**\* Self-taught Curriculum**

- Date: 2022 July~

1. Motivation/Mindset

1) Tips/Advice from Succeeded Youtubers

2) Googling(Blogs, Comments, etc)

2.Python

-> Variables, Data Types, Flow Control, List, Dictionary, Function, OOP, etc

1) Dip into the unknown programming world

* Python for Everybody - Full University Python Course (by. Charles Severance)

2) With built confidence, Thoroughly read and follow assignments

* Python Crash Course: A Hands-On, Project-Based Introduction to Programming

3. C

→ Data Types, Flow Control, Arrays, Strings, Pointers, Memory Allocation, Function, etc

1) Learn very basic concepts used in C & Understand what’s going on under the hood of a compiler and memory

* CS50: Introduction to Computer Science

4.Basic Computer Science

1) Web(client/server, network, IP address, TCP/IP, OSI model, etc)

* TheOdinProject
* Introduction to Networking: How the Internet Works (by. Charles Severance)

2) Hardware(Binary, CPU, RAM, Hard Disk, etc)

* CS50's Understanding Technology 2017
* <Youtube> Crash Course: Computer Science, Computerphile

5. Command-line interface

* The world of operating systems, What is Linux?, Shells and Bash
* pwd, ls, cd, mkdir, touch, rm, cp, cat, piping, etc

6. Discrete mathematics

\* Online university located in Seoul (Course & Exam based)

→ Propositional logic, Proofs, Set theory, Matrices and Matrix Operations, Permutation&Combination, Probability, Graph Theory, etc

 2023

**<Jan>**

1. Javascript - TheOdinProject

→ Data types, Numbers, Variables, Conditionals, Execution Context, Call Stack, Function expression vs. Function declaration, Chrome Developer Tool, etc

2. Done with 2 books

1) Python Crash Course: A Hands-On, Project-Based Introduction to Programming

→ Part 1: Basics Files and Exceptions, Refactoring, Unit test, etc

2) Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People

→ It helped me a lot to visualize various algorithm methods and to get interested in more advanced topics such as B-Tree, LSM Tree, KNN algorithm, Linear programming, SHA algorithm, etc.

3. Begin 3 online courses – https://academy.zerotomastery.io/

1) Complete Web Developer

\* Brush up on HTML/CSS/Flexbox and learn Bootstrap/Grid to build a simple website

→ Stop the course before moving to a Javascript part and begin **Complete SQL + Databases** course. Just want to focus on building some Python projects and learning Back-end development using Django

2) Complete Python Developer

\* Brush up on basic concepts quickly and learn more things such as

- Functional programming(Lambda, Map, Filter, Reduce/Comprehensions)

- Developer Environment(VS Code, PyCharm)

- OOP(4 pillars, Dunder methods, Multiple Inheritance(MRO), etc)

- Decorator and Generator

- Modules(random, sys, collections, time, array)

3) Master the Coding interview: Data Structures + Algorithms

\* Big O

**<Feb>**

1. ZeroToMastery

1) Complete Python Developer (→ Finished)

\* Additionally incorporate the book: <Python Tricks\_A Buffet of Awesome Python Features> to understand the behaviors of Python more closely to reach an intermediate level

\* Scripting with Python : Image processing, PDF merge and watermark, Sending Emails/SMS, Password checker, Twitter Bot using Tweepy API

\* Machine Learning + Data Science: Follow very basic code and understand the tools

2) Complete SQL + Databases Bootcamp: SQL Fundamentals, Environment Setup, SQL Intermediate(Join, Group by, Window function, etc)

3) Master the Coding interview: Data Structures + Algorithms

\* Arrays, Hash Tables, Linked Lists, Stacks+Queues

4) Complete Machine Learning and Data Science

\* Machine learning 101, Environment Setup, Pandas, Numpy

2. Begin 3 online courses

- Google Data Analytics Professional Certificate(Focused)

- Algebra(Less Focused)

- Statistics(Less Focused)

3. LeetCode: Practice & Review others’ codes to learn efficient algorithmic approach

**<Mar>**

Continuing..!