The Ruby Programming Language

Reference Notes

Travis Craft



Table of Contents

Introduction 2

Ruby Overview 2

Conditionals 2

Appendix 3

# Introduction

I’ve started learning a little bit about the Ruby programming language and wanted a place to store some of the interesting things I’m learning about. This isn’t exhaustive or anything but here I’ll be making some notes about the specifics of the language. Things that stood out to me as being different than I’m use to in other languages, such as C#, or that I felt I should pay attention to and try to remember. Such as the different types of conditional statements with examples I can reference later. This might even help someone else some day. Enjoy!

# Ruby Overview

The Ruby programming language is a dynamic and strongly typed language. Methods are also pass-by-value. So passing in whole objects to a method and expecting changes to its properties within a method to be reflected in the object once the method completes is wrong. You’ll need to create methods to operate on the object (namely class and instance methods). Which really just means using object-oriented programming techniques. ☺ Sounds good.

It’s dynamic in that it is complied in some cases, and interpreted in others. Depending on the implementation and flavor.

[Stack Overflow Question regarding interpreted versus compiled Ruby](http://stackoverflow.com/questions/7284179/is-ruby-a-scripting-language-or-an-interpreted-language)

# Conditionals

Ruby has several different ways to write Boolean expressions. Aside from the regular if else structures Ruby includes the unless conditional.

[Informational page on the unless conditional](http://www.railstips.org/blog/archives/2008/12/01/unless-the-abused-ruby-conditional/)

# Appendix

Looking to install Ruby or learn more about the language? Here’s a list of some helpful resources:

* Derek Banas is a YouTuber who produces many programming tutorials. I myself used his Ruby tutorial as I created these notes. You can find his video at the following URL: <https://www.youtube.com/watch?v=Dji9ALCgfpM>
* The official Ruby website. You’ll find a lot of great information about installing the language and more information on the different aspects of why many people love it. <https://www.ruby-lang.org/en/>