Python

이채영

Python

"약 6년 전인 1989년 12월, 크리스마스를 전후하여 취미로 만들어 볼 프로그래밍 프로젝트를 찾고 있었죠. 그 때 사무실은 잠겨있어지만, 집에 컴퓨터가 있었고, 뭐 특별히 할 일도 없었죠. 그래서 그 때 당시 한동안 생각하고 있었던 새 스크립트 언어에 대한 인터프리터를 만들어 보기로 했죠. 유닉스/C 해커들에게 어필할 수 있는, ABC 언어로부터 파생된 언어말이죠. 나는 그 프로젝트명으로 Python이라는 이름을 선택했는데, 그 당시 약간은 불손한 기분이 들어서이기도 했고, 또한 당시 Monty Python's Flying Circus(BBC 코메디)에 열성팬이기도 하여..."

- 1996, Guido

Python 특징

- **오픈 소스로 무료로 제공된다.**
  - 다양한 라이브러리를 지원한다.
- **매우 간결하며 명시적이다.**
  - 높은 생산성을 가진다.
  - 가독성이 좋다.
  - 문법이 쉬워 빠르게 배울 수 있다.
  - 문법이 매우 엄격하다.
- **플랫폼 독립적인 언어**
  - 운영체제에 종속되지 않는다.
  - Python 바이트 코드를 생성하여 소스코드 없이도 다른 컴퓨터에서 수행된다.

python은 어떤 언어인가?1

대화 기능의 인터프리터 언어

- 객체지향 기능을 지원하는 대화형 인터프리터 언어
- 사용이 쉽고, 컴파일-실행-에러 수정이 아닌 작성-테스트

동적인 데이터 타입 결정 지원

- 동적으로 데이터 타입을 결정하므로 데이터 타입에 관계없는 일반화된 코드 작성 가능

플랫폼 독립적 언어

- 리눅스, 유닉스, 윈도 등 대부분의 운영 체제에서 동작.
- 플랫폼 독립적이며, 컴파일하지 않고도(내부적으로 자동 수행됨) 동작하기 때문에 사용하기 쉽다.
- 자바와 같이 파이썬 바이트 코드를 생성하므로 소스 코드 없이도 다른 컴퓨터에서 즉시 수행된다.

python은 어떤 언어인가?2

개발 기간 단축에 초점을 둔 언어

- 실행의 효율성보다는 개발 기간 단축에 초점을 둔 언어.
- 실행 속도로 말하자면 어셈블리 언어를 제외하고는 C에 견줄 만한 프로그래밍 언어는 없다. C는 효율적인 코드 생성에 큰 의미를 둔 언어이기 때문이다.
- 반면 파이썬은 개발의 효율성에 무게 중심을 두고 있다.

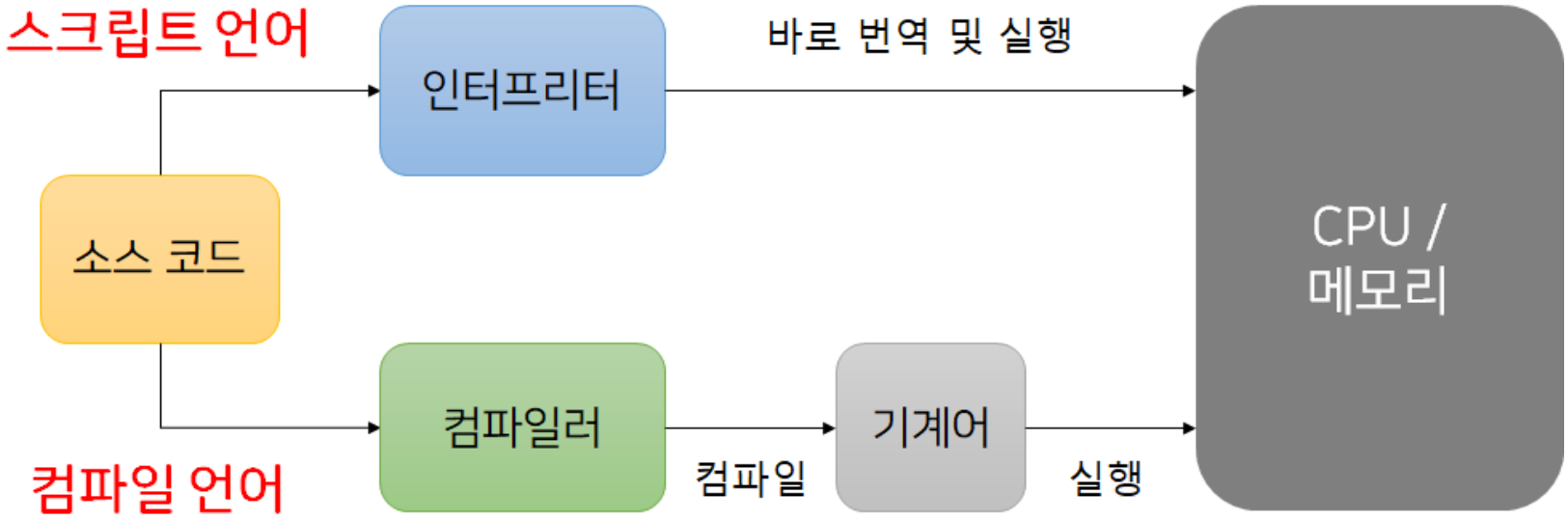
간단하고 쉬운 문법

- 간단한 문법과 깔끔한 구문은 배우기 쉽고, 사용하기 쉽다.
- 객체지향 언어로서 재사용 가능한 코드를 쉽게 만들 수 있다.

고수준의 내장 객체 자료형 제공

- 일반적으로 사용되는 **리스트(List), 사전(Dictionary) 및 문자열(String), 튜플(Tuple)** 등의 자료 구조를 제공한다.
- 이러한 자료형들을 유연하고 쉽게 사용할 수 있다.

컴파일 언어 vs 인터프리터 언어



컴파일 언어

소스 코드를 컴파일 한 후 기계어를 CPU와 메모리를 통해 읽어들인 후, 실행하는 방식으로 동작하는 언어

- 규모가 큰 프로그램은 컴파일 시간이 오래 걸릴 수 있다.
- 컴파일 후에는 모든 소스코드가 기계어로 변환되어 있기 때문에 실행 시간이 빠르다.
- C, C++, Java, C#

인터프리터 언어(스크립트 언어)

소스 코드를 컴파일하지 않고 인터프리터로 소스코드를 한 줄씩 읽어서 바로 실행하는 방식으로 동작하는 언어

** 인터프리터 : 프로그래밍 언어의 소스 코드를 바로 실행하는 컴퓨터 프로그램 또는 환경

- 컴파일을 하지 않고 인터프리터가 직접 한 줄씩 읽어서 실시간으로 실행하기 때문에 컴파일 언어에 비해 속도가 느리다. → 번역과 실행이 동시에
- 빌드 과정 없이 바로 실행이 가능하다.
  → 별도의 실행 파일이 존재하지 않음
  - R, Python, Ruby

인터프리터

- 명령문 단위로 해독하고 실행
  예)통역사:한 문장을 듣고 한 문장을 번역해서 발표하는 방식

(단점) 프로그램 전체의 실행 속도가 느림
- 반복문의 경우 동일한 명령문을 매번 실행, 다시 해독하고 실행

(장점) 개발 단계 단위 모듈 또는 명령문 테스트나 간단한 코드를 삽입한 스크립트 형식 사용에 간편함
- 에러가 발생한 명령문을 즉시 수정해서 다시 실행할 수 있음

컴파일러

- 프로그램 전체를 기계어로 해독한 후 실행
- 해독과 실행이 완전하게 구분
- 프로그램 파일을 처음부터 끝까지 기계어로 해독한 후 그 기계어를 실행

  예) 번역가 : 책 한권을 통째로 번역해서 출판 하는 것

(장점) 인터프리터에 비해 실행 속도가 빠름
(단점) 한 개의 명령문만 잘못되어도 프로그램 전체를 다시 컴파일해야 하기 때문에 효율적인 개발이 어렵고, 스크립트 형식으로 사용불가

대화형 모드 == 인터프리터 모드

- **대화형 모드(interactive mode)**

- 대화 창인 파이썬 셸에서 실시간으로 명령어를 입력하는 모드

- **스크립트,인터프리터 모드(interpreter mode)**

- 파이썬 프로그램 파일을 실행하는 모드

```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021,
15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for
more information.
>>> print('hello python')
hello python
>>>
```

(정리)파이썬의 인터프리터*

- 파이썬은 인터프리터를 사용한다고 하지만 실제로는 "컴파일러 + 인터프리터 방식"을 사용(그러나 인터프리터 형식임)
- 파이썬 소스 코드는 먼저 **바이트 코드(byte code)라는 중간 단계의 코드로** 일괄 변환.
- 바이트 코드는 소스 코드도 아니고 완전한 기계어도 아님. 바이트 코드는 소스 코드보다 기계어로 변환되는 속도가 빠름.
- 소스 코드를 바이트 코드로 일괄 해독해 놓고 그 바이트 코드를 하나씩 가져다 해독, 실행.
- 바이트 코드가 만들어지는 과정까지는 "컴파일 방식"이며, 이후 바이트 코드를 하나씩 해독, 실행하는 과정은 "인터프리터 방식"

통합개발환경(IDE)

- 파이썬으로 코드를 작성할 수 있는 도구

- 텍스트 에디터: 메모장, VS code + 코드 실행기
- 파이썬 IDLE, 아나콘다, 주피터 노트북, 구글 코랩, 파이참 등등
- List of integrated development environments for Python

파이썬
바이트코드

(참고)파이썬 바이트코드 이해하기

- https://towardsdatascience.com/understanding-python-bytecode-e7edaae8734d

- Python은 일반적으로 인터프리터 언어라고 하며, 엄밀하게 말하면 컴파일과 인터프리팅이 결합된 언어.

- 소스 코드를 실행할 때(파일.py확장자), Python은 먼저 이를 바이트 코드로 컴파일.

- 바이트 코드는 소스 코드의 낮은 수준의 플랫폼 독립적 표현이지만 이진 기계 코드가 아니며 대상 시스템에서 직접 실행할 수 없다. 실제로 Python 가상 머신(PVM)이라고 하는 가상 머신에 대한 명령 집합.

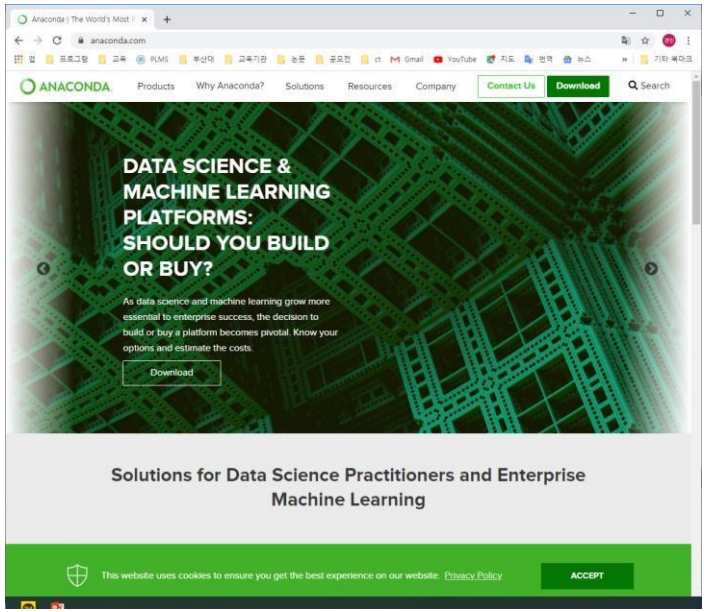
파이썬 바이트코드 이해하기

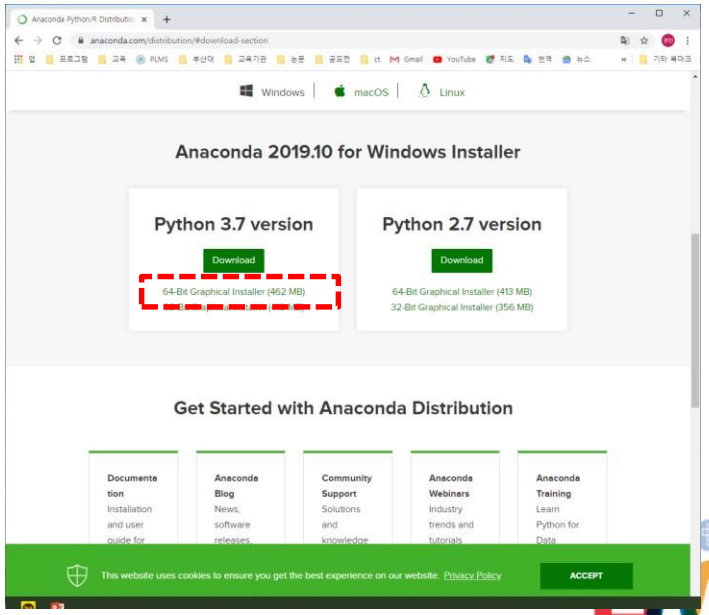
- 컴파일 후 바이트 코드는 실행을 위해 PVM으로 전송. PVM은 바이트코드를 실행하는 인터프리터이며 Python 시스템의 일부.
- 바이트코드는 플랫폼에 독립적이지만 PVM은 대상 시스템에 따라 다름.
- Python 프로그래밍 언어의 기본 구현은 C 프로그래밍 언어로 작성된 CPython.
- CPython은 파이썬 소스 코드를 바이트 코드로 컴파일하고 이 바이트 코드는 CPython 가상 머신에 의해 실행.

실습 환경 구축

Python 환경 설정

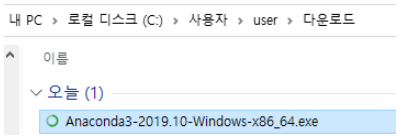
- 아나콘다
  - Python 기본 패키지에 각종 수학/과학 라이브러리들을 같이 패키징해서 배포하는 버전
- 아나콘다 설치
  - https://www.anaconda.com

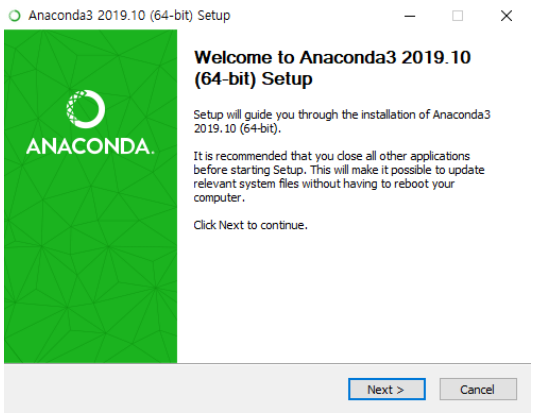


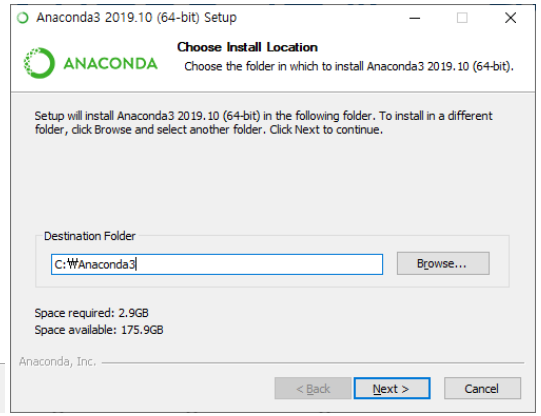


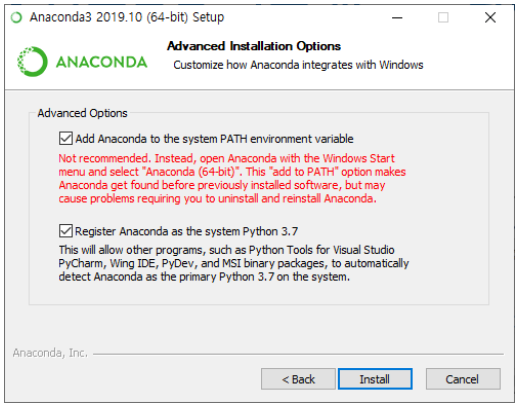
Python 환경 설정

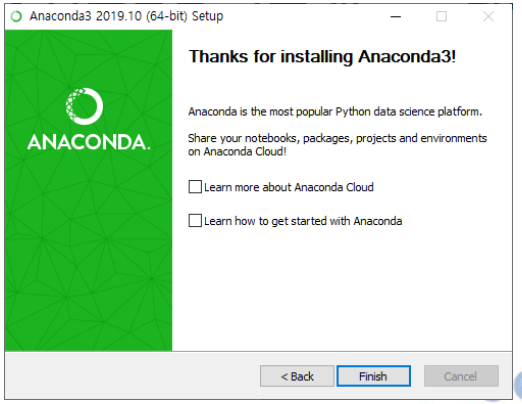
- 다운로드 받은 파일 클릭하여 설치











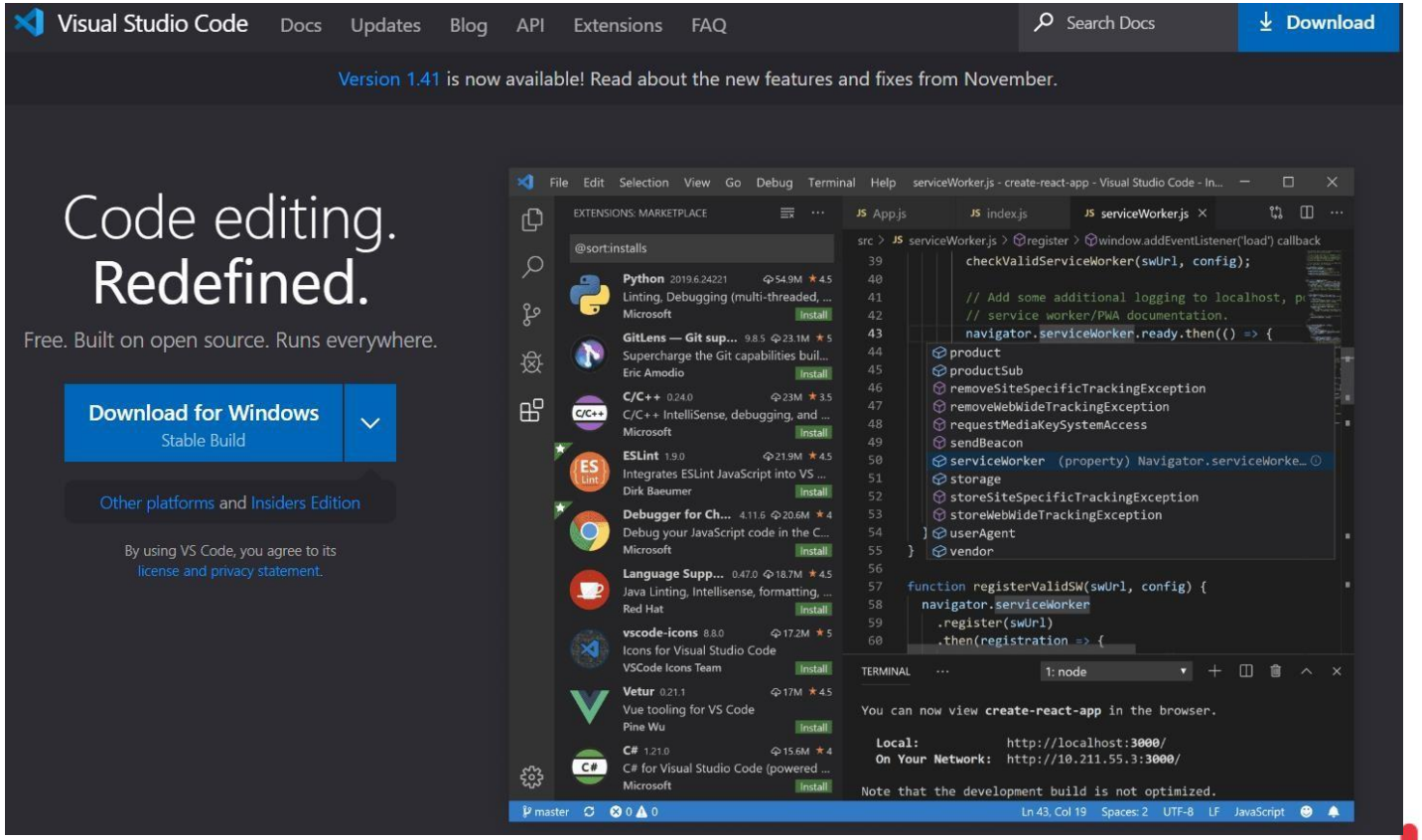
Python 설치확인


모두
앱
문서
설정
웹
더 보기
가장 정확
명령 프롬프트
앱
앱
Anaconda Prompt (Anaconda3)
웹 검색
cmd - 웹 결과 보기
설정 (1)
명령 프롬프트
앱
열기
관리자 권한으로 실행
파일 위치 열기
시작 화면에 고정
작업 표시줄에 고정
cmd

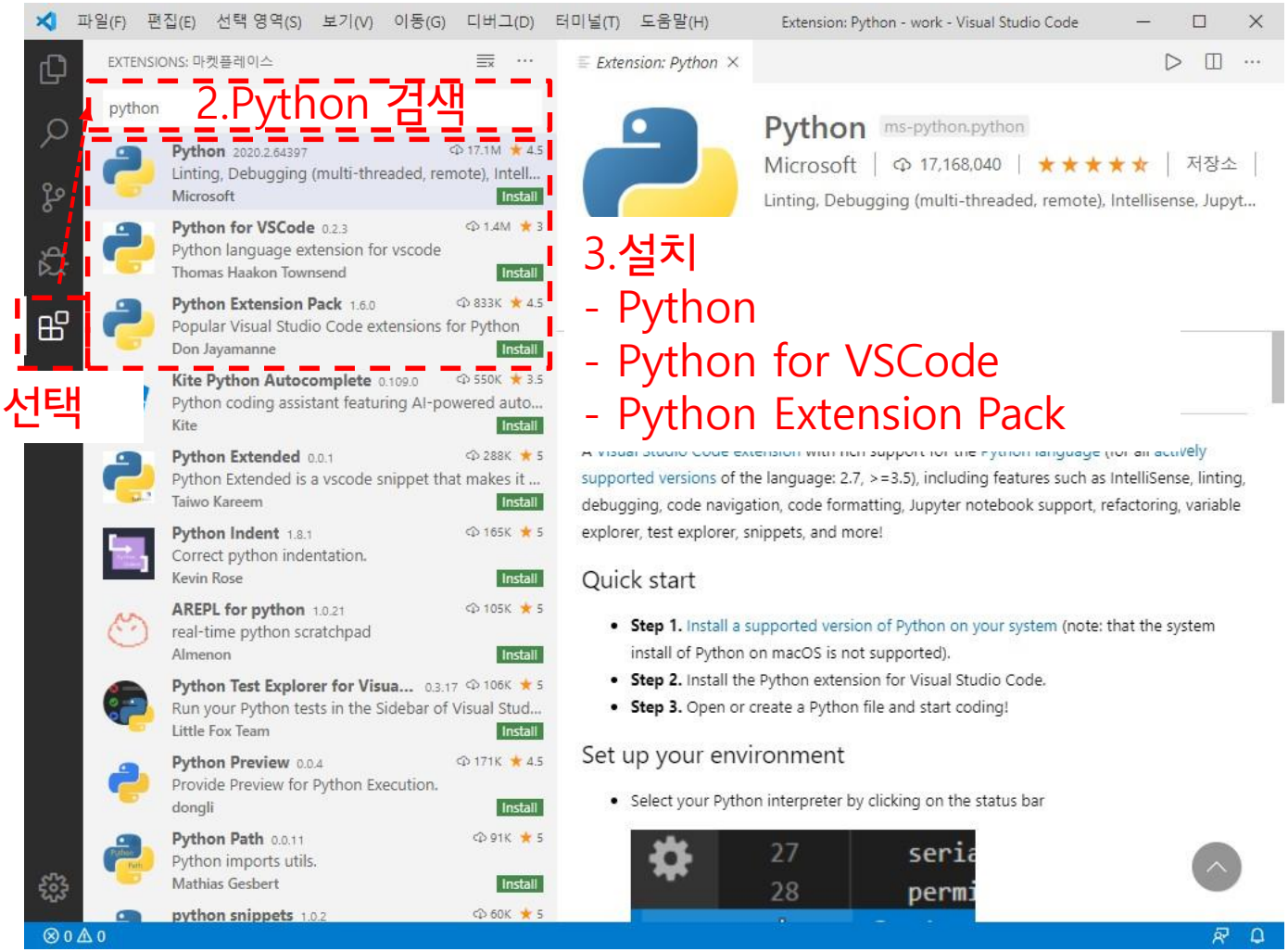

명령 프롬프트
C:₩Users₩user>python -V
Python 3.7.4
C:₩Users₩user>

VS code 설치

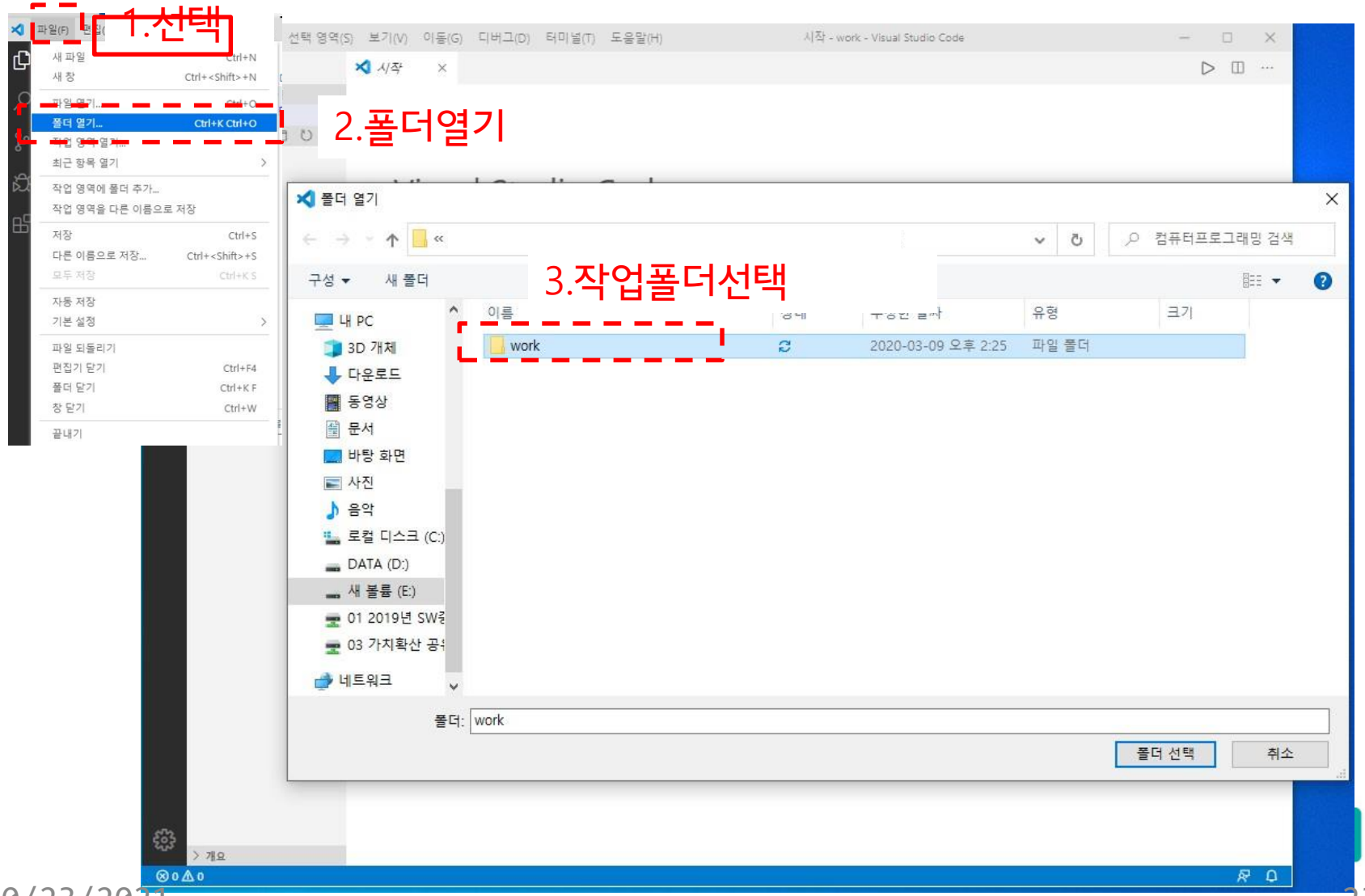
- https://code.visualstudio.com



VS code extention 추가

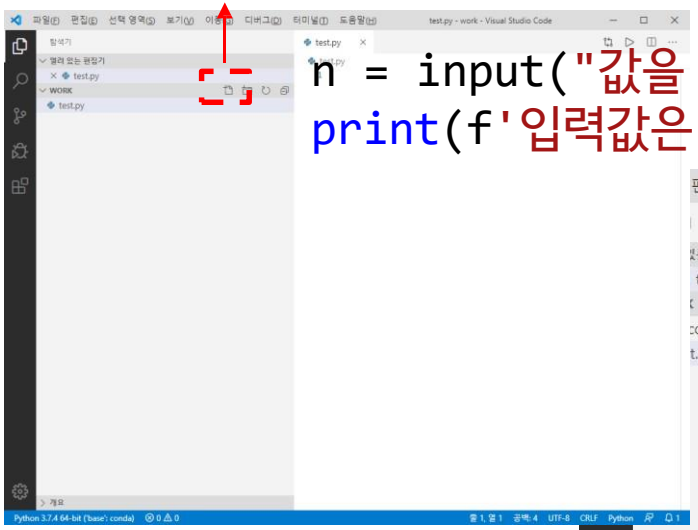


VS code 실행 – 작업 폴더 선택



VS code 실행 – 파일 만들기

1.선택 한 후 test.py파일명입력



2.편집기에 명령입력

3.편집기 오른쪽 마우스 누른 후 Run Python File in Terminal 선택

4.아래쪽 터미널 화면에 10입력 후 결과 확인