



Interwoven Threads

Scatter, learn things, and regroup

Kai kai@42.us.org

Summary: Keep a friend at your side, and study in the path of inspired ones who have given their knowledge out for free.

Contents

I	Roads	2
I.1	Ruby	2
I.2	Python	2
II	Checklist	4
III	Choosing	6
IV	Advanced	7
V	Guidelines	8
VI	TL;DR	9

Chapter I

Roads

Choose one from the list of freely offered, noncommercial Intro to Programming courses that follows.

Skim through them with your partner and choose one that seems appealing to both of you.

You'll complete as much as you need of your chosen class to understand the concepts we outlined above. Later, it may be a comforting reference manual for you.

I.1 Ruby

- Chris Pine's Learn to Program: <https://pine.fm/LearnToProgram/>
- Bastards' Book of Ruby, sections from "Style, Conventions, and Debugging" to "File input/output": <http://ruby.bastardsbook.com/toc/> (Don't worry about the tweet fetching scripts, they no longer work with Twitter's new API.)
- Test-first Ruby: http://testfirst.org/learn_ruby
- Learn Ruby the Hard Way: <https://learnrubythehardway.org/book/>

I.2 Python

- MIT Gentle Introduction to Programming using Python: <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-189-a-gentle-introduction-lectures/>
- IntroPython.org: http://introtopython.org/hello_world.html
- Google's open Python class: <https://developers.google.com/edu/python/>
- A Byte of Python: <https://python.swaroopch.com/>



Be aware, some of them may be written for an older version of the programming language. If that's the case, check the appendix for instructions on using `rbenv` and `pyenv` to switch the version on your computer.

Chapter II

Checklist

After learning the method of peer corrections, meeting your terminal and installing your text editor, it's time to take a hunting party into the wild plains of free knowledge.

Find a partner to anchor you on your journey. An intellectual peer. Your mission together is to learn the ins and outs of a programming language: either Ruby or Python. You need to learn just enough to have a sense of how it all fits together, and enough to pass the "Hacker, Baby" challenge which is next on the project tree.

Make sure that you have practice with these topics by the time you choose to move on:

- Variables: how to declare them and set their contents
- Common variable types such as strings, ints, doubles, and floats
- Printing output and using string interpolation to include variables
- How to add comments to your code
- Boolean variables and (in Ruby) the concept of `Nil` or `None`
- If, If else, Else
- Conditional expressions such as `==`, `>`, and `!=`
- Taking input through `ARGV` and also from the user after printing a prompt
- Arrays (in Ruby) or Lists (in Python)
- Hashes (in Ruby) or Dicts (in Python)
- How to write reusable **methods** or **functions** or that take in a variable, perform an operation on it, and return the result.



The next project on the path after this one is Hacker, Baby. It's a project based test to see if you have mastered the programming basics.

If you can't pass Hacker, Baby right away, start by working with your partner through one of the tutorials listed on the next page.

Chapter III

Choosing

Which one should I take?

If you do not know either Ruby or Python, you may not have anything to base your preference on.

Programmers love to argue about the superiority of their favorite tools but the truth is that both of these are popular, useful, modern programming languages that have similar functions and are easy to learn!

You can research opinions online if you would like to read something about their characteristics.

If you have no prejudice, I arbitrarily recommend that you go with Ruby =)

It's what Intra is written in and thus it is what we love.

Chapter IV

Advanced

What's that, kid? Not your first rodeo??

Go ahead, solve Hacker, Baby and open that gate to projects ahead.

You can come back and earn your bragging rights properly by finishing one of these detailed classes:

- Ruby Koans: <http://rubykoans.com/>
- Python Koans: <http://python-intro.readthedocs.io/en/latest/koans.html>
- Automate the Boring Stuff with Python: <http://automatetheboringstuff.com/>

Or - find a Creative Commons licensed class for some other language and learn about all the basic concepts in that language, too. If you are advanced enough to skip the intro to Python or Ruby classes you can get XP on this project by turning in your work learning an additional language.

Chapter V

Guidelines

- Work with your partner closely, and consider picking someone who you can chat with outside of class to make progress faster!
- This project will be graded by your peers. You will need to explain to them what you learned about each of the coding concepts listed above.
- You should turn in all code snippets that you wrote while following the tutorial and while playing around to your Vogsphere repository.
- You can skip it and turn in Hacker, Baby right away if you prefer.

Chapter VI

TL;DR

- You need a partner.
- You need to pick one option from the list in the "Roads" chapter.
- You need to follow that tutorial and push all code written to your Git repo until you understand all concepts listed in "Goals" chapter.
- You need to have the project graded through peer corrections in order to get credit.
It's a fun way to compare the craziest tricks that you learned!

