

# Beginning Ruby on Rails

Session One

# Anthony Lewis

- Information Security Analyst at TEA
- Freelance Developer (Ruby / Javascript)
- Teacher... ?
- <http://anthonylewis.com>

# About You

- The class roll is online:
  - <http://radiant-stone-722.herokuapp.com>
- Please add your name and e-mail address
- The URL and Bio are optional

# Session Overview

- Installation / Configuration
- Tools Overview
- Your First Application
- Homework



A close-up photograph of a light-colored wooden board with numerous circular holes. Several wooden pegs of varying heights are inserted into some of the holes. A flat wooden plank is leaning against the board, its end resting in one of the holes. The scene is lit from the side, creating soft shadows.

# Installation

The Hardest Thing We'll Do In This Course

# What We Need

- Ruby 1.8.7 or 1.9.2
- Rails 3.0
- SQLite 3
- Git 1.7

# Windows

- Starting from scratch?
  - Use Rails Installer
  - Everything you need in one download
  - <http://railsinstaller.org>

# Windows - DIY

- Download and Install Ruby
  - <http://rubyinstaller.org>
  - `gem install rails`
- Download and Install Git
  - <http://git-scm.com>



# Mac OS X

- Check your Ruby version
  - `ruby --version`
- Have 1.8.7 already?
  - `sudo gem install rails`
- Download and Install Git
  - <http://git-scm.com>

# Linux

- Check your package system
  - Does it have Ruby 1.8.7?
  - Does it have RubyGems?
  - Does it have Git?
- Install Rails using Gem
  - `sudo gem install rails`

# Ruby Version Manager

- For Mac OS X and Linux
- Install multiple versions of Ruby
- Builds from source - dev tools required
- <https://rvm.beginrescueend.com>

# Gem Error?

- `gem --version`
- should be 1.3.7 or newer
- `gem update --system`

# Ready To Go?

- `ruby --version`
  - 1.8.7 or 1.9.2
- `rails --version`
  - `>= 3.0`
- `git --version`
  - `>= 1.7`



# Tools Overview

The Bits And Pieces That Make Everything Work

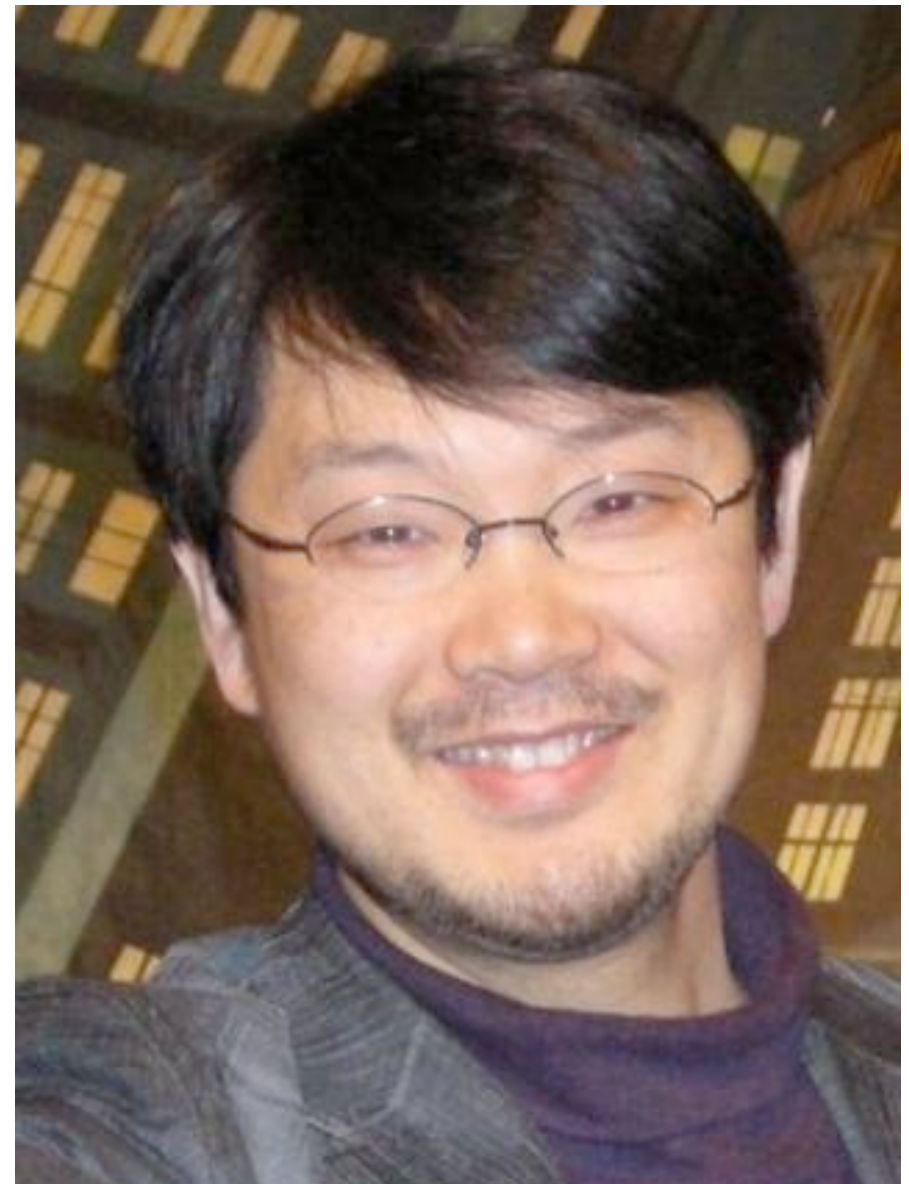


# Ruby

- A dynamic, open source programming language with a focus on simplicity and productivity. It has an elegant syntax that is natural to read and easy to write.
- Created by Yukihiro “Matz” Matsumoto
- <http://www.ruby-lang.org/en>

# MINASWAN

Matz is nice and so we are nice.



# Rails

- An open-source web framework that's optimized for programmer happiness and sustainable productivity. It lets you write beautiful code by favoring convention over configuration.
- Created by David Heinemeier Hanson
- <http://rubyonrails.org>

# Gem

- RubyGems is the premier ruby packaging system. It provides a standard format for distributing Ruby programs and libraries, and an easy to use tool for managing the installation of gem packages.
- <https://rubygems.org>

# Bundler

- Bundler manages an application's dependencies through its entire life across many machines systematically and repeatably.
- <http://gembundler.com>

# Git

- Git is a free & open source, distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
- <http://git-scm.com>
- <https://github.com>



# Text Editors

- The Classics: Emacs, Vim
- Mac Only: Textmate
- Linux: gedit, kate
- Windows: Notepad++, E Text Editor
- Cross Platform: Sublime Text 2, Redcar



# Your First Web App

The Moment We've All Been Waiting For...

# Create The App

- Open a command prompt
- Create a directory for this class:
  - `mkdir rails-class`
  - `cd rails-class`
- Type this command:
  - `rails new blog`

```
Terminal — bash — 120x30

create public/index.html
create public/robots.txt
create public/images
create public/images/rails.png
create public/stylesheets
create public/stylesheets/.gitkeep
create public/javascripts
create public/javascripts/application.js
create public/javascripts/controls.js
create public/javascripts/dragdrop.js
create public/javascripts/effects.js
create public/javascripts/prototype.js
create public/javascripts/rails.js
create script
create script/rails
create test
create test/fixtures
create test/functional
create test/integration
create test/performance/browsing_test.rb
create test/test_helper.rb
create test/unit
create tmp
create tmp/sessions
create tmp/sockets
create tmp/cache
create tmp/pids
create vendor/plugins
create vendor/plugins/.gitkeep
Tonys-Macbook:rails-class tony$
```

# Prepare The App

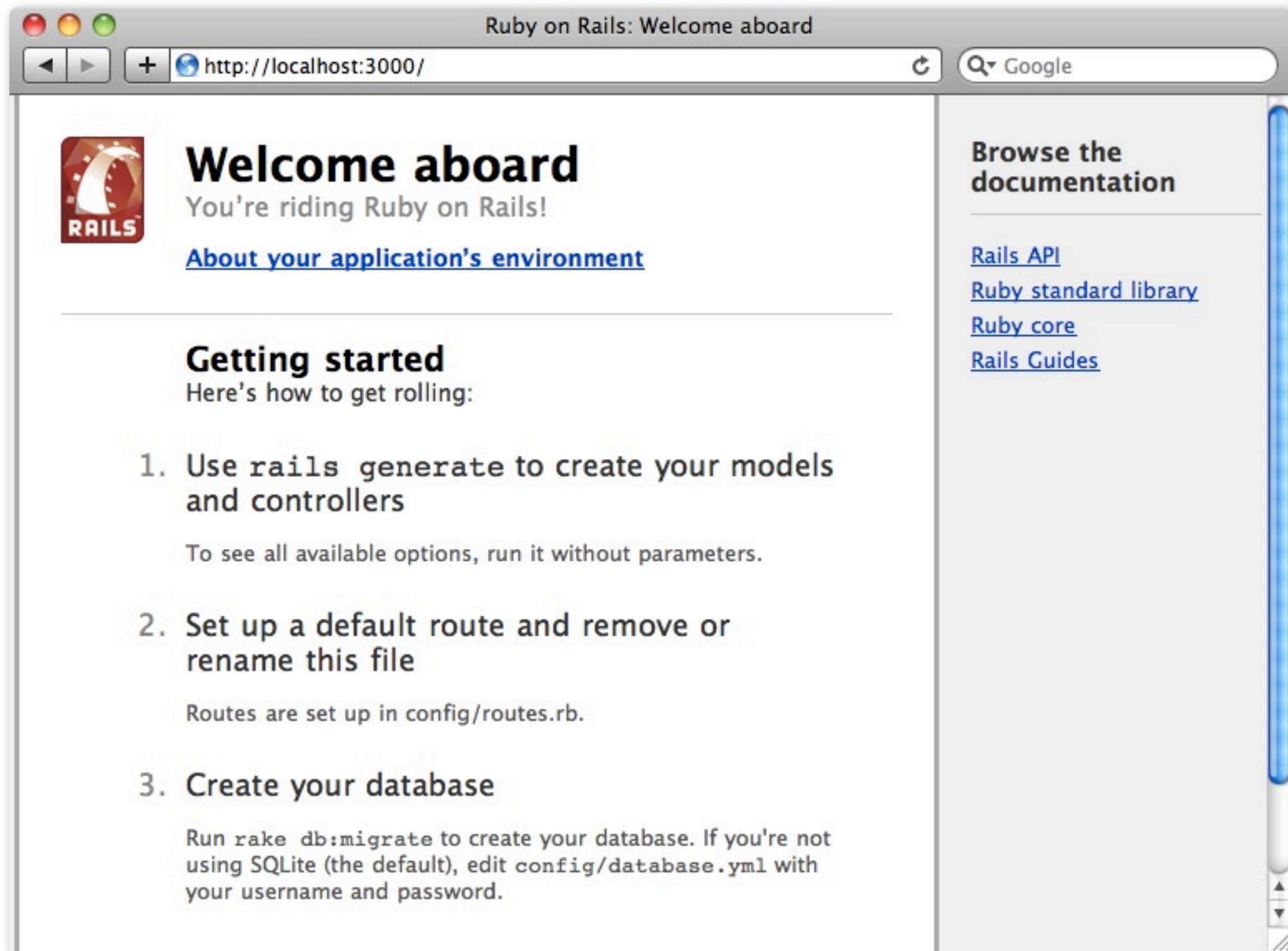
- `cd blog`
- Install dependencies:
  - `bundle install`
- Create the database:
  - `rake db:create`

```
Tonys-Macbook:blog tony$ bundle install
Using rake (0.8.7)
Using abstract (1.0.0)
Using activesupport (3.0.6)
Using builder (2.1.2)
Using i18n (0.5.0)
Using activemodel (3.0.6)
Using erubis (2.6.6)
Using rack (1.2.3)
Using rack-mount (0.6.14)
Using rack-test (0.5.7)
Using tzinfo (0.3.27)
Using actionpack (3.0.6)
Using mime-types (1.16)
Using polyglot (0.3.1)
Using treetop (1.4.9)
Using mail (2.2.19)
Using actionmailer (3.0.6)
Using arel (2.0.10)
Using activerecord (3.0.6)
Using activeresource (3.0.6)
Using bundler (1.0.10)
Using thor (0.14.6)
Using railties (3.0.6)
Using rails (3.0.6)
Using sqlite3 (1.3.3)
Your bundle is complete! Use `bundle show [gemname]` to see where a bundled gem is installed.
Tonys-Macbook:blog tony$ rake db:create
(in /Users/tony/Sites/rails-class/blog)
Tonys-Macbook:blog tony$
```



# View The App

- Start the development server:
  - rails server
- Open your browser:
  - <http://localhost:3000>



# Welcome aboard

You're riding Ruby on Rails!

[About your application's environment](#)

## Getting started

Here's how to get rolling:

1. Use `rails generate` to create your models and controllers

To see all available options, run it without parameters.

2. Set up a default route and remove or rename this file

Routes are set up in `config/routes.rb`.

3. Create your database

Run `rake db:migrate` to create your database. If you're not using SQLite (the default), edit `config/database.yml` with your username and password.

## Browse the documentation

[Rails API](#)

[Ruby standard library](#)

[Ruby core](#)

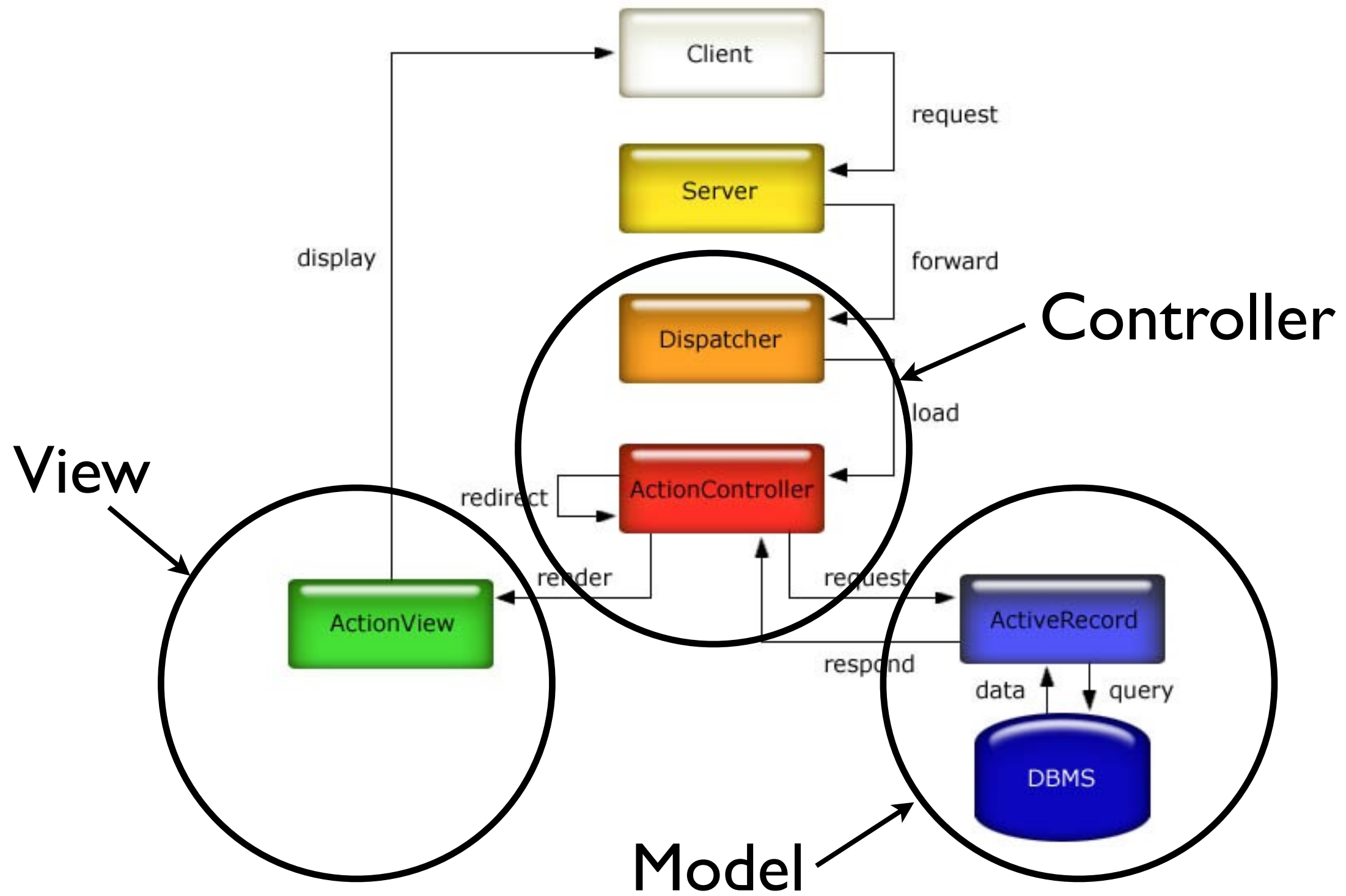
[Rails Guides](#)

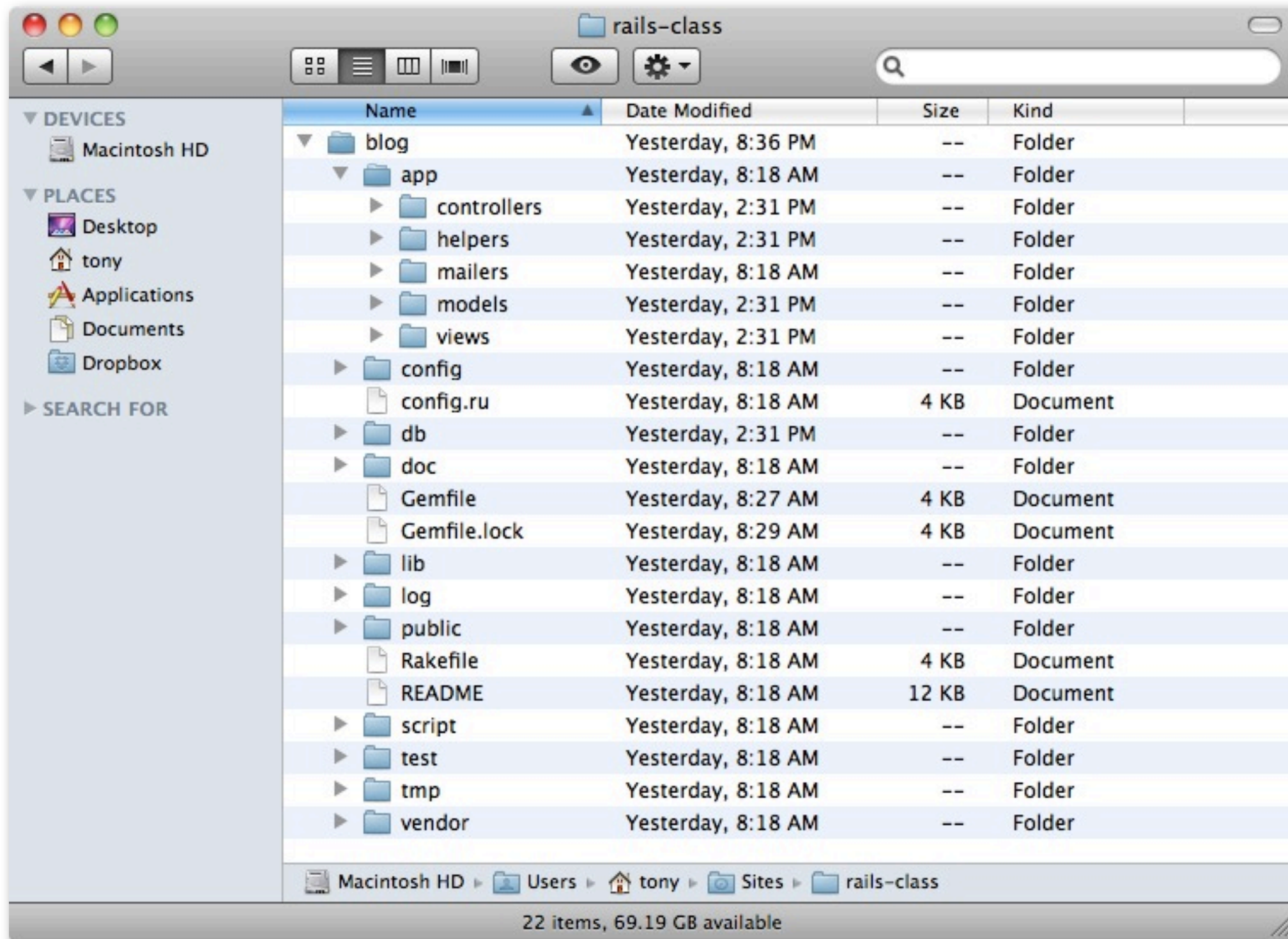
# Rails Philosophy

- Convention over configuration
  - Follow the conventions and there will be very little configuration.
- DRY - Don't Repeat Yourself
  - Specifying the same thing in more than one place leads to errors.

# Rails Architecture

- Model
  - Data and rules for manipulating it.
- View
  - The User Interface
- Controller
  - Glue that ties the UI to the Data.

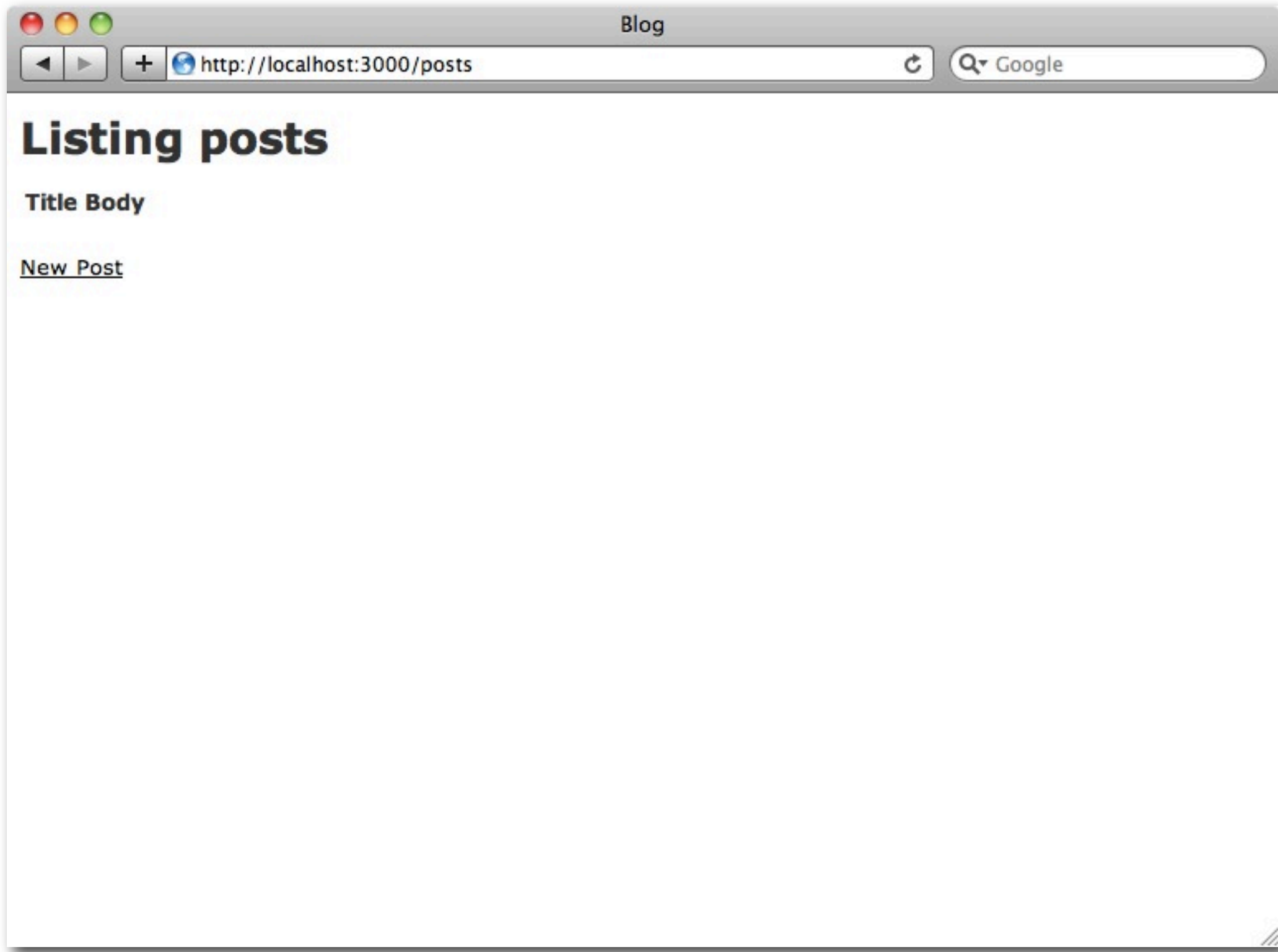






# Add Posts

- Press Ctrl+C to stop the server
  - rails generate scaffold Post title:string body:text
  - rake db:migrate
- <http://localhost:3000/posts>





# Home Work

It's good for you. I promise...

# Learn A Little Ruby

- *Learn to Program* by Chris Pine
  - <http://pine.fm/LearnToProgram>
- *Beginning Ruby* by Peter Cooper
  - <http://no.gd/begruby2.pdf>
- Ruby Koans
  - <http://rubykoans.com>

# Local Community

- Austin on Rails - 4th Tuesdays
  - <http://austinsonrails.org>
- Austin.rb - 3rd Thursdays
  - <http://austinrb.org>
- Lone Star Ruby Conf. - Aug 11-13, 2011
  - <http://www.lonestarrubyconf.com>

# Online Community

- Riding Rails
  - <http://weblog.rubyonrails.org>
- Ruby Inside
  - <http://www.rubyinside.com>
- Ruby Flow
  - <http://www.rubyflow.com>

# GitHub

- <https://github.com>
- Social Coding
- Create a free account
- Read the GitHub Bootcamp

# Heroku

- <http://www.heroku.com>
- Simple Rails App Deployment
- Create a free account
- Look over the Getting Started info