**코딩 입문 매뉴얼**

2022. 09. 08

작성자: 최동훈(201821505 소프트웨어학과)

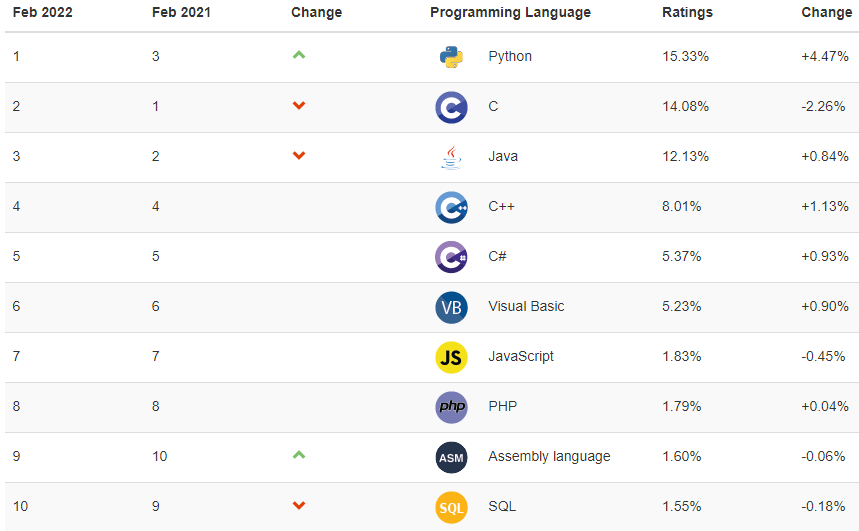
**1. 왜 코딩을 배우는가?**

SQL, Java, Python, C, C++, javascript 등 다양한 프로그래밍 언어 중에 무엇을 선택해야 할 것인가? 프로그래밍 입문자들이 가장 먼저 주변의 개발자들 혹은 커뮤니티에 던지는 질문이다. 하지만 모든 것을 해보는 것을 추천한다. 자신이 코딩을 배우고 싶은 목적을 명확히 하고 얼마 만큼의 시간과 비용을 본인이 투자할 것인지 계산해보면 정보 조달을 위한 수단을 적절히 설정하는 데도 도움이 된다. 프로그래밍을 전공했다고 하더라도, 대학에서 배우는 코딩이 시대에 뒤떨어지고 있다는 말이 자꾸만 언급되다 보니 코딩 관련 온/오프라인 강좌에 대한 관심이 대폭 증가하고 관련 책도 쏟아져 나오고 있는 상황. 다양한 선택지 사이에서 어떤 것을 선택해야 할지는 코딩을 배우고 싶은 이유를 명확히 한다면 답을 조금 더 쉽게 찾을 수 있을 것이다.

**2. c언어를 배우는 이유와 알맞은 프로그래밍 언어 선택하기**

c언어는 unix운영체제 작성을 위해 설계된 시스템 프로그래밍 언어이다. c언어는 수천 가지의 프로그래밍 언어 중 시초이자 모태 언어로 불리며 c언어의 단점을 보완하여 개발된 것이 python, java 등이다. 최소한 c언어를 잘해두면 컴퓨터 관련 전공(자료구조, 알고리즘, 디지털 회로, 시스템프로그래밍, 운영체제…)를 수행하는데 큰 지장이 안 생긴다.

자신이 코딩을 배우려는 목적을 설정한 후, 이제 배워야 할 프로그래밍 언어를 결정해야 한다. 최고의 프로그래밍 언어가 뭘까? 라는 방식으로 접근하기보다는 자신의 목적에 맞는 단 한 가지의 언어, 그것을 활용해 단기적인 목표를 달성하는 것이 먼저일 것이다. 단기적인 목표를 달성한 후 다른 언어를 배우게 되면 자신만의 노하우가 생겨 자연스레 프로그래밍에 익숙해지는 자신을 발견하게 될 것이다.



**-> 2022년 기준 많이 쓰는 언어**

**3. 컴퓨터 프로그래밍 및 실습에서 좋은 성적을 받으려면..?**

강의노트에서 대부분의 시험문제가 나오며 기본 개념이라 상당히 중요하다. 마르고 닳을 때까지 내용을 외우는 것이 중요하며 과제를 수행할 때 이를 적용하는 연습을 자주 해야 한다. 추가적으로 백준, 코드잇, 프로그래머스 등의 사이트를 통해 문제를 연습해 볼 수 있다.

<https://www.acmicpc.net/workbook/top>

-> 백준 문제집

<https://leetcode.com/>

-> LEETCODE

<https://www.codeit.kr/challenges>

-> Codeit 챌린지

<https://school.programmers.co.kr/learn/courses/30/lessons/43165>

-> 프로그래머스 연습문제

<https://solved.ac/>

* 백준 문제 rank별 정리

**4. 온라인 강의 수강해보기**

비용과 시간의 제약이 있거나, 다양한 강의를 접해보고 싶은 이들에게는 온라인 강의를 수강하는 것도 좋은 방법이 된다. 하나의 언어를 두고, 다양한 온라인 강의들이 존재하기 때문에 무엇이 좋은지를 선택하는 일은 쉽지 않을 것이지만 온라인 강의 수강을 결정할 때 최소한으로 고려해보아야 할 점은 다음과 같다. 참고로 유튜브에도 양질의 콘텐츠를 찾아볼 수 있다. 단순히 코드를 따라치지 말고 이해하고 연습하는 것이 중요하다.

<http://www.kocw.net/home/index.do>

* kocw : 전공 인강 제공

<https://ose.ajou.ac.kr/member/rl/login/loginForm.do>

* 아주대학교 녹화강의

<https://www.udemy.com/courses/search/?src=ukw&q=maxmillian>

* Udemy: 전세계 인터넷 강의(유료)

<https://coding-x.com/class>

* 코딩엑스 강의(머신러닝 강의 유료)

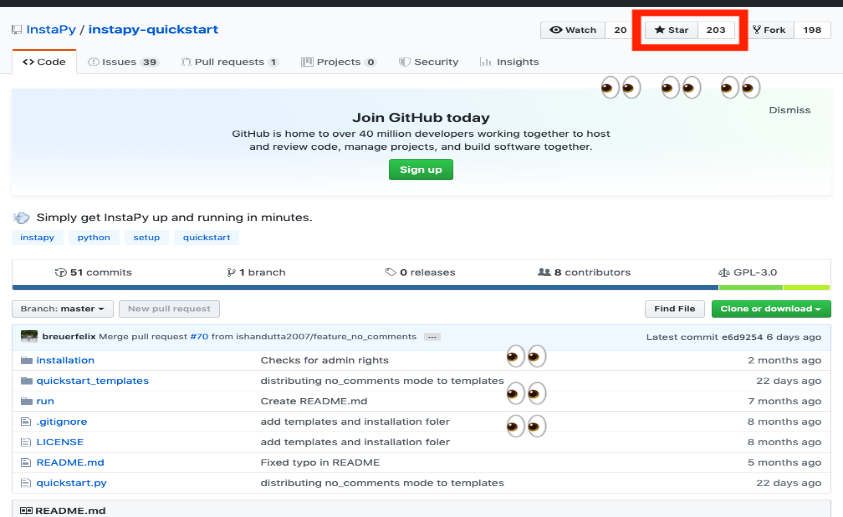
**5. 서적 구매 및 도서관 활용**

텍스트이(가) 표시된 사진

자동 생성된 설명

아무리 인터넷이 정보의 바다라고 할지라도 정보 전달이 너무 산발적이기 때문에 나의 것으로 만들기란 꽤 어려운 지점이 있다. 이보다 포괄적이고 압축적으로 프로그래밍에 대해 알려줄 수단은 책이라고 자부할 수 있는데 여기서 또 고민은 시작된다. 서점에 가서 관심이 있는 분야에 대해 책을 읽는 것을 추천한다. 커뮤니티 등에서 추천해 주는 책이 좋은 경우가 다반사다.

**5. 타인의 코드 참고하고 연습하기**

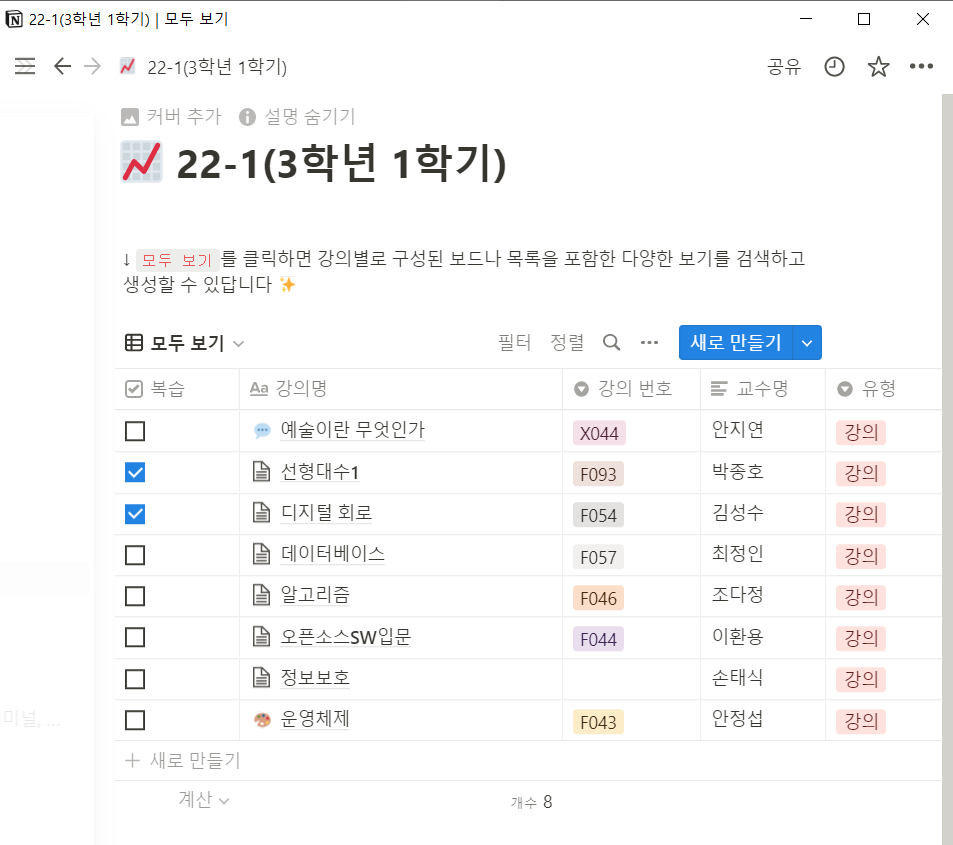
****

 구글링을 잘 하는 것이 중요하다. 이를 통해 깃허브, stackoverflow 등의 사이트에서 잘 짠 코드를 보고 연습하는 것이 중요하다.

프로그래밍을 공부할 때 자신의 것 외에 다른 사람의 코드를 대조해 보며 리딩하는 것도 큰 도움이 된다. 인터넷은 오픈 소스 코드가 넘쳐나는 천국이니 (대게 깃헙의 Star가 많은 라이브러리를 둘러보며) 좋은 코드들을 읽어보기에 용이하다. 이 과정에서 ‘타인의 알고리즘을 이해하면서 스스로 성장하는 기분을 느꼈다’는 많은 개발자의 말은 빈말이 아닐 것이다.

**6. 블로그나 위키에 자신의 공부 과정 쌓기**

직접 시도한 모든 것, 생각의 흐름 등을 하나의 위키나 블로그에 모아 보자. 세상 사람들이 내가 무슨 일(공부)을 했는지 알 수 있게 하고, 긴 글을 작성하는 훈련을 할 수 있음과 동시에 나만의 지식도 논리적으로 빌딩할 수 있다. 보는 이가 없어도 이 활동을 지속적으로 하다보면 지식의 습득 속도 뿐 아니라 밀도까지 높아져 프로그래밍 실력을 체화시키는 데 큰 도움이 된다. 활동을 하다보면 유입되는 다양한 사용자들과 의사소통을 하며 다양한 의견의 장을 열 수 있다는 건 덤! 필자는 노션(notion)을 활용하여 공부한 내용을 메모한다.



**7. 최대한 고민해보고 질문하자.**

질문을 하는 것은 좋으나 대부분의 질문은 코드의 사소한 실수나 개념의 불충분한 이해 때문에 발생하는 경우이다. 따라서 궁금한 것에 대해 최대한 고민을 하고 질문을 하도록 한다.