

Full Stack Notes

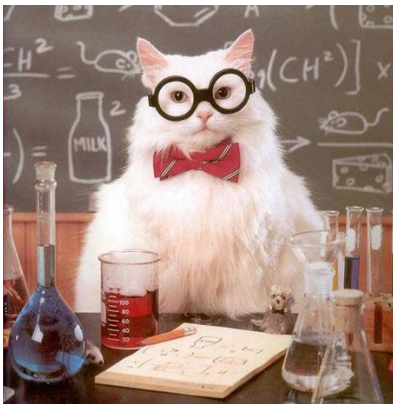
Course Introduction

You are an experiment and so is this course. :)

Table of Contents

- 1 [Full Stack Web Development](#)
- 2 [Why Ruby?](#)
- 3 [Ruby Design Principles](#)
- 4 [Why Rails?](#)
- 5 [Who's Using Rails?](#)
- 6 [Education, Learning and Responsibility](#)

Full Stack Web Development



The goal of this course is to expose you to:

- **Ruby** Programming Language
- **Rails** Web Programming **Framework**
- **Model-View-Controller** Design Pattern
- **RESTful** Routing Convention
- Windows Subsystem for **Linux**
- **Git Source Control Tool**
- **API Consumption** and **CSV Importing**
- Web **Server Configuration & Deployment**
- **Pair** Programming
- **CSS** Frameworks & **Tooling**
- **Credit Card Processing**
- Containerization (**Docker**)

- Moar!

Why Ruby?



- **Dynamic interpreted** languages are gaining popularity (Perl, Python, Javascript, Lua, &c).
- Ruby is **free** (with a very open license for program distribution)
- Strong programming **community** (good tutorials, example code, forums, &c)
- Optimized for "Programming Happiness"

If you've got 30 minutes, why don't you [give Ruby a try in your browser](#).

Ruby Design Principles

- Principle of Least Surprise

- Principle of **Succinctness** (a.k.a. Principle of Least Effort)
- Write more understandable code in **less lines**
- Less code means **less bugs**
- The **quicker** we program, the more we accomplish

RESOURCES

- [Ruby - A programmers best friend](#)
- [Ruby @ Wikipedia](#)
- A brief discussion of [The Principle of Least Surprise](#)

Why Rails?



- Model View Controller (**MVC**) is becoming the **defacto** standard for coding web apps.
- Rails code is very DRY (With Rails you **Don't Repeat Yourself**)

- **Convention** over Configuration (Sane defaults but still **customizable**)
- **Open Source** with over 4,500 contributors.

Careful! The Rails framework is 'opinionated' meaning that it tries to steer you to code in the 'Rails Way'. If your program becomes overly complex or bogged-down in details, you have **drifted from** the 'Rails Way'.

RESOURCES

- [Ruby on Rails Official Website](https://rubyonrails.org/)

Who's Using Rails?



These popular sites were all built on Rails:

- [Shopify](#)
- [Airbnb](#)
- [Twitch](#)
- [SoundCloud](#)
- [GitHub](#)
- [GitLab](#)
- [Hulu](#)

- [Basecamp](#)
- [Square](#)
- [Fun or Die](#)
- [Dribbble](#)
- [Strava](#)
- [Groupon](#)
- [KickStarter](#)
- [Indiegogo](#)
- [Urban Dictionary](#)
- [Genius](#)
- [Instacart](#)
- [Bloomberg](#)
- [Codecademy](#)

Education, Learning and Responsibility



*“Education is an admirable thing, but it is well to remember **from time to time** that nothing that is worth knowing can be taught.”* – Oscar Wilde

Understanding Requires Action

If you find yourself falling behind in your understanding of the course material *please* contact me, or ask a classmate for help. Remember that **you reap what you sow**. If you put minimal effort into this course you will get very little out of it. This means:

- Reading all assigned reading materials.
- Completing (and understanding) all challenges.
- Mastering the Rails / Ruby documentation.
- Typing out Code (No Copypasta)
- Experimentation is better than Stack Overflow and Google-coding
- Permission for Failure / Learning from Mistakes

RESOURCES

- [The Secret of Being a Great Apprentice](#)
-