



CHAPTER 02

OPENCV와 파이썬

PART 01 영상 처리 개요 및 OPENCV 소개



목차

2.1 OPENCV와 파이썬 개요

2.2 파이썬(PYTHON) 설치 및 사용

2.3 파이참(PYCHARM) 설치

2.4 파이참 환경 설정

2.1 OpenCV와 파이썬 개요

◆ 2.1.1 OpenCV 소개

❖ OpenCV - Open Source Computer Vision Library

- 영상 처리와 컴퓨터 비전 관련 오픈 소스 라이브러리
- 2,500개가 넘는 알고리즘으로 구성
 - ✓ 영상 처리, 컴퓨터 비전, 기계 학습과 관련된 전통적인 알고리즘
 - 얼굴 검출과 인식, 객체 인식, 객체 3D 모델 추출, 스테레오 카메라에서 3D 좌표 생성
 - 고해상도 영상 생성을 위한 이미지 스티칭, 영상 검색, 적목 현상 제거, 안구 운동 추적
 - 4만 7천 이상의 사용자 그룹과 1,800만 번 이상의 다운로드 횟수
- 구글, 야후, 마이크로소프트, 인텔, IBM, 소니, 혼다, 도요다와 같은 대기업부터 Applied Minds, Videosurf 및 Zeitera와 같은 신생 기업들까지 사용
- C, C++, 파이썬(Python), Java, 매트랩 인터페이스 제공
- 윈도우즈, 리눅스, 안드로이드, 맥 OS 등 다양한 운영체제 지원
- MX(Multimedia Extension)와 SSE(streaming SIMD Extensions) 명령어 통해 고속의 알고리즘 구현
- CUDA와 OpenCL 인터페이스 개발

2.1.1 OpenCV 소개

◆ OpenCV - Open Source Computer Vision Library

- ❖ 영상 처리와 컴퓨터 비전 관련 오픈 소스 라이브러리
- ❖ 2,500개가 넘는 알고리즘으로 구성
 - 영상 처리, 컴퓨터 비전, 기계 학습과 관련된 전통적인 알고리즘
 - ✓ 얼굴 검출과 인식, 객체 인식, 객체 3D 모델 추출, 스테레오 카메라에서 3D 좌표 생성
 - ✓ 고해상도 영상 생성을 위한 이미지 스티칭, 영상 검색, 적목 현상 제거, 안구 운동 추적
 - ✓ 4만 7천 이상의 사용자 그룹과 1,800만 번 이상의 다운로드 횟수
- ❖ 구글, 야후, 마이크로소프트, 인텔, IBM, 소니, 혼다, 도요다와 같은 대기업부터 Applied Minds, Videosurf 및 Zeitera와 같은 신생 기업들까지 사용
- ❖ C, C++, 파이썬(Python), Java, 매트랩 인터페이스 제공
- ❖ 윈도우즈, 리눅스, 안드로이드, 맥 OS 등 다양한 운영체제 지원
- ❖ MX(Multimedia Extension)와 SSE(streaming SIMD Extensions) 명령어 통해 고속의 알고리즘 구현
- ❖ CUDA와 OpenCL 인터페이스 개발

2.1.1 OpenCV 소개

〈표 2.1.1〉 OpneCV 버전별 특징

1.0 버전	2.0 버전	2.2 버전
<ul style="list-style-type: none"> • C 언어 기반 API • 구조체 기반 데이터 구조 사용 • 비주얼 스튜디오에서 라이브러리 컴파일 후 사용 • highgui 모듈에서 8비트 PNG, JPEG2000 입출력 지원 • 샘플 예제 파일 추가 (calibrate.cpp, inpaint.cpp, leter_recog. cpp 등) 	<ul style="list-style-type: none"> • C++ 언어 기반 API • 클래스 기반 데이터 구조 도입 • CMake를 이용하여 라이브러리 컴파일 후 사용 가능 • highgui 모듈에서 스테레오 카메라 지원 • 소스 디렉터리 구조 구성 	<ul style="list-style-type: none"> • 템플릿 자료구조 추가 • 기존 5개 라이브러리를 12개의 모듈로 재구성(opencv_core, opencv_imgproc, opencv_highgui, opencv_ml 등) • 안드로이드 지원 가능 • highgui 모듈에서 16비트 LZW-압축 TIFF 지원 • GPU 처리 지원
2.4 버전	3.0 버전	3.4 버전
<ul style="list-style-type: none"> • cv::Algorithm 클래스 도입 • SIFT와 SURF 모듈 유료화 • SIFT 성능 대폭 개선 • 컬러 영상 캐니 에지 수행 	<ul style="list-style-type: none"> • cv::Algorithm 적극 사용 • 1500개 패치 github 제출 • OpenCL을 사용하는 투명 GPU 가속 레이어 도입 • NEON 내장 함수 사용한 OpenCV 함수 가속화 • Python & Java 바인딩 확장 및 Matlab 바인딩 도입 • Python 3.0 지원 향상 • 안드로이드 지원 향상 • 비디오 캡처 및 멀티스레딩 함수 개선 	<ul style="list-style-type: none"> • dnn 모듈 개선 <ul style="list-style-type: none"> - fast R-CNN 지원 - Javascript 바인딩 - OpenCL 가속화 포함 • OpenCL 커널 바이너리에 디스크 캐시 및 수동 로딩 구현 • GSoC 프로젝트 통합으로 백그라운드 감산 알고리즘 구현

2.1.1 OpenCV 소개

4.0 버전	4.1 버전	4.2 버전
<ul style="list-style-type: none">• 1.x 버전 C API 대량 제거• 효과적인 그래픽 기반 영상처리 엔진으로 G-API 모듈 추가• OpenVION 딥러닝 툴킷으로 dnn 모듈 업데이트• 키넥트 퓨전 알고리즘 구현• QR코드 검출기 추가• 효과적인 광류 알고리즘 추가	<ul style="list-style-type: none">• core와 imgproc 모듈 실행 최적화• dnn 모듈 개선<ul style="list-style-type: none">- NN Builder API로 교체- 인텔 Neural ComputerStick2 지원• 안드로이드 미디어 NDK API 지원• Hand-Eye 캘리브레이션 추가	<ul style="list-style-type: none">• dnn 모듈 개선<ul style="list-style-type: none">- cuda와 통합된 GSoC 프로젝트• 성능 개선<ul style="list-style-type: none">- SIMD 지원 확대- pryDown 멀티스레딩 지원• FSR 알고리즘

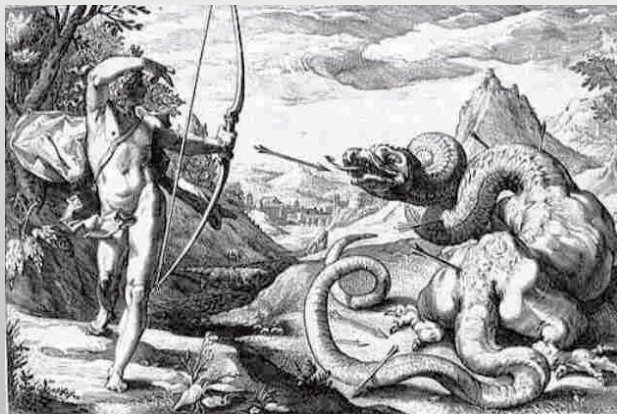
2.1.2 파이썬 개요

◆파이썬 개발

- ❖ 귀도 반 로섬(Guido Van Rossum) 발표(1991 년)
- ❖ 인터프리터 언어
 - 소스 코드를 1행씩 해석하고 실행해 바로 결과를 확인할 수 있는 언어
- ❖ 고급(high level) 프로그래밍 언어
- ❖ 플랫폼에 독립적, 객체지향적이고 동적 타입의 대화형 언어

◆명명이유

- ❖ 반 로섬이 좋아했던 영국의 코미디 프로인 "몬티 파이썬의 날아다니는 서커스"에서 따옴



2.2 파이썬(Python) 설치 및 사용

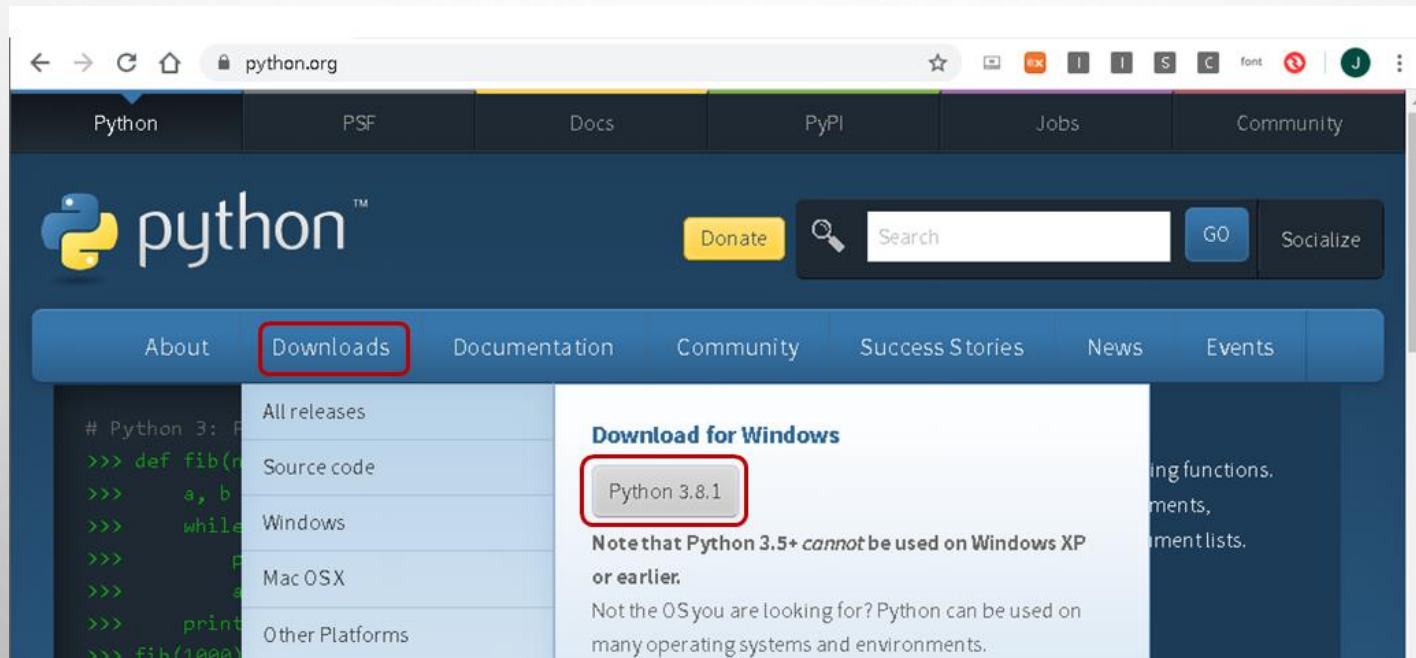
◆ 2.2.1 다운로드 및 설치

◆ 2.2.2 IDLE로 파이썬 프로그램 작성하기

2.2.1 다운로드 및 설치

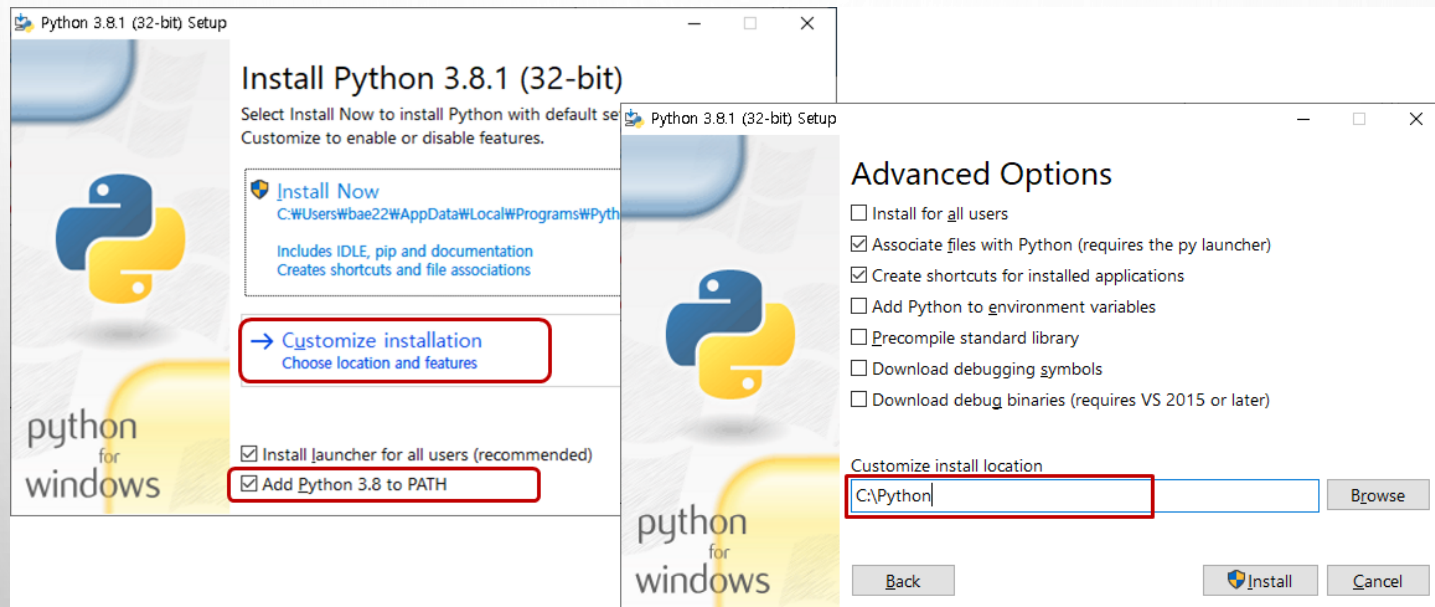
◆파이썬 다운로드

- ❖ 파이썬 홈페이지(<http://www.python.org>)
- ❖ [Downloads] 메뉴 클릭 → 'Python 3.8.1' 아이콘 클릭



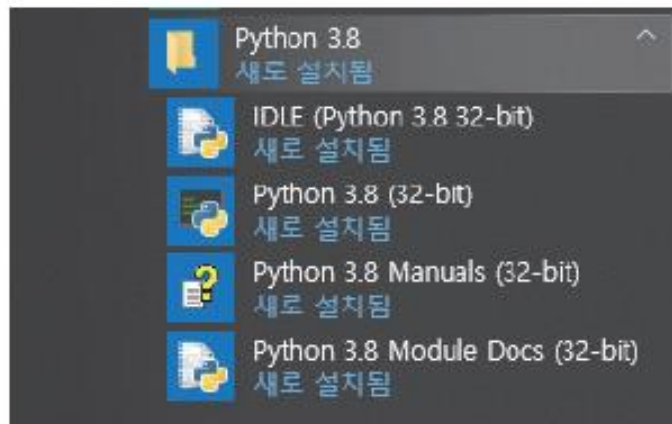
2.2.1 다운로드 및 설치

◆설치 과정

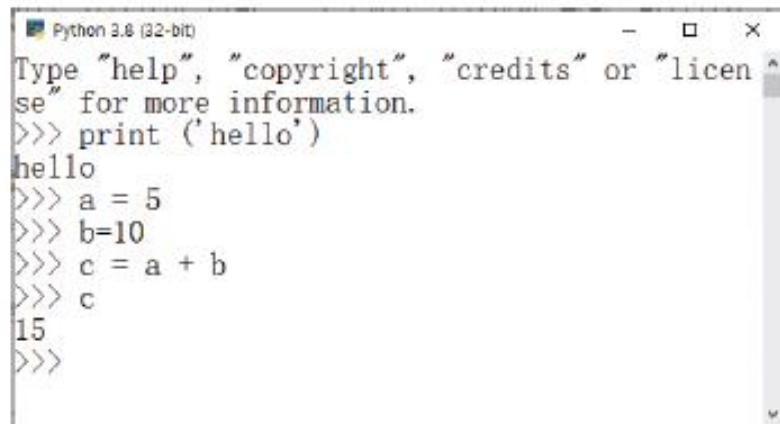


2.2.1 다운로드 및 설치

◆설치 완료 후



〈그림 2.2.3〉 윈도우 시작 메뉴

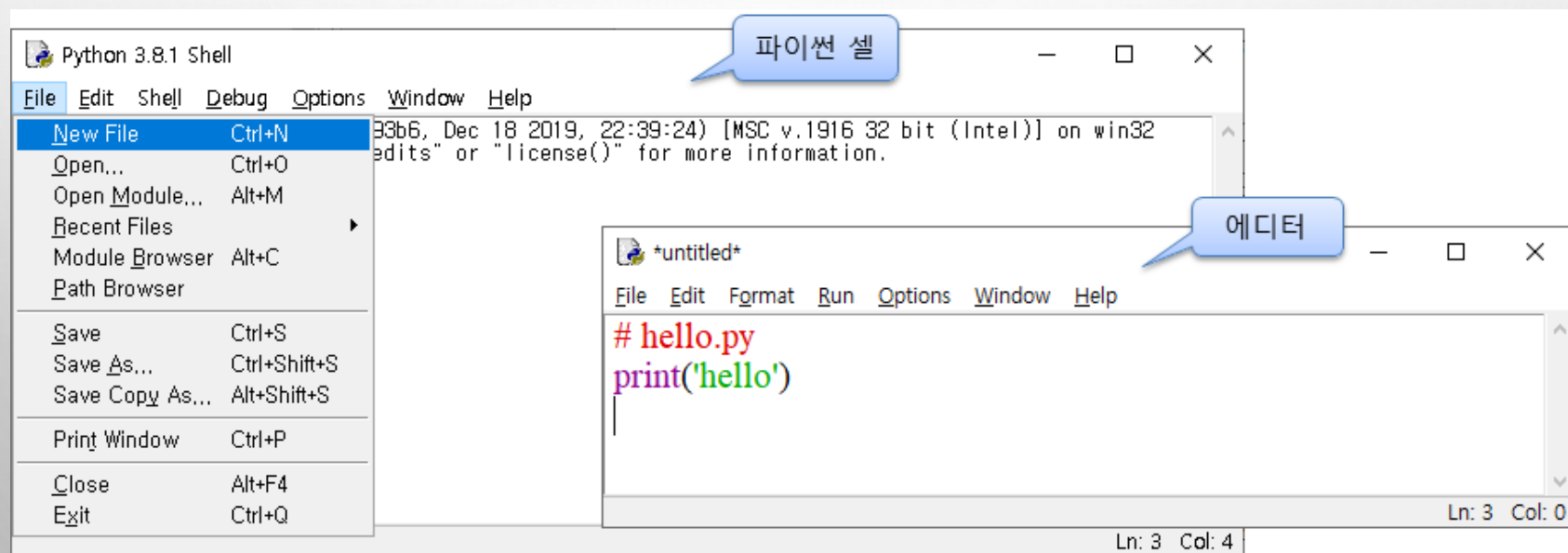


〈그림 2.2.4〉 Python 3.8(32-bit) 실행 화면

2.2.2 IDLE로 파이썬 프로그램 작성하기

◆파이썬에서 제공하는 통합 개발 환경

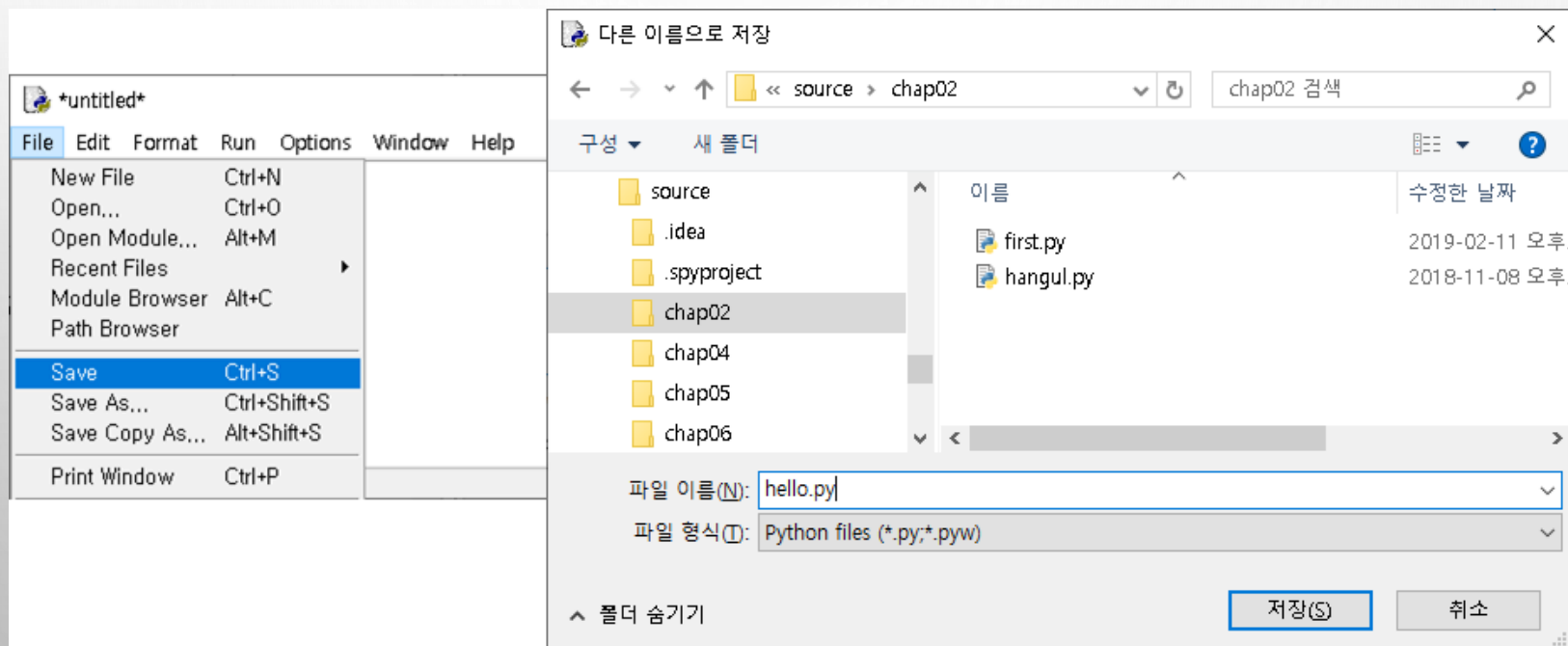
- ❖ IDLE(Integrated Development and Learning Environment)
- ❖ 간단한 소스편집과 실행을 할 수 있는 셸(Shell) 프로그램
- ❖ 윈도우 시작 메뉴 → [Python 3.8] 폴더 클릭 → [IDLE (Python 3.8 32-bit)] 클릭



2.2.2 IDLE로 파이썬 프로그램 작성하기

◆파일 저장

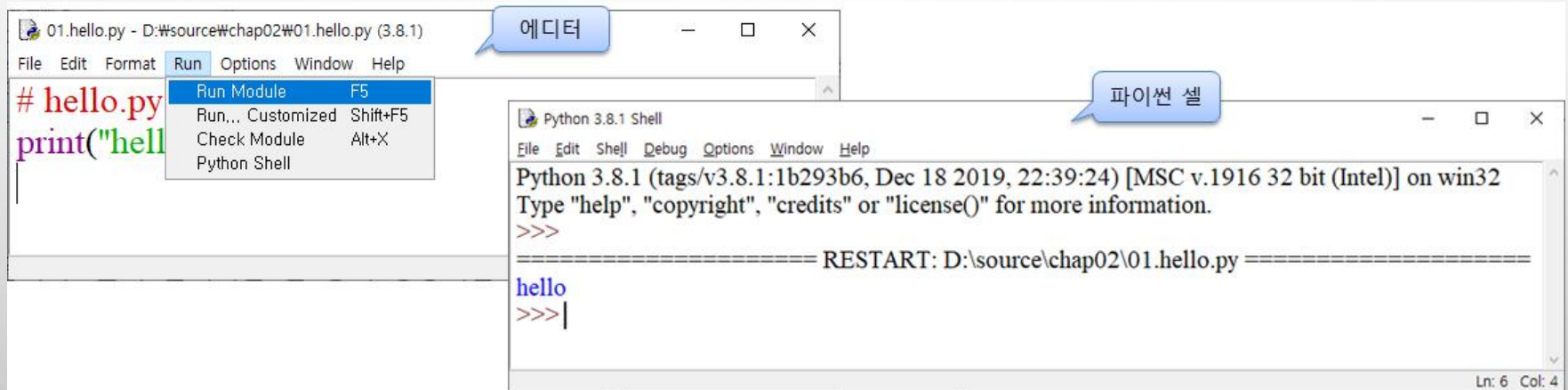
- ❖ [File] → [Save] 클릭 → [다른 이름으로 저장하기] 창 → 적당한 폴더 선택
- ❖ → "hello.py"로 파일 이름 지정 → [저장] 클릭



2.2.2 IDLE로 파이썬 프로그램 작성하기

◆ 실행

❖ 메뉴에서 [Run] → [Run Module] 클릭 → 해당 소스가 파이썬 셸에서 실행



2.3 파이참(PyCharm) 설치

◆파이참

- ❖ 젯브레인즈(JetBrains)사의 IntelliJ IDEA에 기반을 두고 개발된 프로그램으로서 파이썬 언어를 위한 거의 모든 기능을 갖춘 통합 개발 환경

◆널리 사용되는 이유

- 프로젝트별로 다른 Python 버전과 환경을 설정할 수 있다.
- 소스 코드의 실행 결과를 바로 확인할 수 있다.
- 직관적인 사용자 인터페이스를 제공하며, 운영체제와 무관하게 사용할 수 있다.

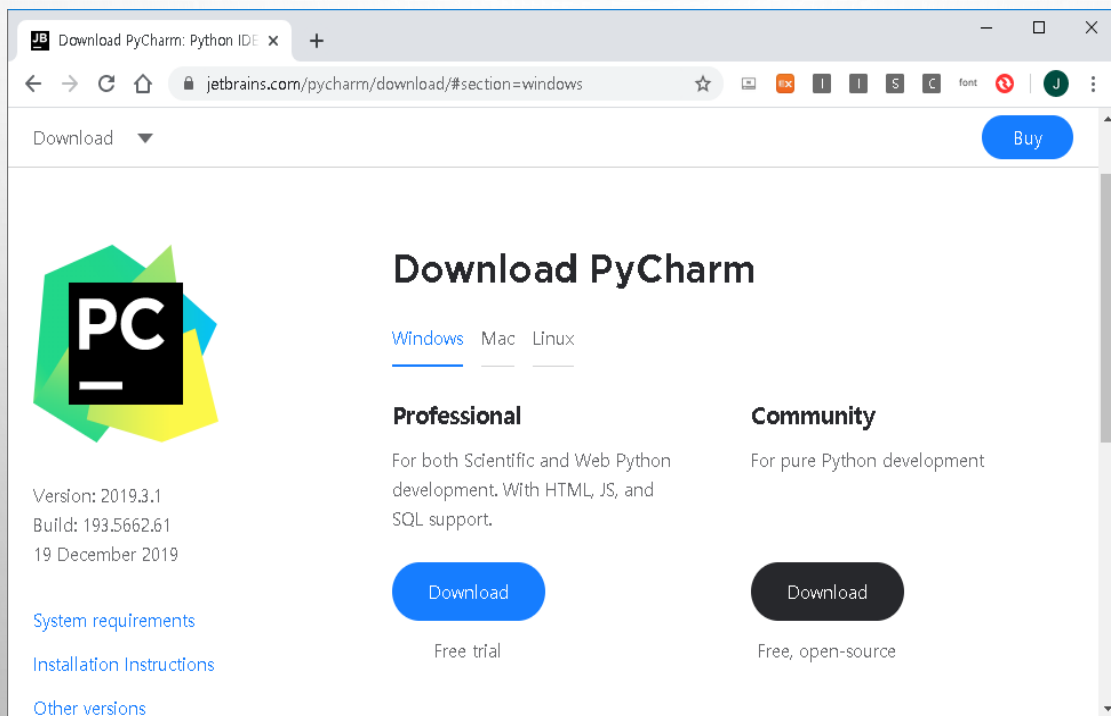
◆버전

- ❖ 커뮤니티 버전(Community Edition) : 무료 제공
- ❖ 프로페셔널 버전(Professional Edition) : 상용

2.3 파이참(PyCharm) 설치

◆파이참 설치

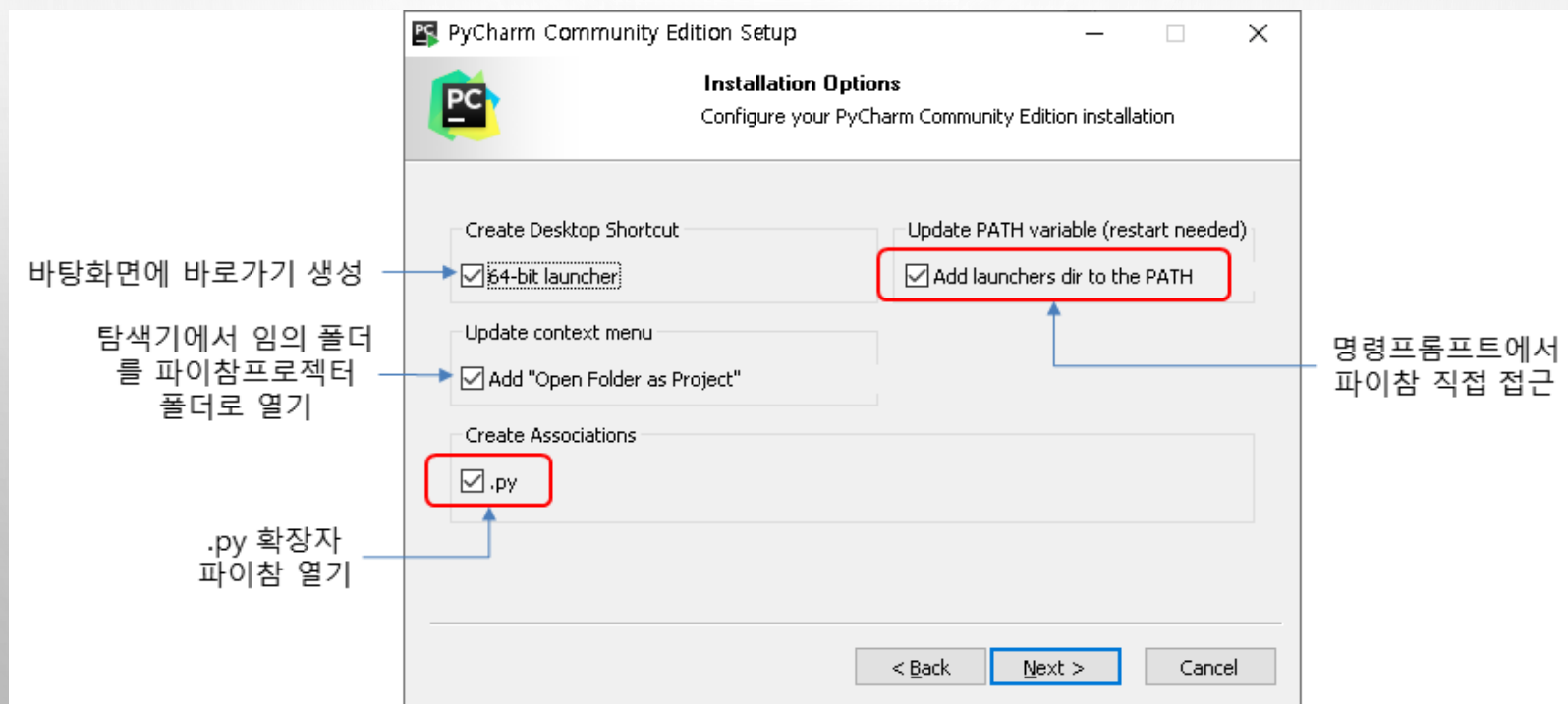
- ❖ 파이참 다운로드 페이지: <https://www.jetbrains.com/pycharm/download/#section=windows>
- ❖ 커뮤니티 버전 다운로드



2.3 파이참(PyCharm) 설치

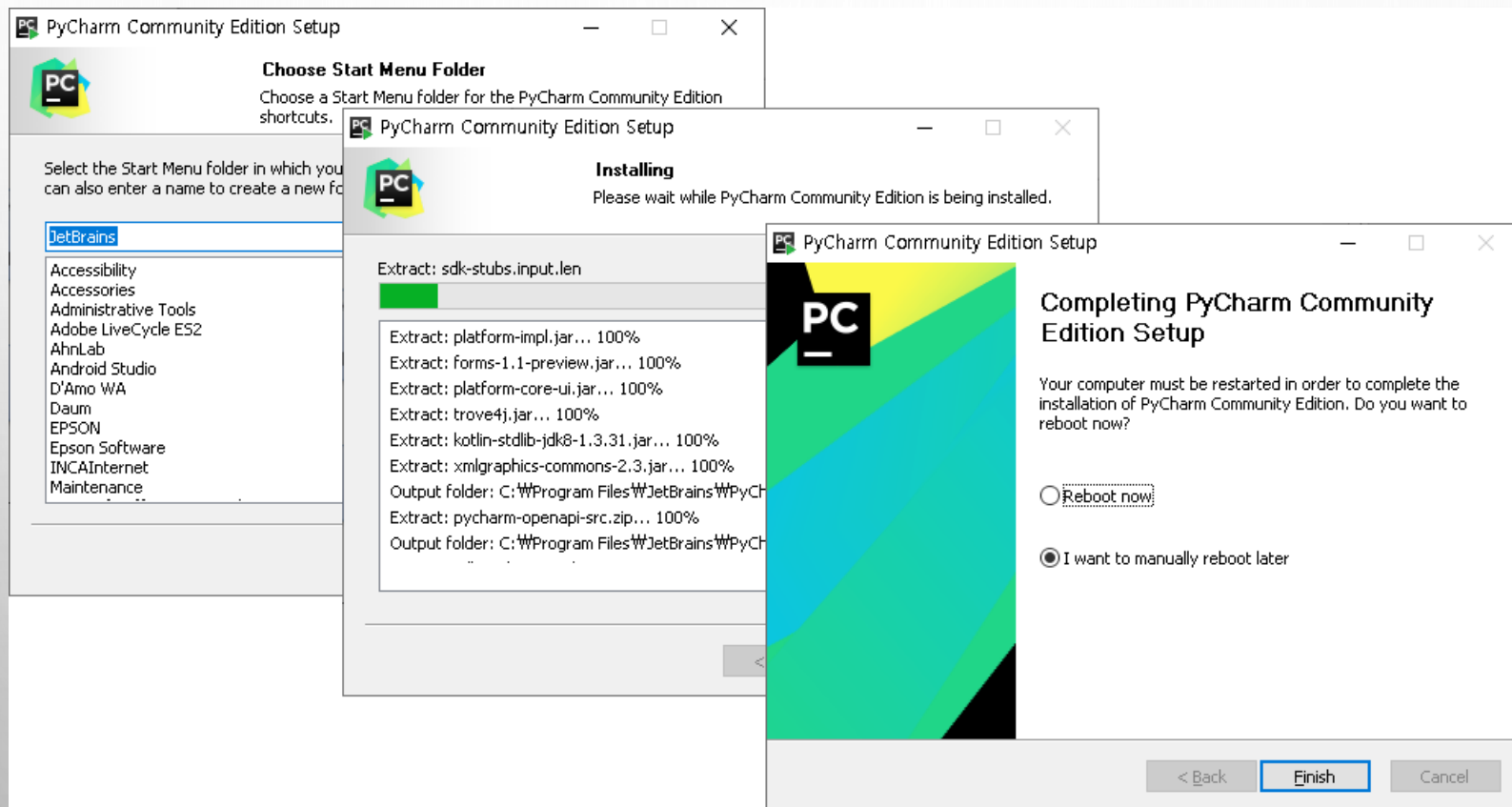
◆ 설치 진행

- ❖ 다음 사항 체크
- ❖ 전부 체크해도 무방



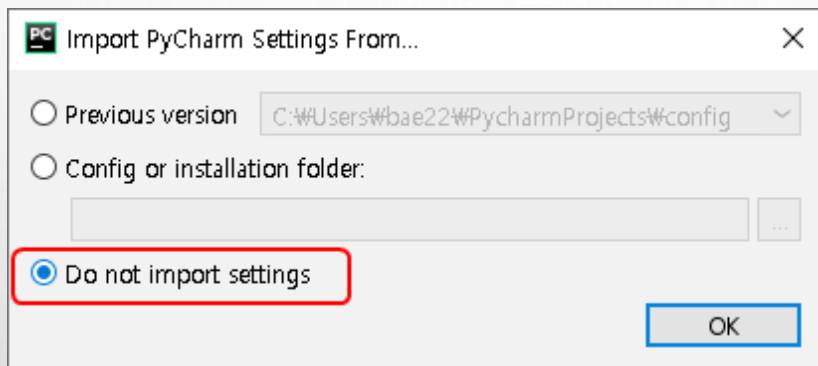
2.3 파이참(PyCharm) 설치

◆설치 진행



2.3 파이참(PyCharm) 설치

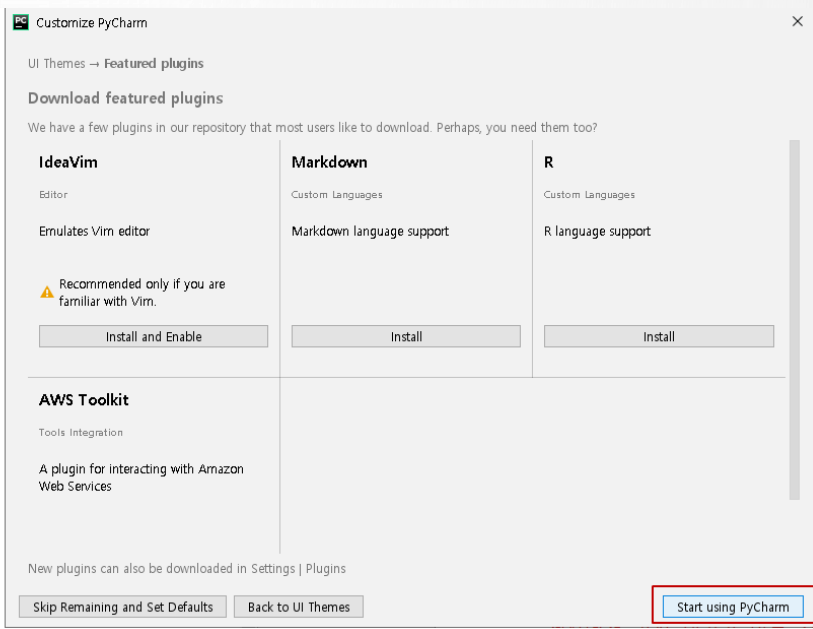
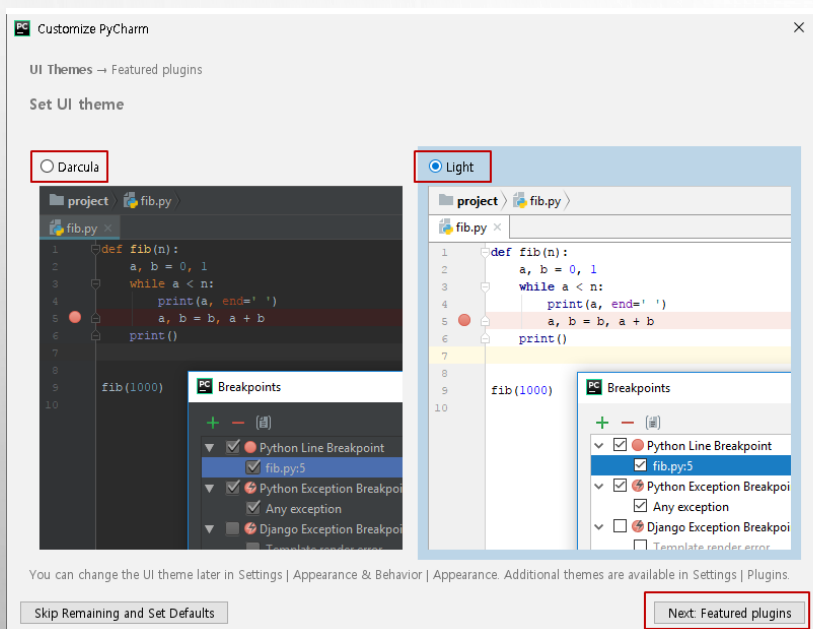
- ◆ 윈도우 시작 메뉴에서 [JetBrains] → [PyCharm Community Edition] 클릭
 - ❖ 기존 설치 버전 환경 가져오기 선택 가능
 - ❖ 기존 버전 없으면 “Do not import settings” 선택 → [OK] 클릭



2.3 파이참(PyCharm) 설치

◆ 사용자 테마 및 플러그인 선택

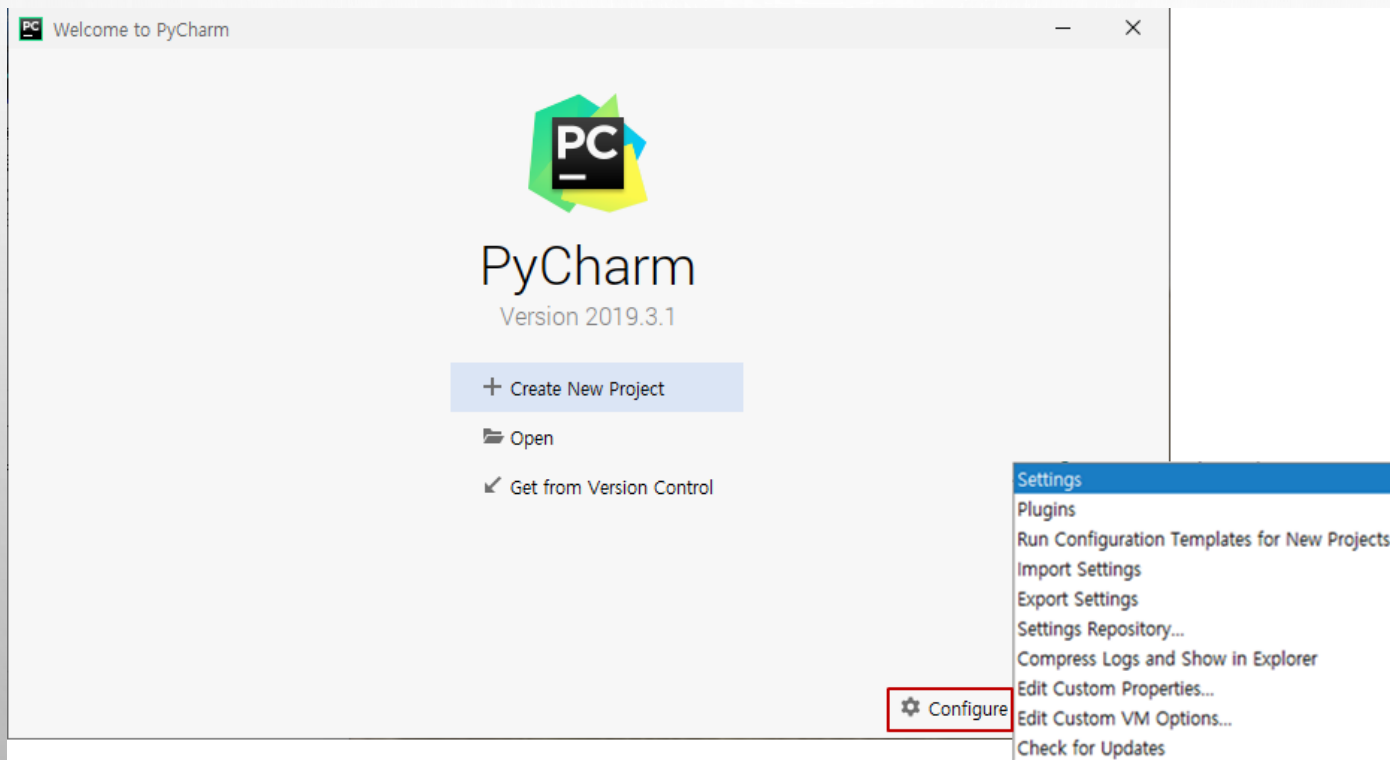
- ❖ 사용자 테마: 어두운 배경 or 밝은 배경 선택 가능
- ❖ 플러그인 인은 추후 선택함



2.4 파이참 환경 설정

◆Python 환경 설정: Python 엔진에 파이참 연결하는 과정

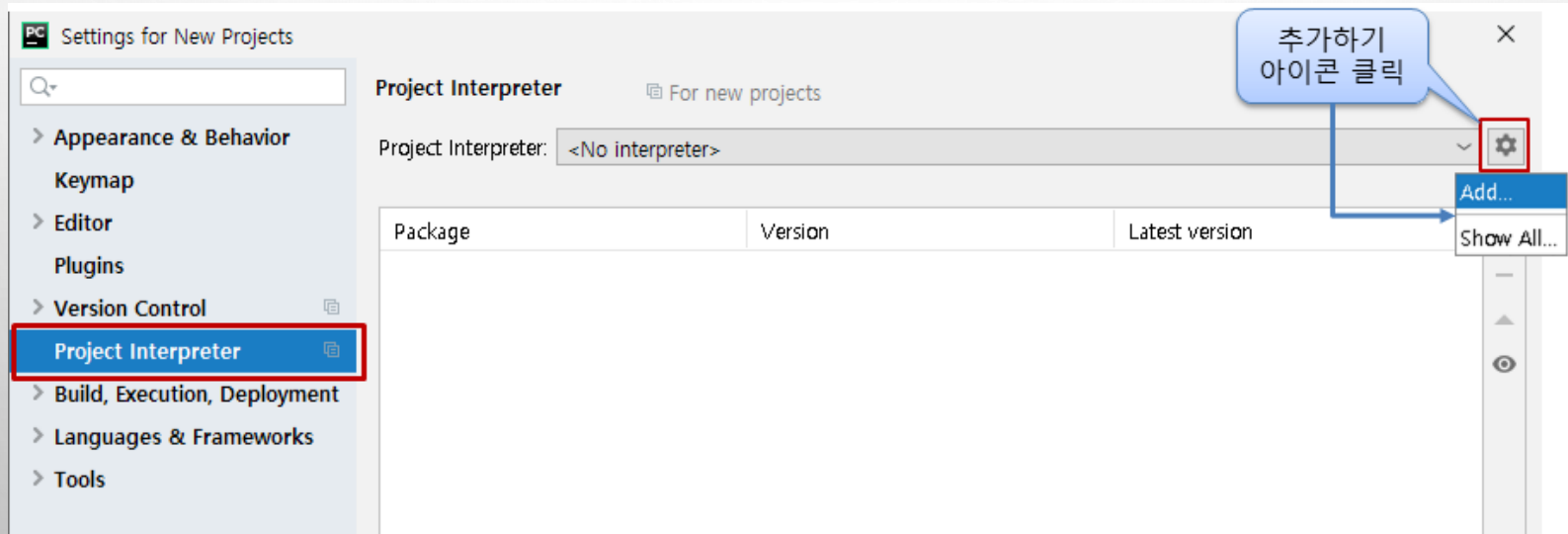
◆파이참 실행 → [Configure]→[Settings] 클릭



2.4 파이참 환경 설정

◆ Setting for New Projects 윈도우

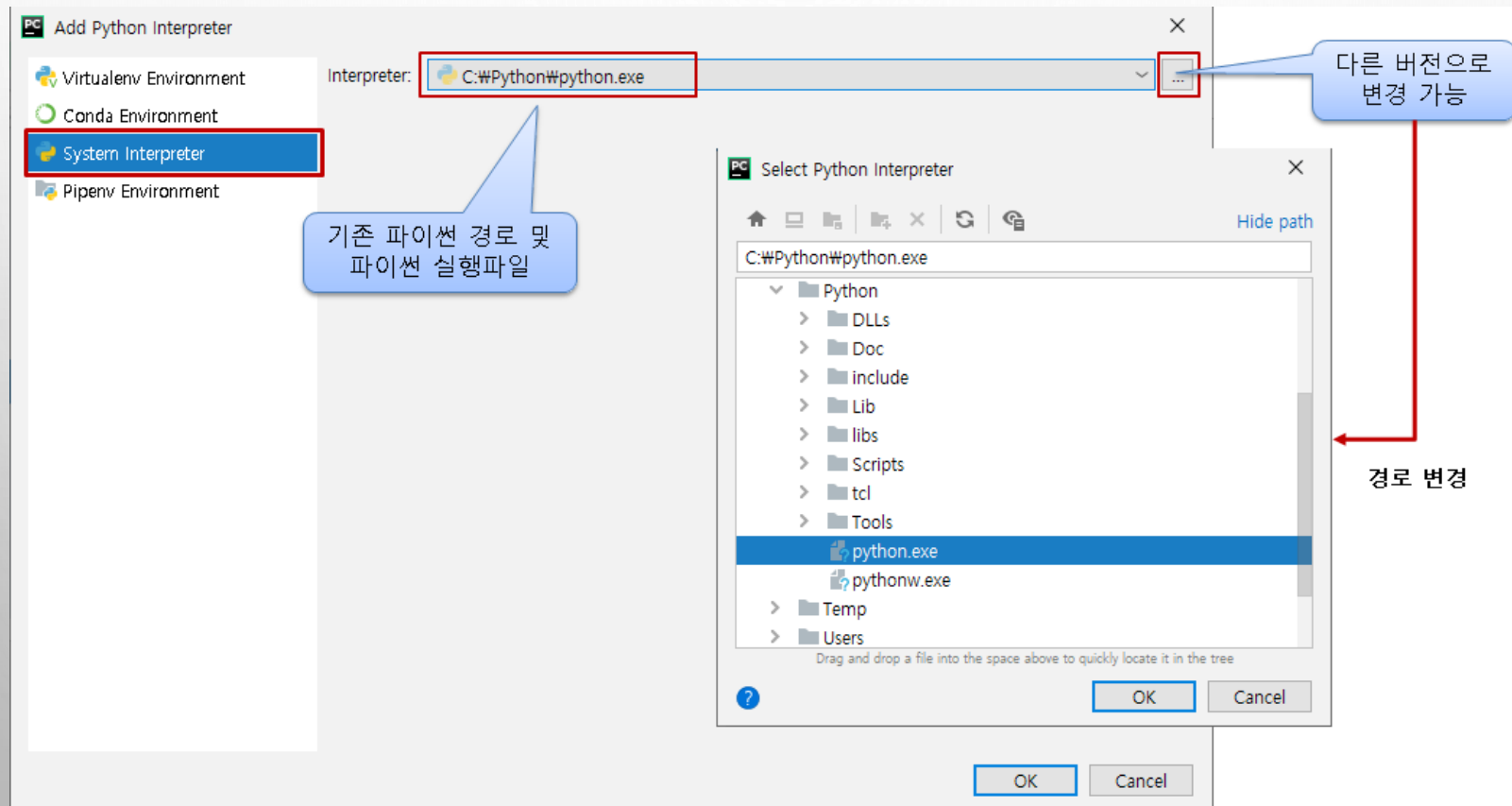
- ❖ 좌측 메뉴에서 [Project interpreter] 클릭
- ❖ 우측 상단에서 추가하기 아이콘 클릭 → [Add] 탭 팝업 → [Add] 클릭



2.4 파이참 환경 설정

◆ “Add Python Interpreter” 윈도우

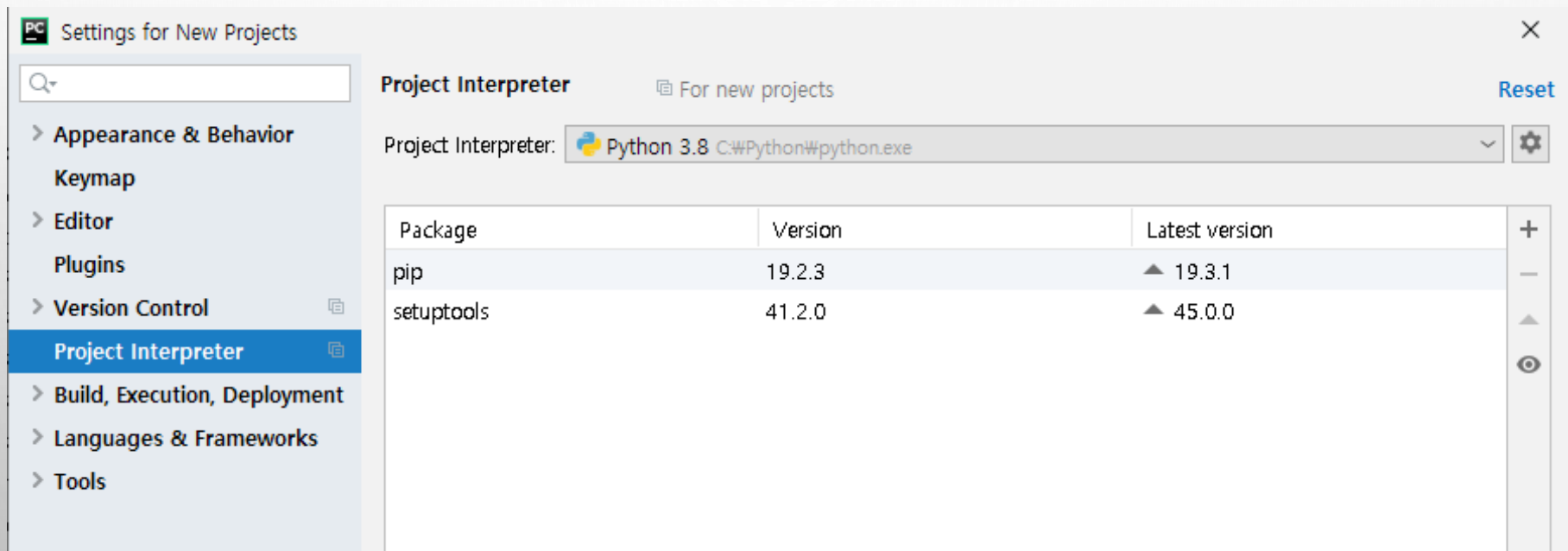
❖ 시스템 인터프리터 설정



2.4 파이참 환경 설정


◆ 설정 완료후 설치된 라이브러리 확인

❖ 현재 설치되어 있는 라이브러리는 pip 모듈과 setuptools 모듈 2개



PC Settings for New Projects

Project Interpreter For new projects [Reset](#)

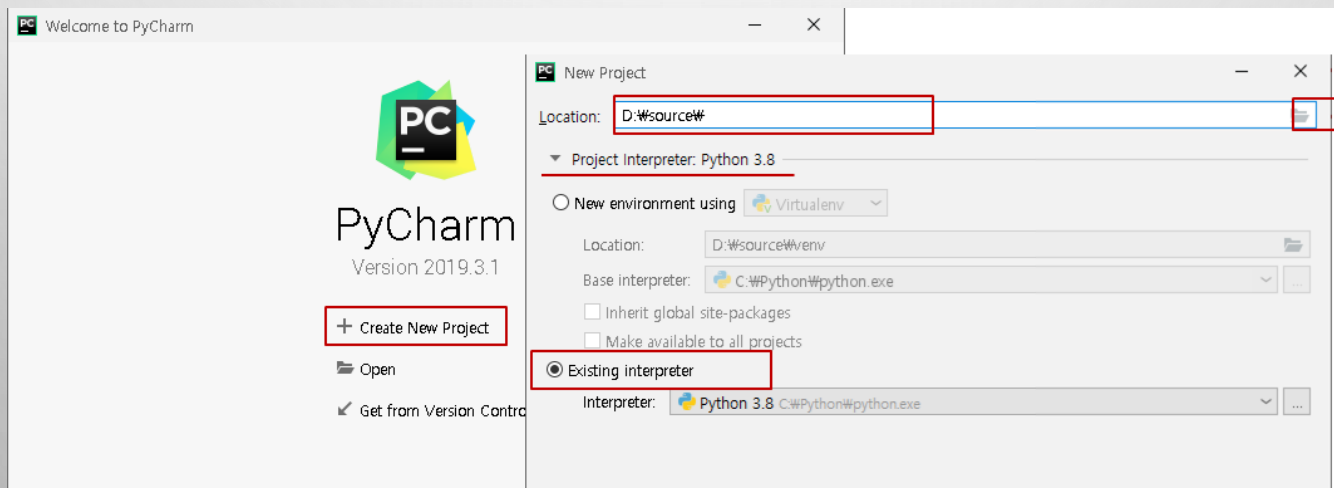
Project Interpreter:  Python 3.8 C:\Python\python.exe

Package	Version	Latest version	
pip	19.2.3	▲ 19.3.1	+
setuptools	41.2.0	▲ 45.0.0	-

2.4 파이참 환경 설정

◆ 새 프로젝트 생성

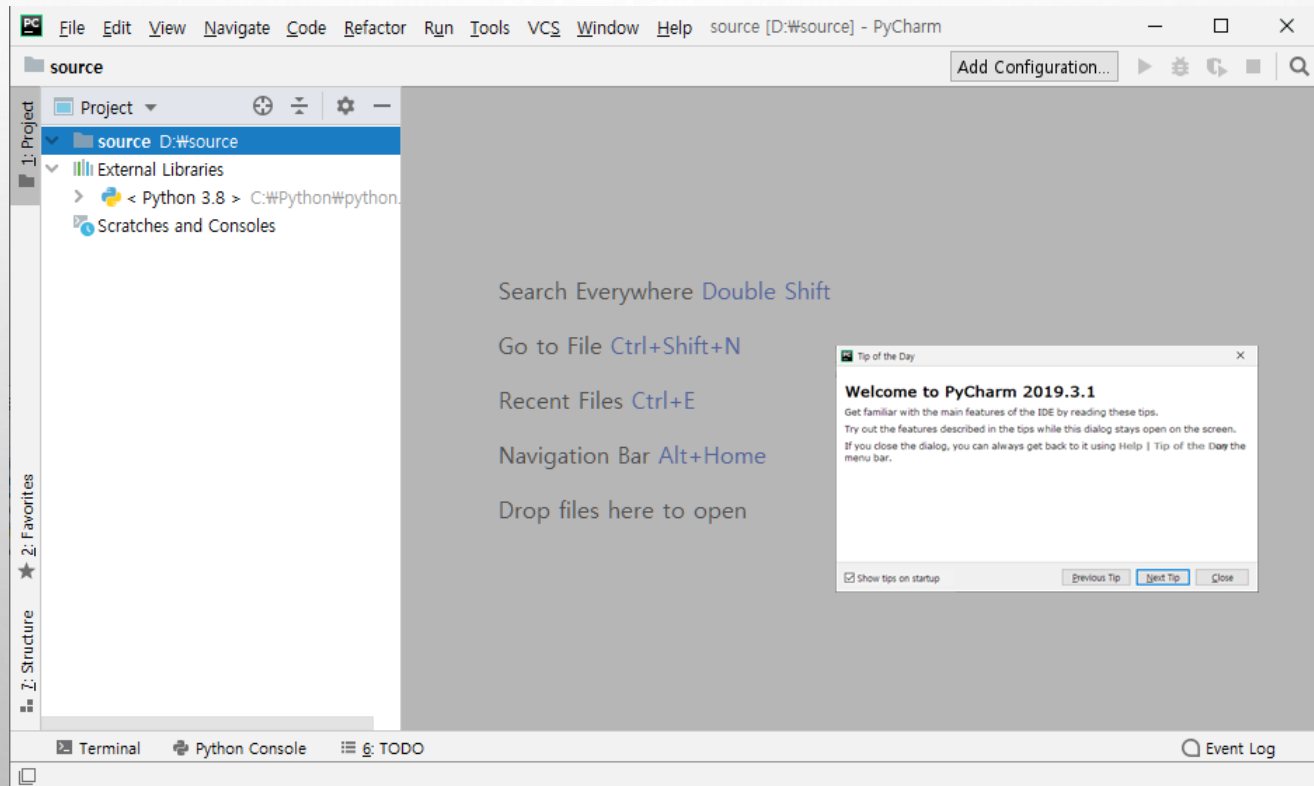
- ❖ [Create New Project] 클릭 → “New Project” 윈도우
- ❖ [Location] 항목 : 프로젝트 이름과 폴더 경로 입력
- ❖ [Project Interpreter Python 3.8] 클릭: 드롭다운 옵션들 보이게 함
- ❖ [Existing interpreter] 항목 체크 → [Create] 버튼 클릭 :: 기본 방법
- ❖ 가상 환경으로 프로젝트 생성하는 경우: [New environment using] 체크
 - 가상 환경으로 생성하면 프로젝트 내부에 파이썬 프로그램을 복사해서 독립적으로 사용하기 때문에 파이썬 버전에 맞는 각각의 라이브러리 버전을 설치해야 하는 경우 버전별 독립성 유지



2.4 파이참 환경 설정

◆ 프로젝트 생성 완료

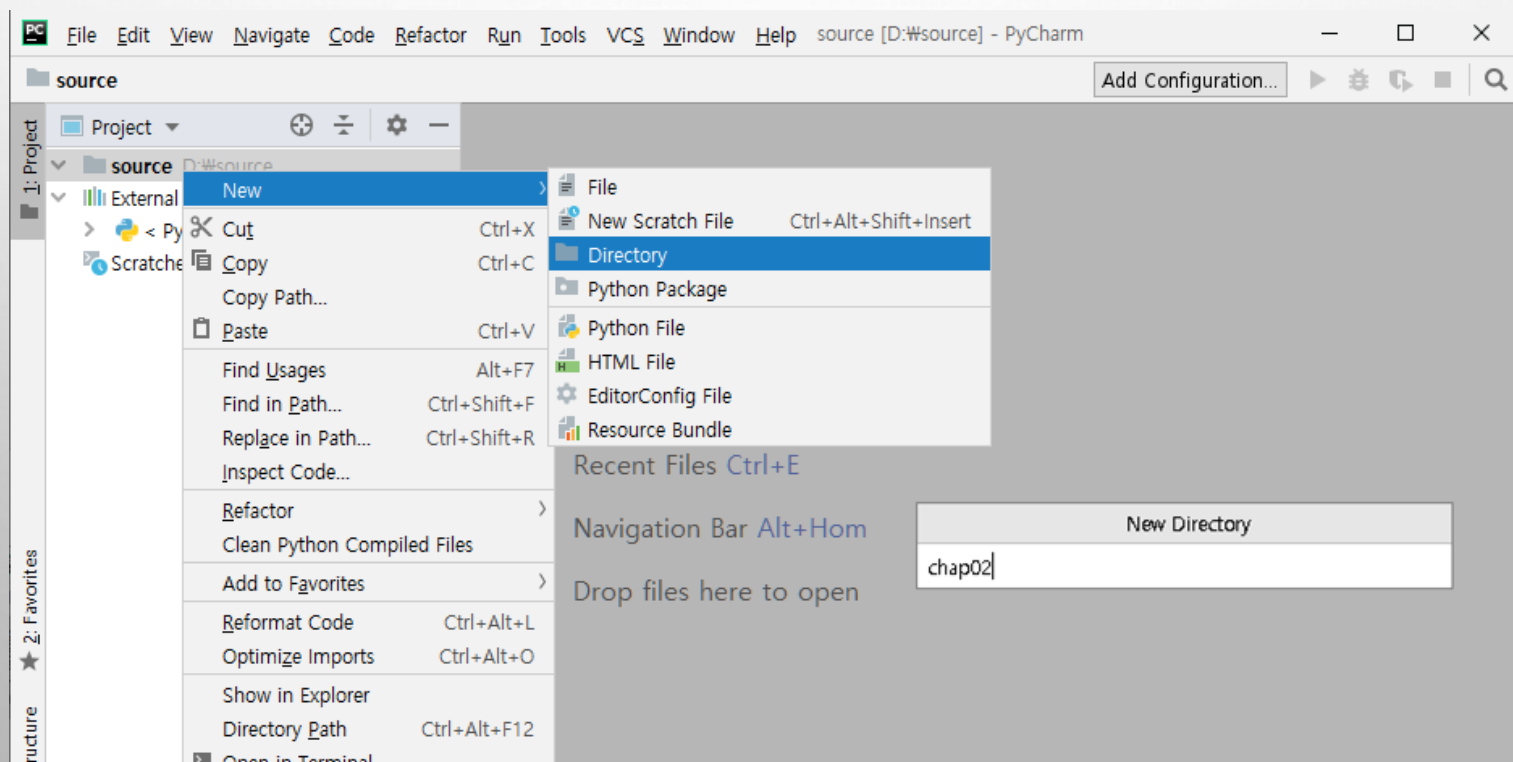
❖ “Welcome to PyCharm”이라는 메시지



2.4 파이참 환경 설정

◆ 프로젝트내 폴더와 파이썬 소스 파일 생성

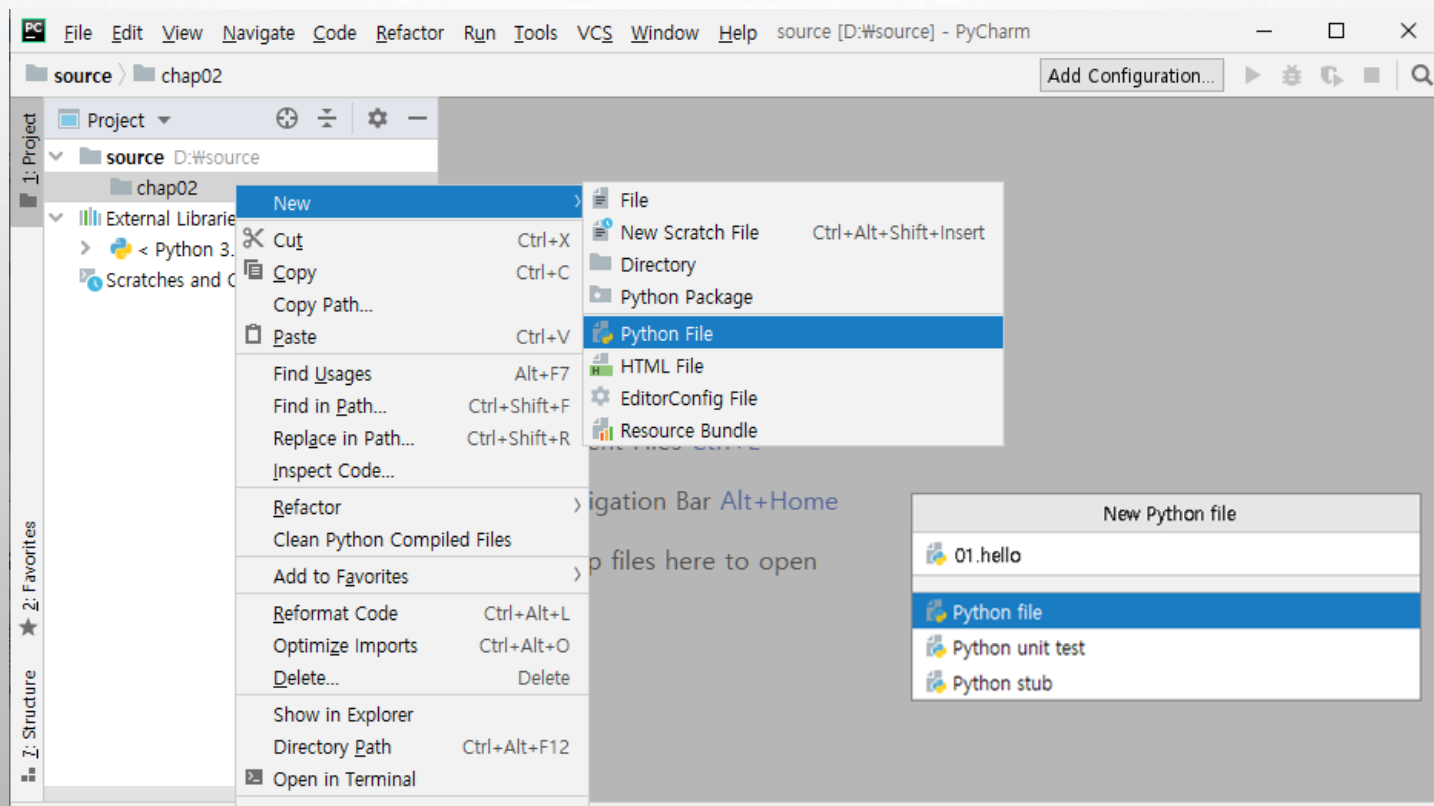
- ❖ source 폴더라는 프로젝트 만들어져 있음
- ❖ 마우스 오른쪽 버튼 클릭 → 팝업 메뉴 → [New] → [Directory]를 클릭
- ❖ New Directory 팝업창에서 폴더명 입력



2.4 파이참 환경 설정

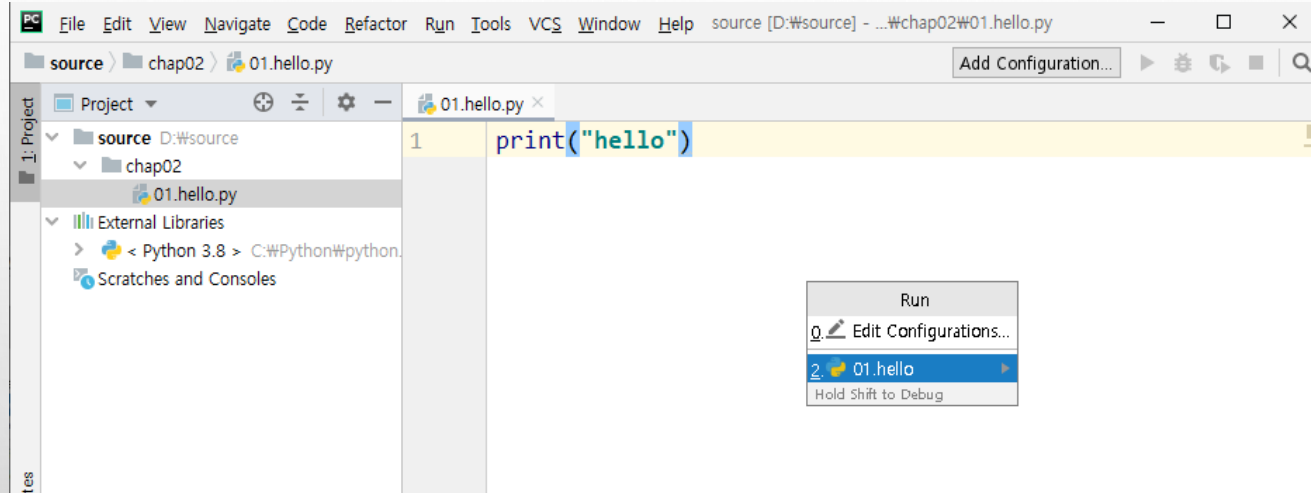
◆파이썬 소스 파일 생성

- ❖ 생성된 폴더 → 오른쪽 마우스 버튼 → [New] → [Python File] 클릭
- ❖ “New Python file” 윈도우 팝업 → 파일명 입력



2.4 파이참 환경 설정

◆ 간단한 소스 작성하기

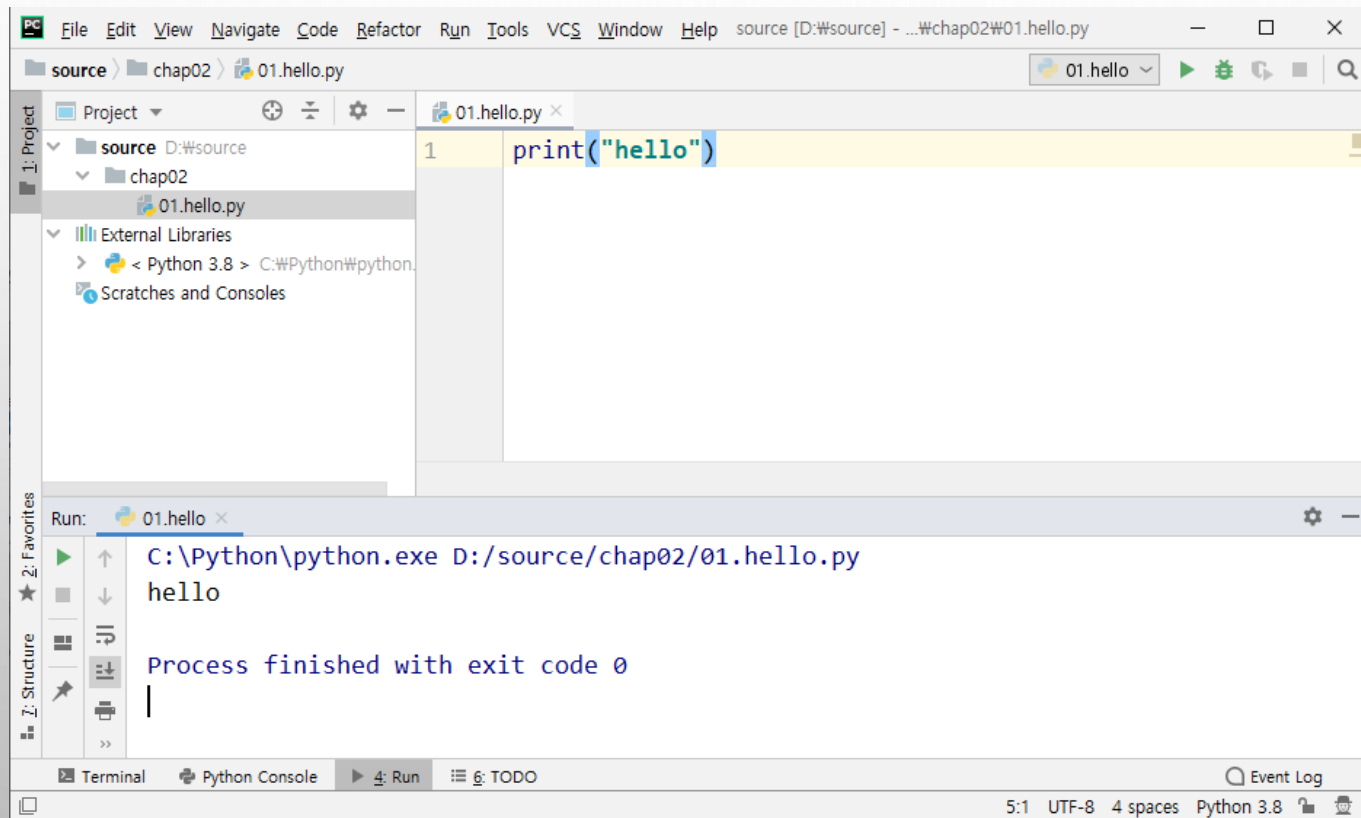


◆ 실행 하기

- ❖ 상단 메뉴에서 [Run] → [Run] 항목 클릭

2.4 파이참 환경 설정

◆ 실행 결과



The screenshot displays the PyCharm IDE interface. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The title bar shows the file path: source [D:\source] - ...#chap02#01.hello.py. The left sidebar contains a Project view showing the directory structure: source (D:\source) > chap02 > 01.hello.py. The main editor window shows the code for 01.hello.py, which consists of a single line: `print("hello")`. The bottom panel is divided into two sections. The top section, labeled 'Run:', shows the command `C:\Python\python.exe D:/source/chap02/01.hello.py` and the output `hello`. The bottom section shows the message `Process finished with exit code 0`. The bottom status bar indicates the file encoding is UTF-8, the indentation is 4 spaces, and the Python version is 3.8.

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help source [D:\source] - ...#chap02#01.hello.py
```

Project view: source (D:\source) > chap02 > 01.hello.py

Editor view: 01.hello.py

```
1 print("hello")
```

Run view: Run: 01.hello

```
C:\Python\python.exe D:/source/chap02/01.hello.py
hello

Process finished with exit code 0
```

Status bar: 5:1 UTF-8 4 spaces Python 3.8

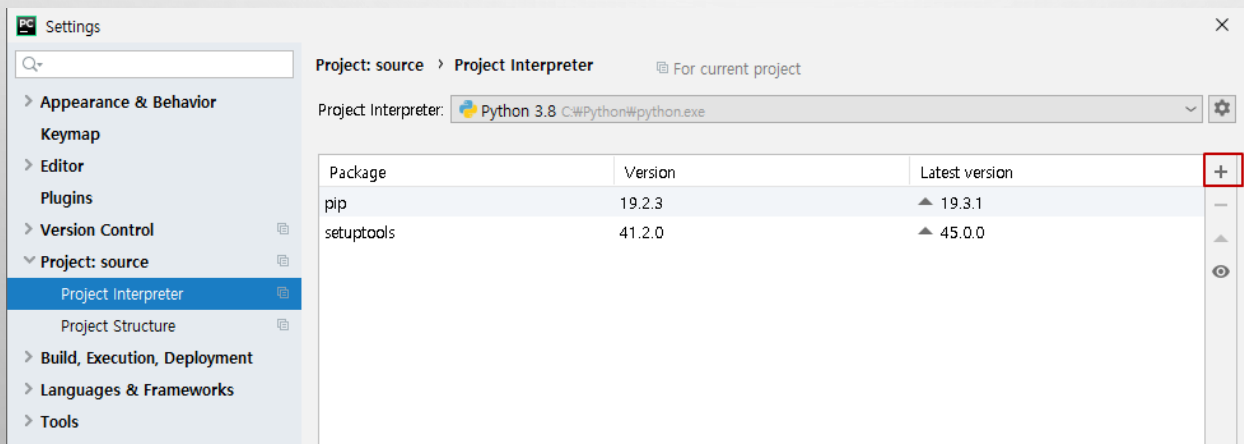
2.5 OpenCV-Python 및 라이브러리 설치

◆파이썬에 추가적인 라이브러리들 설치 방법

- ❖ 콘솔창에서 직접 pip 명령어를 통해서 라이브러리의 명칭 입력
- ❖ 파이참에서는 대화창을 통해서 라이브러리를 검색하여 클릭만으로 설치하는 방법 제공

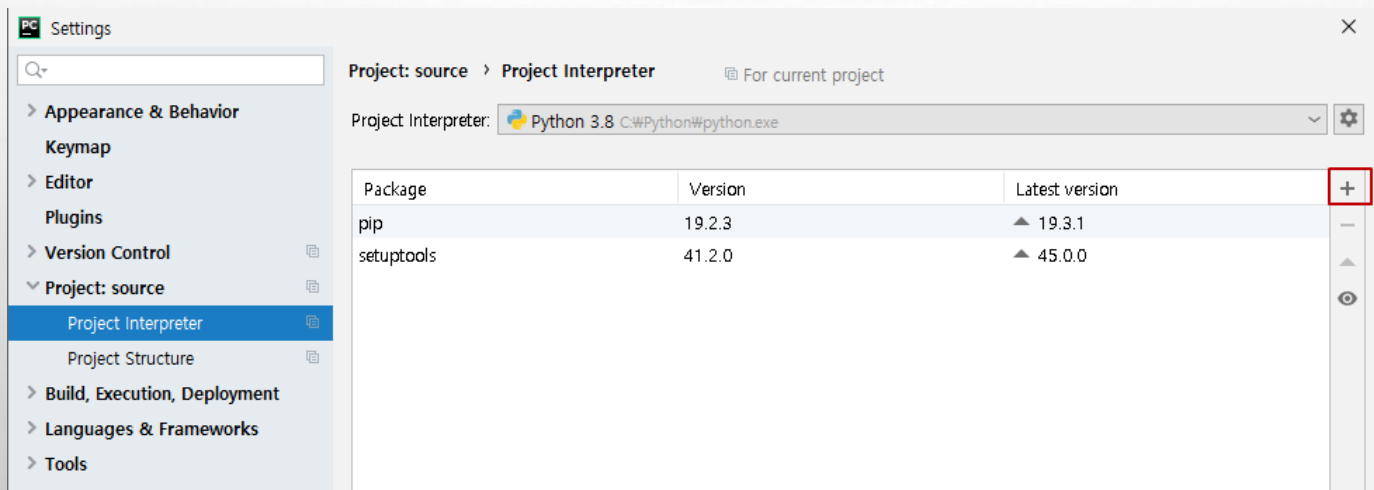
◆라이브러리 설치 하기

- ❖ 파이참 메뉴에서 [File] → [Settings] 클릭
- ❖ [Settings] 윈도우



2.5 OpenCV-Python 및 라이브러리 설치

- ❖ 왼쪽 메뉴에서 [Project source] → [Project interpreter] 메뉴 클릭
- ❖ 중앙 상단에 [Project interpreter] 항목 : 파이참에 결된 파이썬 설치 버전
- ❖ 아래 [Package] 항목에 현재 설치된 라이브러리명, 설치된 버전, 최신 버전 표시됨

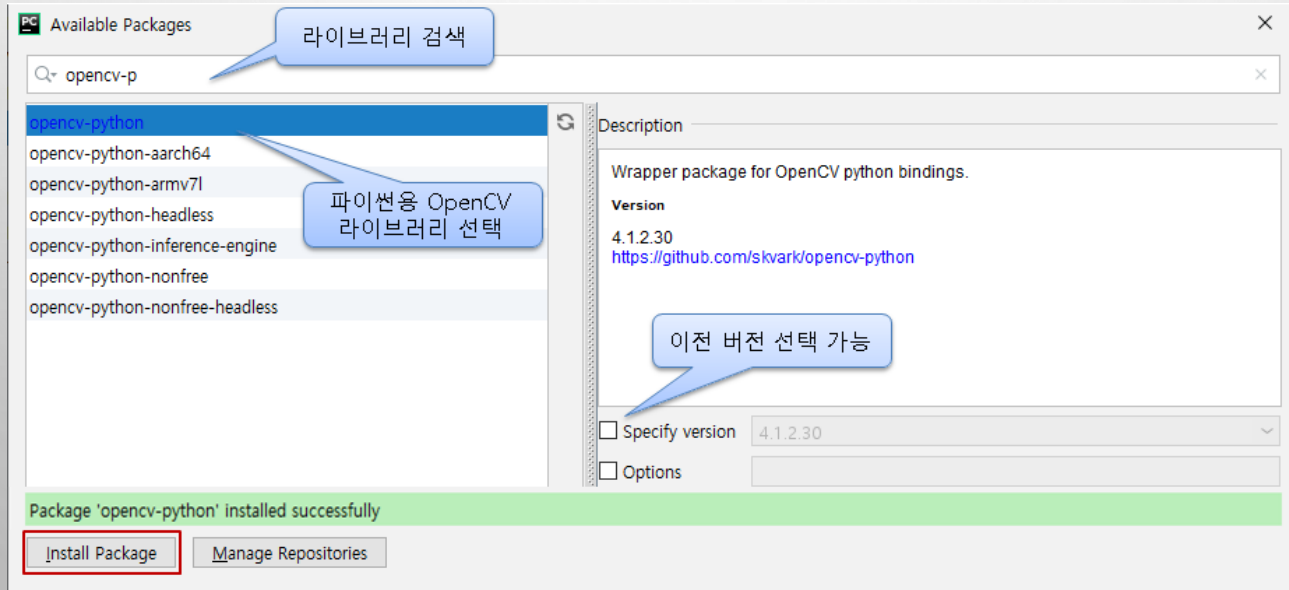


- ❖ 설정 대화창의 오른쪽에 라이브러리 추가하기 버튼[+] 클릭

2.5 OpenCV-Python 및 라이브러리 설치

❖ Available Packages 윈도우

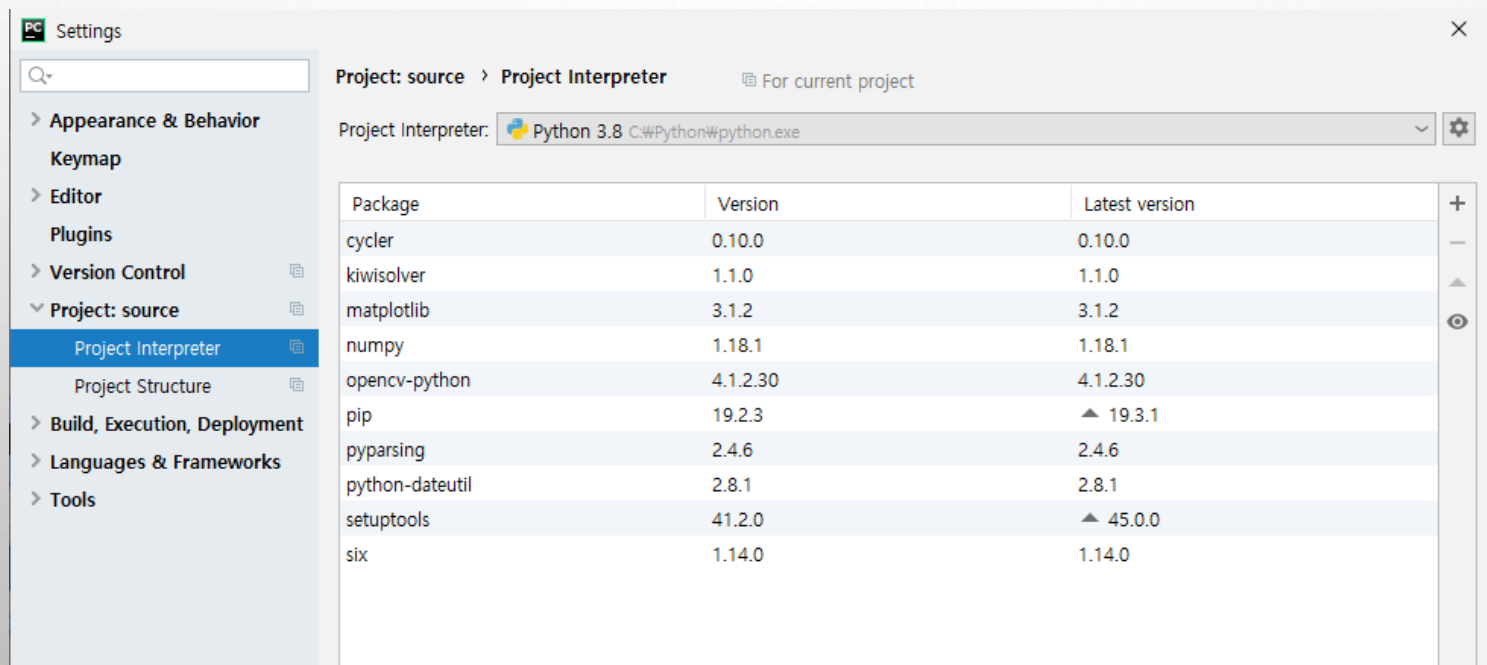
- 파이썬에 설치할 수 있는 다양한 라이브러리들 검색 가능
- 왼쪽 상단 라이브러리 검색에서 설치하려는 라이브러리 이름을 몇 글자만 입력
- 왼쪽 아래 창에서 검색어와 비슷한 이름의 라이브러리들이 검색되어 표시
- OpenCV-python 라이브러리를 선택 → [Install Package] 버튼 클릭
- 최신 버전이 아니라 이전 버전의 설치하려면
 - ✓ 오른쪽 하단의 [Specify version] 항목 체크하여 원하는 이전 버전 선택하여 설치 가능



2.5 OpenCV-Python 및 라이브러리 설치

◆라이브러리 설치 완료 후

❖ 그래프를 쉽게 그릴 수 있는 matplotlib 라이브러리도 설치함



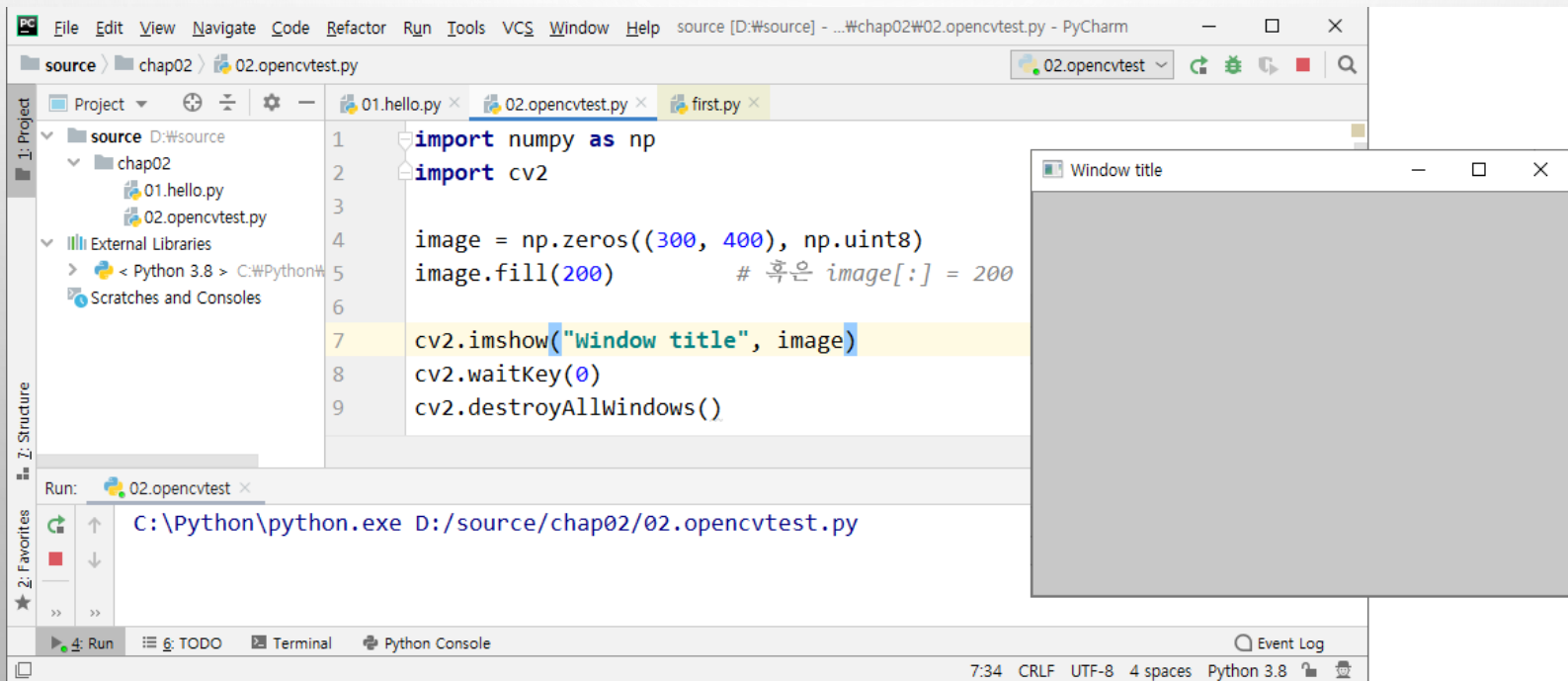
The screenshot shows the PyCharm Settings window with the 'Project Interpreter' tab selected. The 'Project Interpreter' dropdown is set to 'Python 3.8 C:\Python#python.exe'. Below it, a table lists the installed packages and their versions.

Package	Version	Latest version
cycler	0.10.0	0.10.0
kiwisolver	1.1.0	1.1.0
matplotlib	3.1.2	3.1.2
numpy	1.18.1	1.18.1
opencv-python	4.1.2.30	4.1.2.30
pip	19.2.3	▲ 19.3.1
pyparsing	2.4.6	2.4.6
python-dateutil	2.8.1	2.8.1
setuptools	41.2.0	▲ 45.0.0
six	1.14.0	1.14.0

◆ 간단한 OpenCV 프로그래밍

❖ 'chap02' 폴더에 '02.opencvtest.py' 소스 파일

- 300행, 400열 크기의 행렬 생성하여 행렬의 모든 원소의 값을 회색(200)으로 지정
- 이 행렬을 "window title" 이름의 윈도우에 영상으로 표시



단원 요약

- ◆파이썬은 대화형 인터프리터 언어이다. 인터프리터 방식은 1행씩 명령어를 입력하면 바로 해석해서 결과를 보여주는 방식이다.
- ◆OpenCV는 영상처리, 컴퓨터 비전, 기계 학습과 같은 응용에 사용할 수 있는 API를 제공하는 라이브러리이다.
- ◆파이참은 파이썬 개발을 위한 통합 개발 환경(IDE)이다. 통합 개발 환경은 에디터, 디버거, 컴파일러, 인터프리터 등을 모두 포함하여 프로그래밍 환경을 제공해주는 소프트웨어이다.
- ◆파이참에서 파이썬 개발을 위해서는 [파이썬 인터프리터]에 파이썬 실행 파일(python.exe)이 연결되어 있어야 한다.
- ◆파이참은 파이썬에서 사용하는 라이브러리들을 쉽게 설치할 수 있는데, [File] 메뉴 아래 [Settings] - [Project Interpreter]에서 라이브러리 추가하기[+] 대화상자를 열어서 검색을 통해서 설치할 수 있다.
- ◆파이참은 프로젝트를 기반으로 파이썬 소스 코드를 관리하며, 프로젝트에서는 폴더와 파일들을 포함할 수 있다.