23.4.1 Custom Matcher DSL

23.4.2 Fluent Chaining

23.5 Helper Methods

23.6 Shared Behaviors

23.7 Shared Context

23.8 Mocks and Stubs

23.8.1 Test Doubles

23.8.2 Stubs

23.8.3 Null Objects

23.8.4 Partial Mocking and Stubbing

23.8.5 `receive_message_chain`

23.9 Running Specs

23.9.1 Failing and Skipping Examples

Table of Contents

23.9.2 Filtering

23.10 Factory Girl

23.10.1 Setup

23.10.2 Factory Definitions

23.10.3 Usage

23.10.4 Static versus Dynamic Attributes

23.10.5 Aliases

23.10.6 Dependent Attributes

23.10.7 Associations

23.10.8 Sequences

23.10.9 Traits

23.10.10 Callbacks

23.10.11 Building or Creating Multiple Records

23.10.12 Factory Girl and Active Record

23.11 RSpec and Rails

23.11.1 spec_helper.rb

23.11.2 rails_helper.rb

23.11.3 Model Specs

23.11.4 Routing Specs

23.11.5 Controller Specs

23.11.6 View Specs

23.11.7 Helper Specs

23.11.8 RSpec and Generators

23.11.9 RSpec Options

23.11.10 Support Files

23.12 Feature Specs with Capybara

23.12.1 Capybara DSL

23.13 Working with Files in Your Specs

23.14 RSpec Tools

Table of Contents

23.14.1 Database Cleaner

23.14.2 Guard-RSpec

23.14.3 Spring

23.14.4 Flatware

23.14.5 Specjour

23.14.6 SimpleCov

23.14.7 Timecop

23.14.8 WebMock

23.14.9 VCR

23.15 Conclusion

Appendix A: Active Model API Reference

A.1 AttributeAssignment

A.1.1 Instance Public Methods

A.2 AttributeMethods

A.2.1 Instance Public Methods

A.3 Callbacks

A.3.1 Instance Public Methods

A.4 Conversion

A.4.1 Instance Public Methods

A.5 Dirty

A.5.1 Instance Public Methods

A.6 Errors

A.6.1 Class Public Methods

A.6.2 Instance Public Methods

A.7 ForbiddenAttributesError

A.8 Lint::Tests

A.8.1 Instance Public Methods

A.9 MissingAttributeError

Table of Contents

A.10 Model

A.10.1 Class Public Methods

A.10.2 Instance Public Methods

A.11 Name

A.11.1 Class Public Methods

A.11.2 Instance Public Attributes

A.11.3 Instance Public Methods

A.12 Naming

A.12.1 Class Public Methods

A.12.2 Instance Public Methods

A.13 SecurePassword

A.14 Serialization

A.14.1 Instance Public Methods

A.15 Serializers::JSON

A.15.1 `active_model/serializers/json.rb`

A.16 Translation

A.16.1 Instance Public Methods

A.17 Type

A.17.1 Class Public Methods

A.18 ValidationError

A.19 Validations

A.19.1 Options

A.19.2 Class Public Methods

A.19.3 Instance Public Methods

A.19.4 Callbacks

A.20 Validator

A.20.1 Class Public Methods

A.20.2 Instance Public Methods

Table of Contents

Appendix B: Active Support API Reference

B.1 Array

- B.1.1 `active_support/core_ext/array/ access.rb`
- B.1.2 `active_support/core_ext/array/ conversions.rb`
- B.1.3 `active_support/core_ext/array/ extract_options.rb`
- B.1.4 `active_support/core_ext/array/ grouping.rb`
- B.1.5 `active_support/core_ext/array/ inquiry.rb`
- B.1.6 `active_support/core_ext/array/ prepend_and_append.rb`
- B.1.7 `active_support/core_ext/array/ wrap.rb`

B.2 Autoload

- B.2.1 `autoload(const_name, path)`
- B.2.2 `autoloads`
- B.2.3 `autoload_at(path)`
- B.2.4 `autoload_under(path)`
- B.2.5 `eager_autoload`
- B.2.6 `eager_load!`

B.3 BacktraceCleaner

B.4 Benchmark

- B.4.1 `ms`

B.5 Benchmarkable

- B.5.1 `benchmark(message = "Benchmarking", options = {})`

B.6 BigDecimal

- B.6.1 `active_support/core_ext/big_decimal/ conversions.rb`
- B.6.2 `active_support/json/encoding.rb`

B.7 Cache::FileStore

B.8 Cache::MemCacheStore

B.9 Cache::MemoryStore

B.10 Cache::NullStore

Table of Contents

B.11 Cache::Store

- B.11.1 cleanup(options = nil)
- B.11.2 clear(options = nil)
- B.11.3 decrement(name, amount = 1, options = nil)
- B.11.4 delete(name, options = nil)
- B.11.5 delete_matched(matcher, options = nil)
- B.11.6 exist?(name, options = nil)
- B.11.7 fetch(name, options = nil)
- B.11.8 increment(name, amount = 1, options = nil)
- B.11.9 mute
- B.11.10 options
- B.11.11 read(name, options = nil)
- B.11.12 read_multi(*names)
- B.11.13 silence!
- B.11.14 write(name, value, options = nil)

B.12 CachingKeyGenerator

- B.12.1 initialize(key_generator)
- B.12.2 generate_key(salt, key_size=64)

B.13 Callbacks

- B.13.1 active_support/callbacks

B.14 Class

- B.14.1 active_support/core_ext/class/ attribute.rb
- B.14.2 active_support/core_ext/class/ attribute_accessors.rb
- B.14.3 active_support/core_ext/class/ subclasses.rb

B.15 Concern

- B.15.1 active_support/concern.rb

B.16 Configurable

- B.16.1 active_support/configurable.rb

B.17 Date

Table of Contents

B.17.1 active_support/core_ext/date/ calculations.rb

B.17.2 active_support/core_ext/date/ conversions.rb

B.17.3 active_support/core_ext/date/ zones.rb

B.17.4 active_support/json/encoding.rb

B.18 DateTime

B.18.1 active_support/core_ext/date_time/cal- culations.rb

B.18.2 active_support/core_ext/date_and_time/ zones.rb

B.19 DateTime

B.19.1 active_support/core_ext/date_time/ acts_like.rb

B.19.2 active_support/core_ext/date_time/cal- culations.rb

B.19.3 active_support/core_ext/date_time/con- versions.rb

B.19.4 active_support/json/encoding.rb

B.20 Dependencies

B.20.1 active_support/dependencies.rb

B.21 DescendantsTracker

B.21.1 active_support/descendants_ tracker.rb

B.22 Digest::UUID

B.22.1 Digest::UUID.uuid_from_hash(hash_ class, uuid_namespace, name)

B.22.2 Digest::UUID.uuid_v3(uuid_ namespace, name)

B.22.3 Digest::UUID.uuid_v4

B.22.4 Digest::UUID.uuid_v5(uuid_ namespace, name) 844 B.23 Duration

B.23 Duration

B.23.1 + (other)

B.23.2 - (other)

B.23.3 ago(time = Time.current)

B.23.4 from_now(time = Time.current)

B.23.5 inspect

B.23.6 since(time = Time.current)

B.23.7 until(time = Time.current)

Table of Contents

B.23.8 Duration.parse(iso8601duration)

B.23.9 iso8601(precision: nil)

B.24 Enumerable

B.24.1 exclude?

B.24.2 index_by(&block)

B.24.3 many?

B.24.4 pluck(*keys)

B.24.5 sum(identity = 0, &block)

B.24.6 without(*elements)

B.25 ERB::Util

B.25.1 html_escape(s)

B.25.2 html_escape_once(s)

B.25.3 json_escape(s)

B.26 EventedFileUpdateChecker

B.27 FalseClass

B.27.1 active_support/core_ext/object/ blank.rb

B.27.2 active_support/json/encoding.rb

B.28 File

B.28.1 atomic_write(file_name, temp_dir = Dir.tmpdir)

B.29 FileUpdateChecker

B.29.1 initialize(files, dirs = {}, &block)

B.30 Gzip

B.30.1 Gzip.compress(source, level= Zlib::DEFAULT_COMPRESSION,
strategy=Zlib::DEFAULT_STRATEGY)

B.30.2 Gzip.decompress(source)

B.31 Hash

B.31.1 active_support/core_ext/array/ extract_options.rb

B.31.2 active_support/core_ext/hash/ compact.rb

Table of Contents

- B.31.3 `active_support/core_ext/hash/ conversions.rb`
- B.31.4 `active_support/core_ext/hash/ deep_merge.rb`
- B.31.5 `active_support/core_ext/hash/ except.rb`
- B.31.6 `active_support/core_ext/hash/ indifferent_access.rb`
- B.31.7 `active_support/core_ext/hash/ keys.rb`
- B.31.8 `active_support/core_ext/hash/reverse_ merge.rb`
- B.31.9 `active_support/core_ext/hash/ slice.rb`
- B.31.10 `active_support/core_ext/object/ to_query.rb`
- B.31.11 `active_support/json/encoding.rb`
- B.31.12 `active_support/core_ext/object/ blank.rb`

B.32 HashWithIndifferentAccess

B.33 Inflector

B.34 Inflector::Inflections

- B.34.1 `acronym(word)`
- B.34.2 `clear(scope = :all)`
- B.34.3 `human(rule, replacement)`
- B.34.4 `inflections(locale = :en)`
- B.34.5 `irregular(singular, plural)`
- B.34.6 `plural(rule, replacement)`
- B.34.7 `singular(rule, replacement)`
- B.34.8 `uncountable(*words)`

B.35 Integer

- B.35.1 `active_support/core_ext/integer/ inflections.rb`
- B.35.2 `active_support/core_ext/integer/ multiple.rb`

B.36 JSON

- B.36.1 `decode(json)`
- B.36.2 `encode(value, options = nil)`

B.37 Kernel

Table of Contents

B.37.1 active_support/core_ext/kernel/ reporting.rb

B.37.2 active_support/core_ext/kernel/ singleton_class.rb

B.38 KeyGenerator

B.38.1 initialize(secret, options = {})

B.38.2 generate_key(salt, key_size=64)

B.39 LazyLoadHooks

B.39.1 on_load(name, options = {}, &block)

B.39.2 run_load_hooks(name, base = Object)

B.40 Locale

B.41 LogSubscriber

B.41.1 color(text, color, bold = false)

B.41.2 ActiveSupport::LogSubscriber:: TestHelper

B.42 Logger

B.42.1 active_support/logger.rb

B.42.2 active_support/logger_silence.rb

B.43 MessageEncryptor

B.43.1 active_support/message_encryptor.rb

B.44 MessageVerifier

B.44.1 active_support/message_verifier.rb

B.45 Module

B.45.1 active_support/core_ext/module/ aliasing.rb

B.45.2 active_support/core_ext/module/ anonymous.rb

B.45.3 active_support/core_ext/module/ attribute_accessors.rb

B.45.4 active_support/core_ext/module/ attr_internal.rb

B.45.5 active_support/core_ext/module/ delegation.rb

B.45.6 active_support/core_ext/module/ deprecation.rb

B.45.7 active_support/core_ext/module/ introspection.rb

B.45.8 active_support/core_ext/module/ remove_method.rb

Table of Contents

B.45.9 `active_support/dependencies.rb`

B.46 `Module::Concerning`

B.46.1 `concerning(topic, &block)`

B.46.2 `concern(topic, &module_definition)`

B.47 `Multibyte::Chars`

B.47.1 `active_support/multibyte/chars.rb`

B.48 `Multibyte::Unicode`

B.48.1 `Unicode.compose(codepoints)`

B.48.2 `Unicode.decompose(type, codepoints)`

B.48.3 `Unicode.downcase(string)`

B.48.4 `Unicode.in_char_class?(codepoint, classes)`

B.48.5 `Unicode.normalize(string, form = nil)`

B.48.6 `Unicode.pack_graphemes(unpacked)`

B.48.7 `Unicode.reorder_characters (codepoints)`

B.48.8 `Unicode.swapcase(string)`

B.48.9 `Unicode.tidy_bytes(string, force = false)`

B.48.10 `Unicode.unpack_graphemes(string)`

B.48.11 `Unicode.upcase(string)`

B.49 `NameError`

B.49.1 `missing_name`

B.49.2 `missing_name?(name)`

B.50 `NilClass`

B.50.1 `active_support/core_ext/object/ blank.rb`

B.50.2 `active_support/core_ext/object/ try.rb`

B.50.3 `active_support/json/encoding.rb`

B.51 `Notifications`

B.51.1 `Instrumenters`

B.51.2 `Subscribers`

Table of Contents

B.51.3 Temporary Subscriptions

B.51.4 Notifications::Fanout

B.52 NumberHelper

B.53 Numeric

B.53.1 active_support/core_ext/object/ blank.rb

B.53.2 active_support/json/encoding.rb

B.53.3 active_support/core_ext/numeric/ bytes.rb

B.53.4 active_support/core_ext/numeric/ conversions.rb

B.53.5 active_support/core_ext/numeric/ inquiry.rb

B.53.6 active_support/core_ext/numeric/ time.rb

B.54 Object

B.54.1 active_support/core_ext/kernel/ agnostics.rb

B.54.2 active_support/core_ext/object/ acts_like.rb

B.54.3 active_support/core_ext/object/ blank.rb

B.54.4 active_support/core_ext/object/ deep_dup.rb

B.54.5 active_support/core_ext/object/ duplicable.rb

B.54.6 active_support/core_ext/object/ inclusion.rb

B.54.7 active_support/core_ext/object/ instance_variables.rb

B.54.8 active_support/core_ext/object/ json.rb

B.54.9 active_support/core_ext/object/ to_param.rb

B.54.10 active_support/core_ext/object/ to_query.rb

B.54.11 active_support/core_ext/object/ try.rb

B.54.12 active_support/core_ext/object/with_ options.rb

B.54.13 active_support/dependencies.rb

B.55 OrderedOptions

B.56 ProxyObject

B.57 Railtie

B.58 Range

Table of Contents

B.58.1 `active_support/core_ext/enumerable.rb`

B.58.2 `active_support/core_ext/range/conversions.rb`

B.58.3 `active_support/core_ext/range/include_range.rb`

B.58.4 `active_support/core_ext/range/overlaps.rb`

B.59 Regexp

B.59.1 `active_support/core_ext/regexp.rb`

B.59.2 `active_support/json/encoding.rb`

B.60 Rescuable

B.60.1 `rescue_from(*classes, &block)`

B.60.2 `rescue_with_handler(exception, object: self, visited_exceptions: [])`

B.61 SecureRandom

B.61.1 `base58(n = 16)`

B.62 SecurityUtils

B.62.1 `secure_compare(a, b)`

B.63 String

B.63.1 `active_support/json/encoding.rb`

B.63.2 `active_support/core_ext/object/blank.rb`

B.63.3 `active_support/core_ext/string/access.rb`

B.63.4 `active_support/core_ext/string/conversions.rb`

B.63.5 `active_support/core_ext/string/exclude.rb`

B.63.6 `active_support/core_ext/string/filters.rb`

B.63.7 `active_support/core_ext/string/indent.rb`

B.63.8 `active_support/core_ext/string/inflections.rb`

B.63.9 `active_support/core_ext/string/inquiry.rb`

B.63.10 `active_support/core_ext/string/multibyte.rb`

B.63.11 `active_support/core_ext/string/output_safety.rb`

B.63.12 `active_support/core_ext/string/starts_ends_with.rb`

B.63.13 `active_support/core_ext/string/strip.rb`

B.63.14 `active_support/core_ext/string/in_time_zone.rb`

Table of Contents

B.64 StringInquirer

B.65 Subscriber

B.66 TaggedLogging

B.66.1 active_support/tagged_logger.rb

B.67 TestCase

B.67.1 active_support/test_case.rb

B.68 Testing::Assertions

B.68.1 active_support/testing/ assertions.rb

B.68.2 active_support/testing/ declarative.rb

B.68.3 active_support/testing/file_ fixtures.rb

B.68.4 active_support/testing/setup_and_ teardown.rb

B.68.5 active_support/testing/time_ helpers.rb

B.69 Thread

B.69.1 active_support/core_ext/thread.rb

B.70 Time

B.70.1 active_support/json/encoding.rb

B.70.2 active_support/core_ext/time/ acts_like.rb

B.70.3 active_support/core_ext/time/ calculations.rb

B.70.4 active_support/core_ext/time/ conversions.rb

B.70.5 DATE_FORMATS

B.70.6 active_support/core_ext/time/ marshal.rb

B.70.7 active_support/core_ext/time/ zones.rb

B.71 TimeWithZone

B.72 TimeZone

B.72.1 active_support/values/time_ zone.rb

B.73 TrueClass

B.73.1 active_support/core_ext/object/ blank.rb

B.73.2 active_support/json/encoding.rb

Table of Contents

B.74 XmlMini

B.74.1 active_support/xml_mini.rb

B.74.2 Constants

Appendix C: Rails API

C.1 Rails API Mode

C.1.1 Serving JSON by Default

C.2 JSON

C.2.1 Jbuilder

C.2.2 ActiveModel Serializers

Index