

# Windows, MacOS에서 OpenCV 설치법

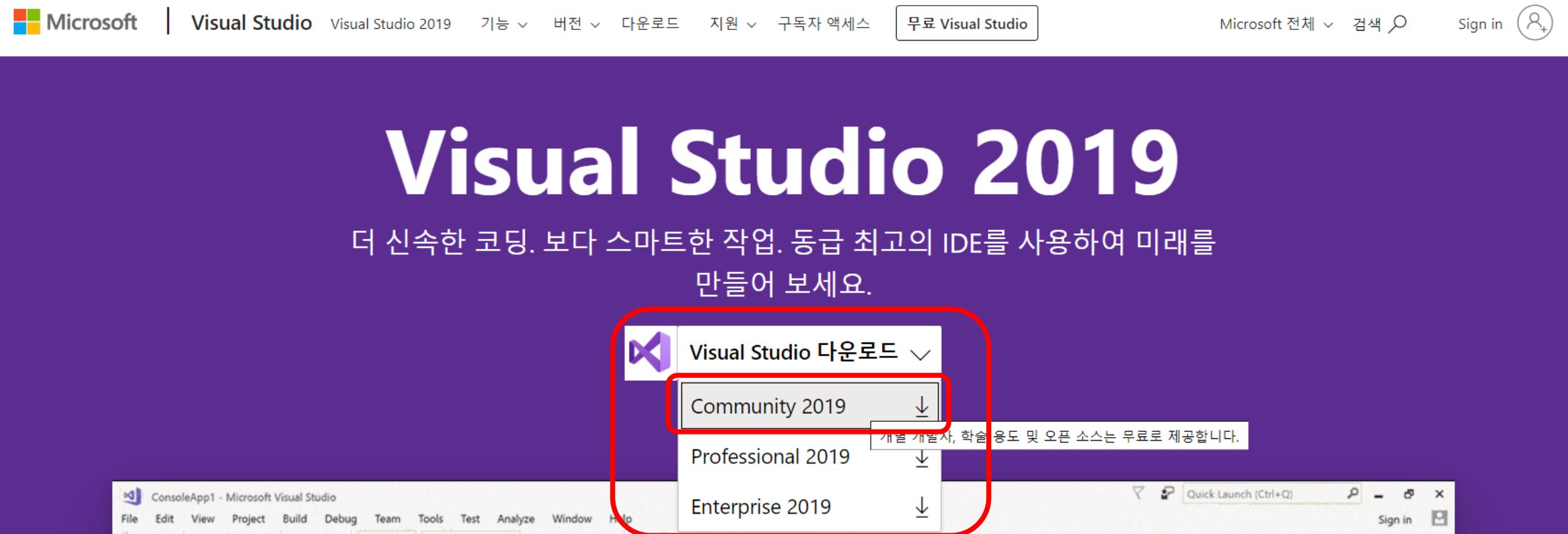
2020년 7월

한동대학교 전산전자공학부 이강

# 부록0: Visual Studio 2019 설치

# Visual Studio 설치: 다운로드(1)

- <https://visualstudio.microsoft.com/ko/vs/>



# Visual Studio 설치: 다운로드(2)

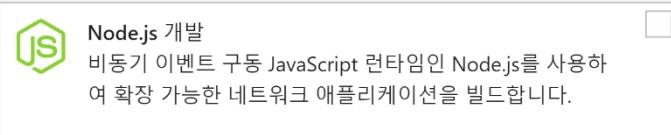
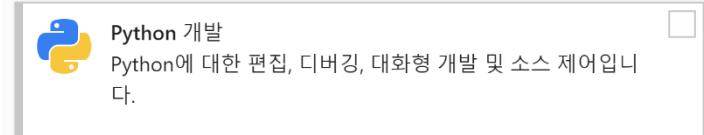
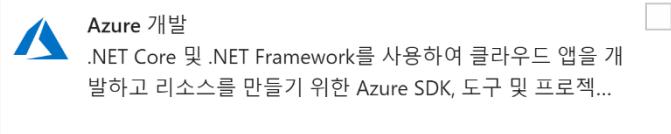
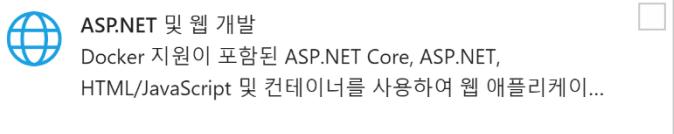


# Visual Studio 설치: C++ 개발 플랫폼 설치

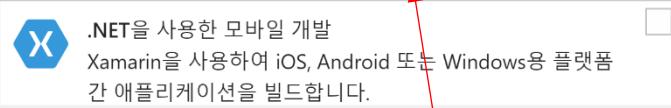
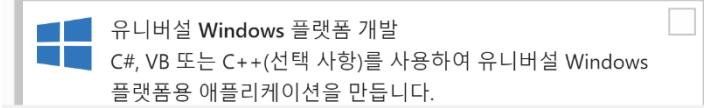
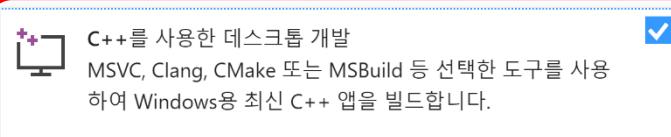
설치 중 — Visual Studio Community 2019 — 16.6.2

워크로드      개별 구성 요소      언어 팩      설치 위치

웹 및 클라우드 (4)



데스크톱 및 모바일 (5)



설치

## 설치 세부 정보

> Visual Studio 핵심 편집기

✓ C++를 사용한 데스크톱 개발  
포함됨

✓ C++ 핵심 데스크톱 기능

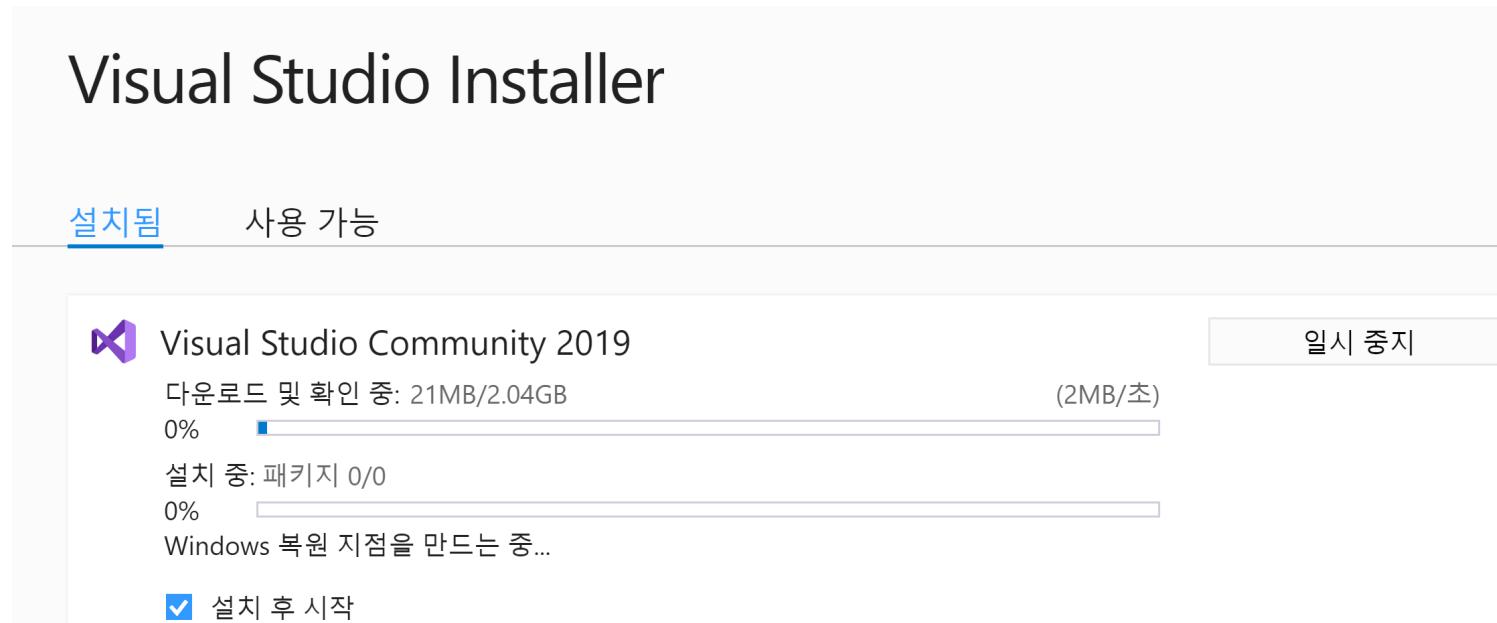
✓ IntelliCode

### 옵션

- MSVC v142 - VS 2019 C++ x64/x86 빌드 도구(v...)
- Windows 10 SDK(10.0.18362.0)
- Just-In-Time 디버거
- C++ 프로파일링 도구
- Windows용 C++ CMake 도구
- 최신 v142 빌드 도구용 C++ ATL(x86 및 x64)
- Test Adapter for Boost.Test
- Test Adapter for Google Test
- Live Share
- C++ AddressSanitizer(실험적)
- 최신 v142 빌드 노드용 C++ MFC(x86 및 x64)
- v142 빌드 도구용 C++/CLI 지원(14.26)

설치옵션은 기본값을 사용

# Visual Studio 설치: C++ 개발 플랫폼 설치



# 부록1-1: Windows에서 OpenCV 사용하기

(환경 변수 설정 없이 opencv 환경 설정 version)

# 웹에서 OpenCV 다운로드하기

- [www.opencv.org](http://www.opencv.org) 접속 > Releases 선택



## OpenCV @CES 2020

See you at our private party on Jan 7

[Learn More](#)

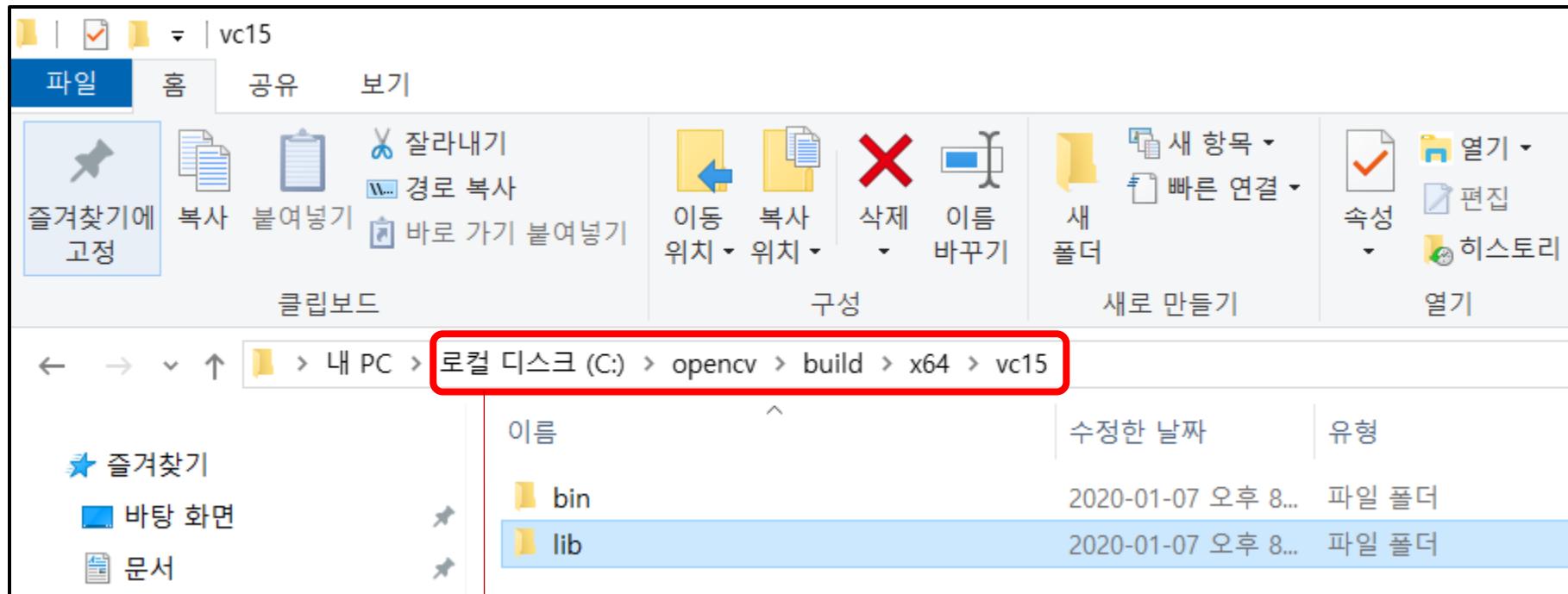
# 웹에서 OpenCV 다운로드하기: Windows 버전선택



A detailed view of the OpenCV 4.3.0 release page. At the top, it shows the release title "OpenCV - 4.3.0" and the date "2020-04-06". Below the title are several download links: "Docs", "Sources", "GitHub", "Windows" (highlighted with a red box), "iOS pack", "Android", and "Release Notes". Each link has a small icon next to it.

# OpenCV 사용 환경 설정

## : OpenCV를 다운로드한 위치 확인 (C:\opencv로 가정)



(opencv 설치 위치는 **C:\opencv**, 64-bit Windows10 or 7, Visual Studio 2019 or 2017를 쓴다고 가정)

\*주의: **build\vc15 폴더가 Visual Studio 2019 또는 2017 환경(64비트)에 해당하는 라이브러리임.**  
(vc14는 Visual Studio 2015 컴파일러 버전임)

# Visual Studio (VS) 프로젝트 설정하기

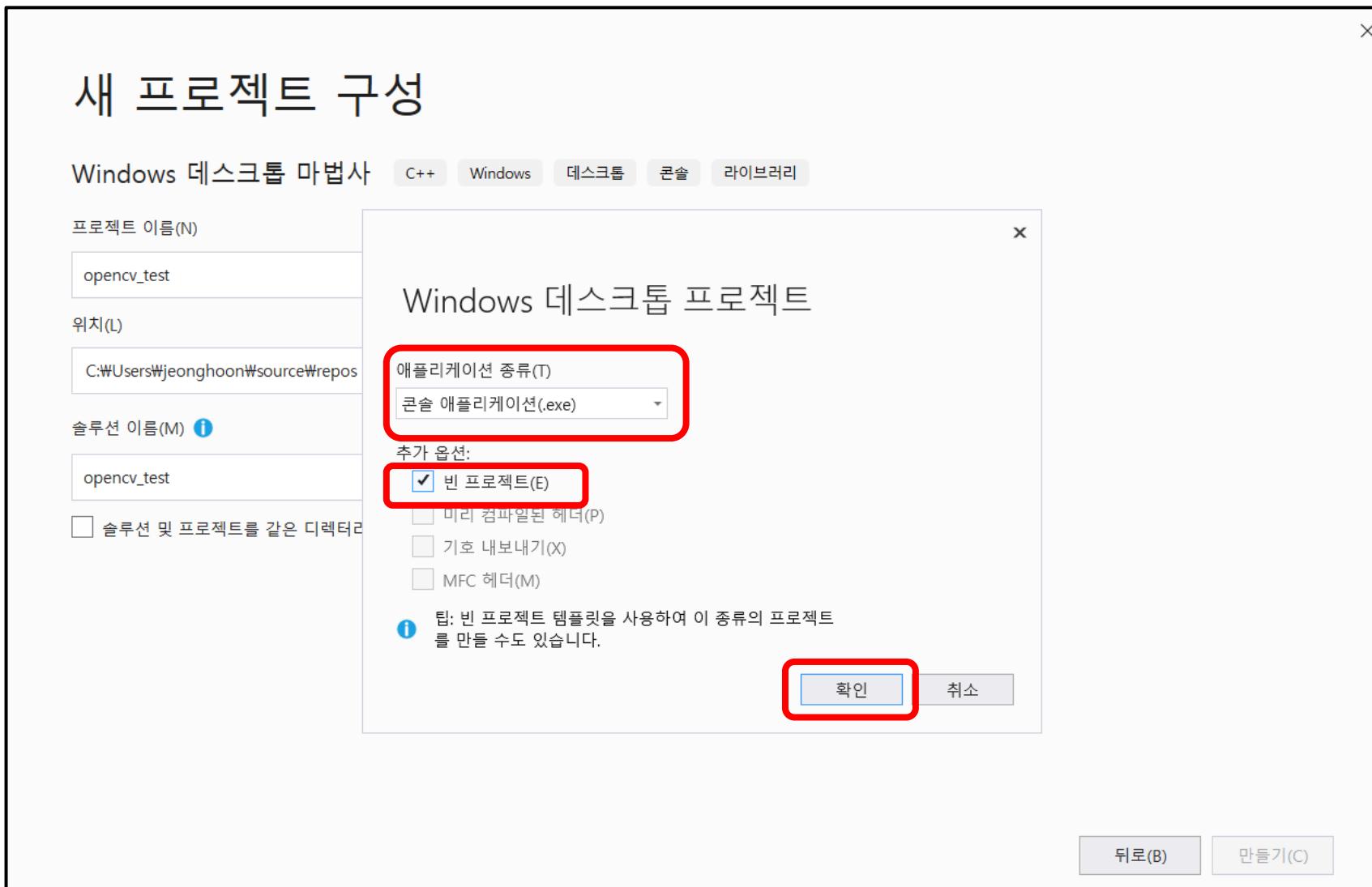
- 각 프로젝트별로, Project 새로 생성 후
  1. Visual Studio의 구성(Configuration)을 x64 플랫폼으로 설정하기
  2. Visual Studio의 속성에서 openCV 헤더파일 경로 설정하기
  3. Visual Studio의 openCV 라이브러리 경로 및 파일명 설정하기

→ 다음 슬라이드를 보면서 설정하세요!

# Visual Studio의 New project 생성(1)

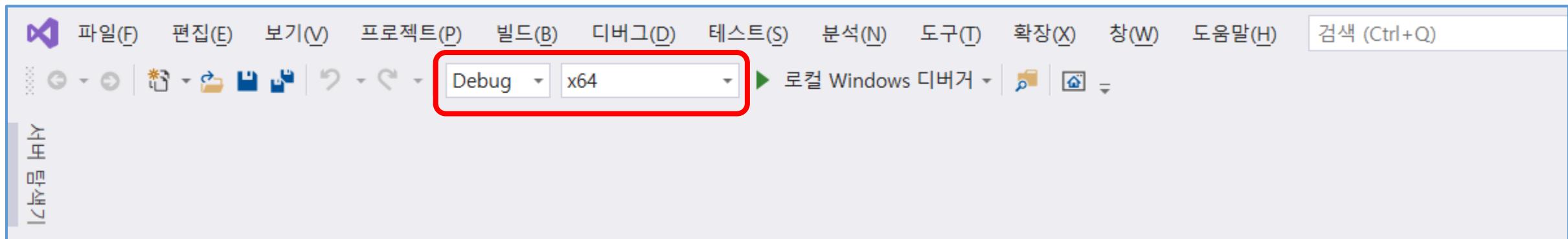


# Visual Studio의 New project 생성(2)



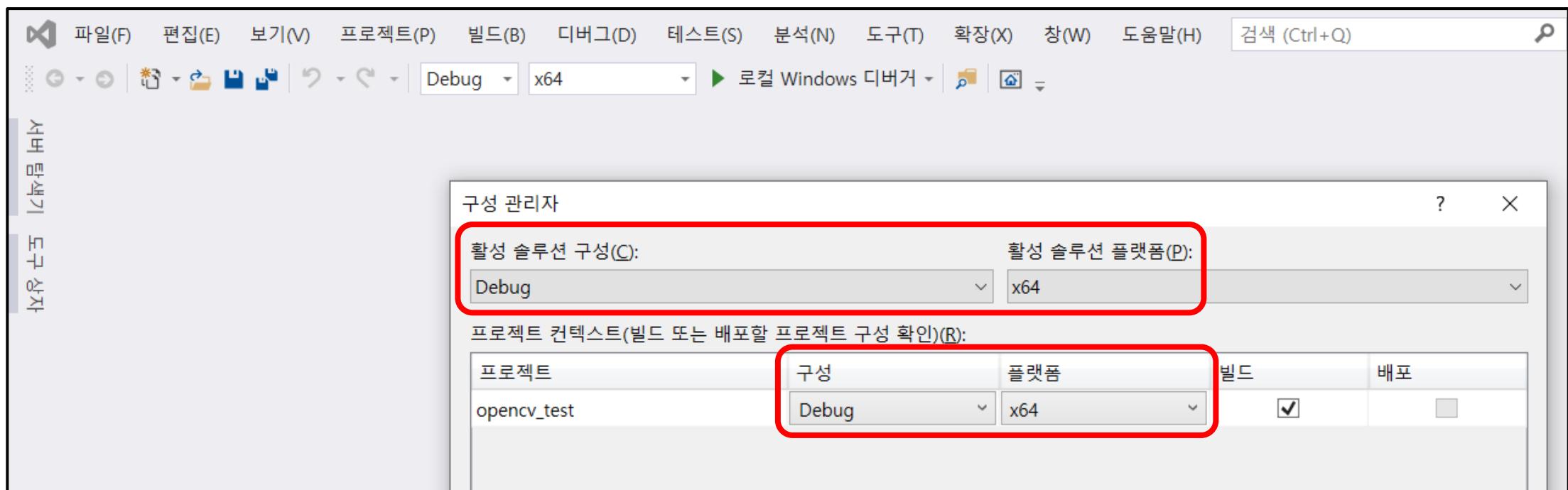
# Visual Studio 프로젝트 구성 설정(1)

- Configuration(구성) 바꾸어 x64 플랫폼으로 설정하기

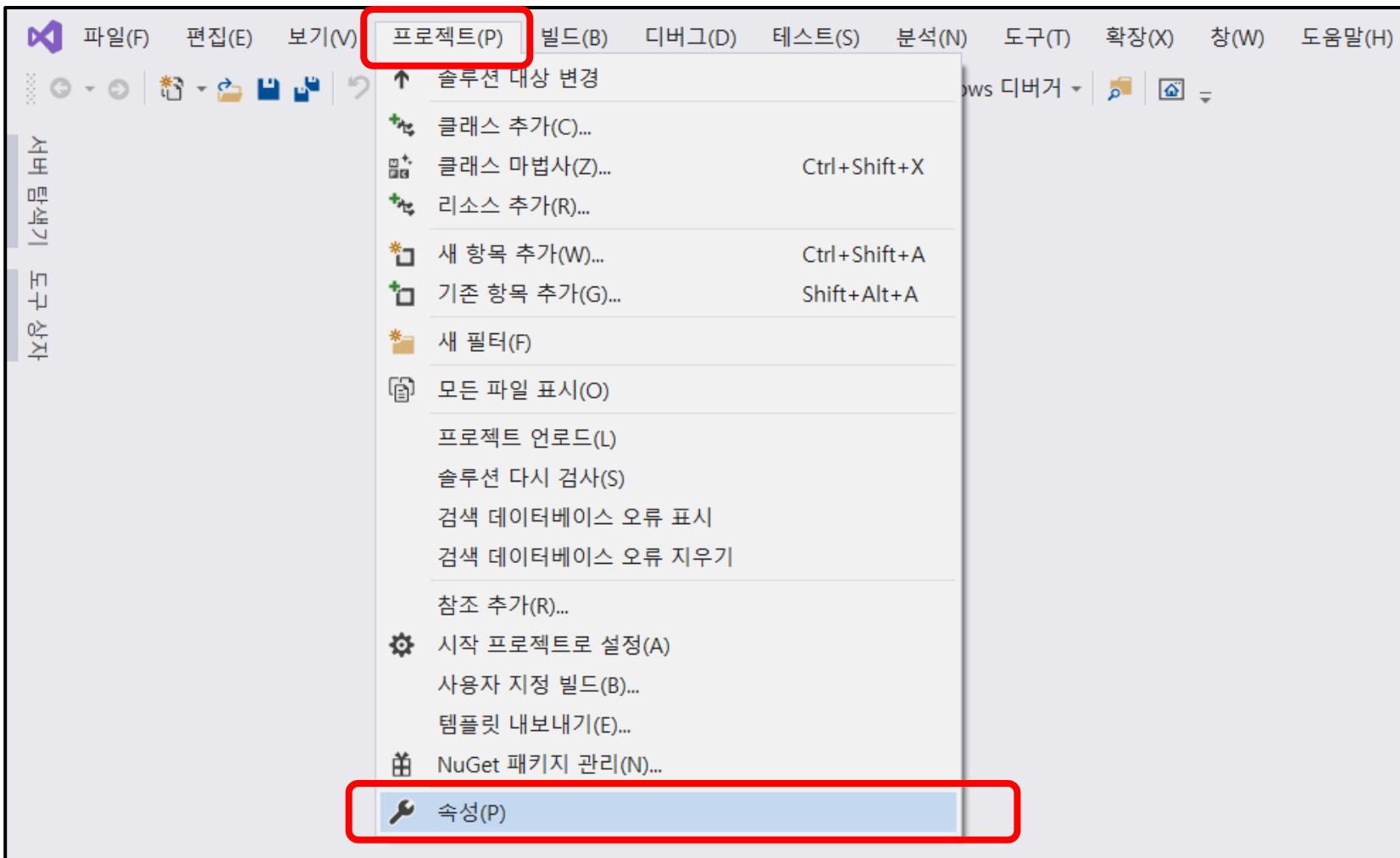


# 1 Visual Studio 프로젝트 구성 설정(2)

- 구성 관리자에서, **Debug** 구성 및 **x64** 플랫폼으로 설정하기  
*or* 빌드 > 구성관리자 메뉴에서 변경할 수도 있다.



# 2 Visual Studio 프로젝트 속성 설정



# 2 VS Project 속성 설정: 헤더파일 경로 설정

- 프로젝트 > 속성 > C++ > 일반  
> "추가 포함 디렉터리"에 다음 2줄 추가

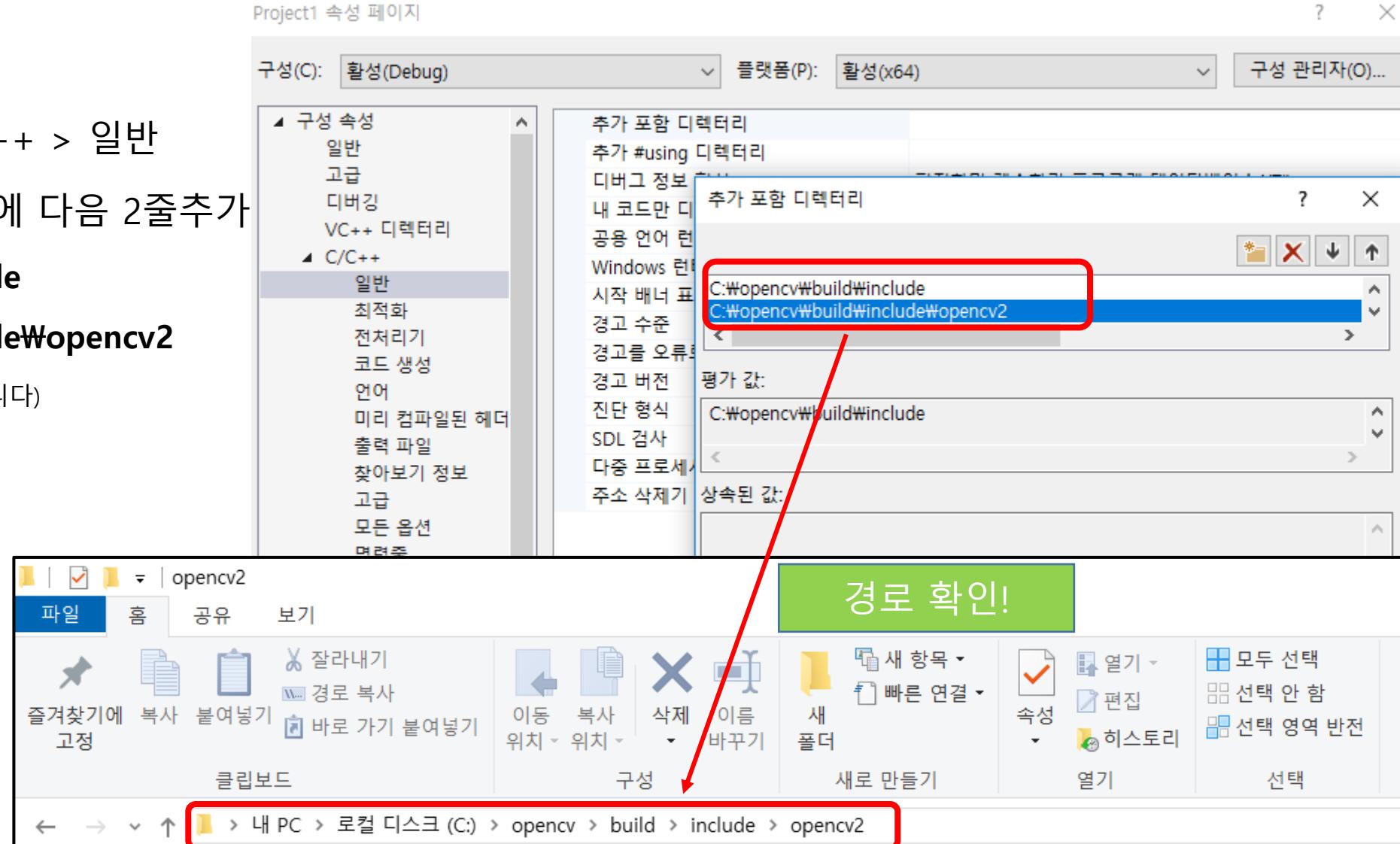
C:\opencv\build\include

C:\opencv\build\include\opencv2

(두 항목은 세미콜론(:)으로 구별합니다)

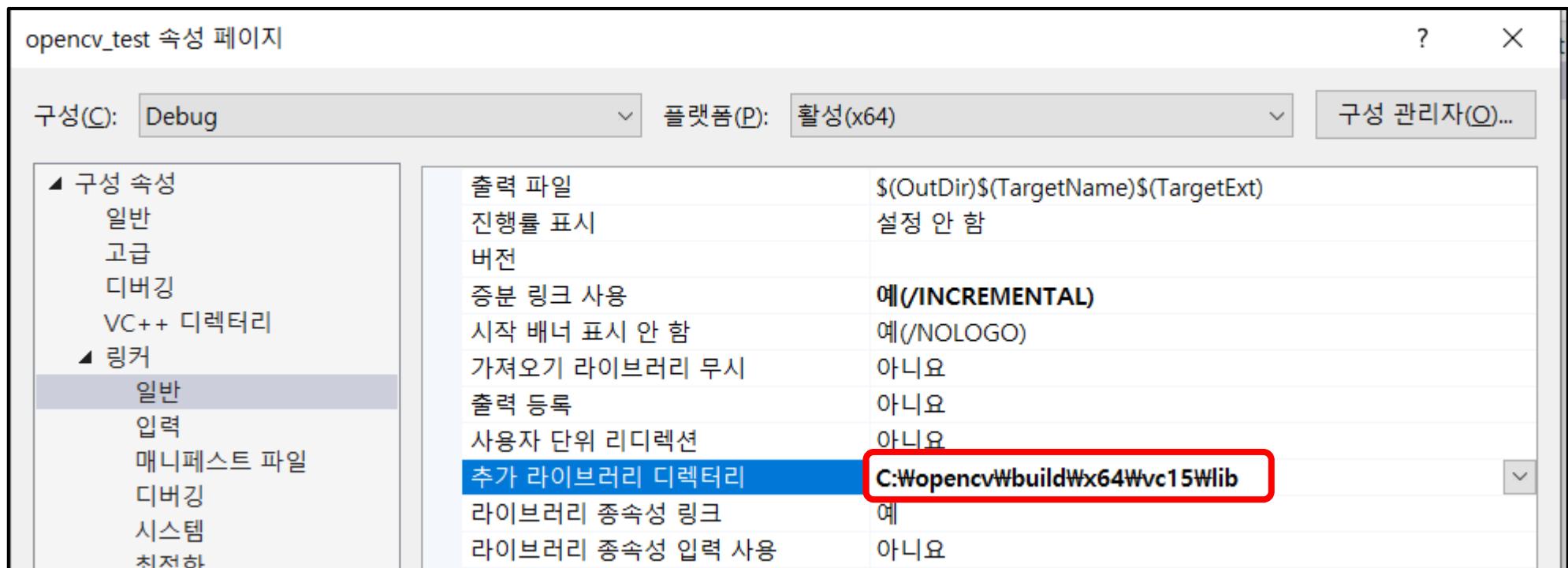
- 추가한 뒤에 "적용"

OpenCV 설치된 경로  
(사용자마다 다를 수 있으니 확인 필요)



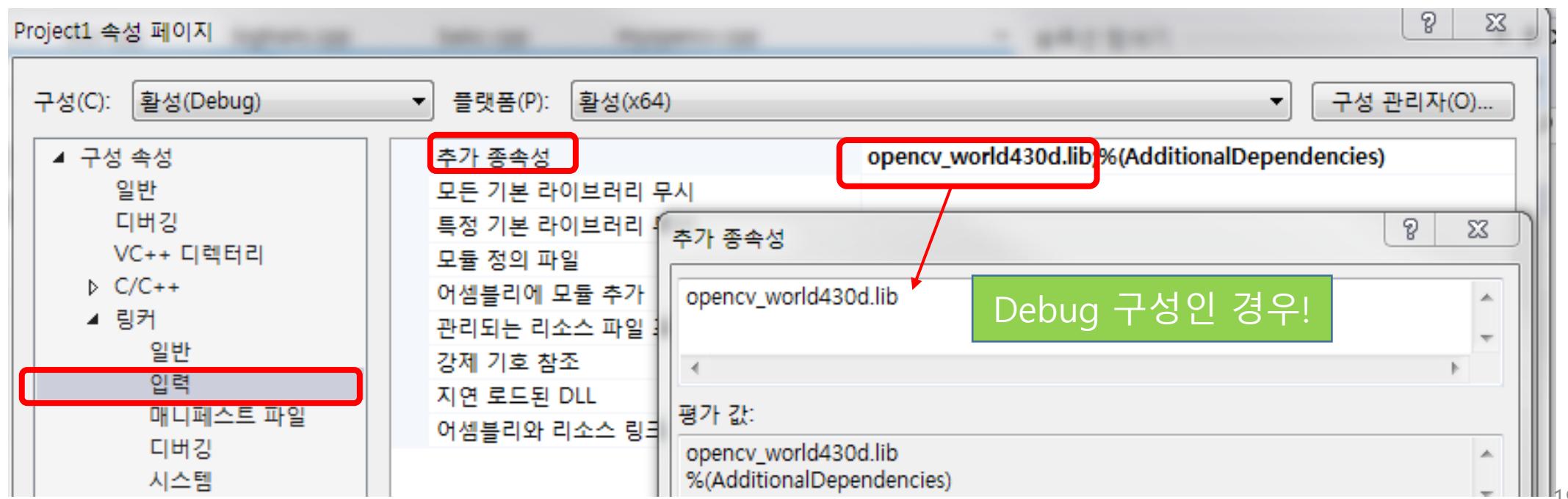
# 3 VS Project 속성 설정: 추가 라이브러리 디렉터리 설정

- 속성: 링커 > 일반 > 추가 라이브러리 디렉터리에 다음 줄 추가, 적용  
→ C:\opencv\build\x64\vc15\lib



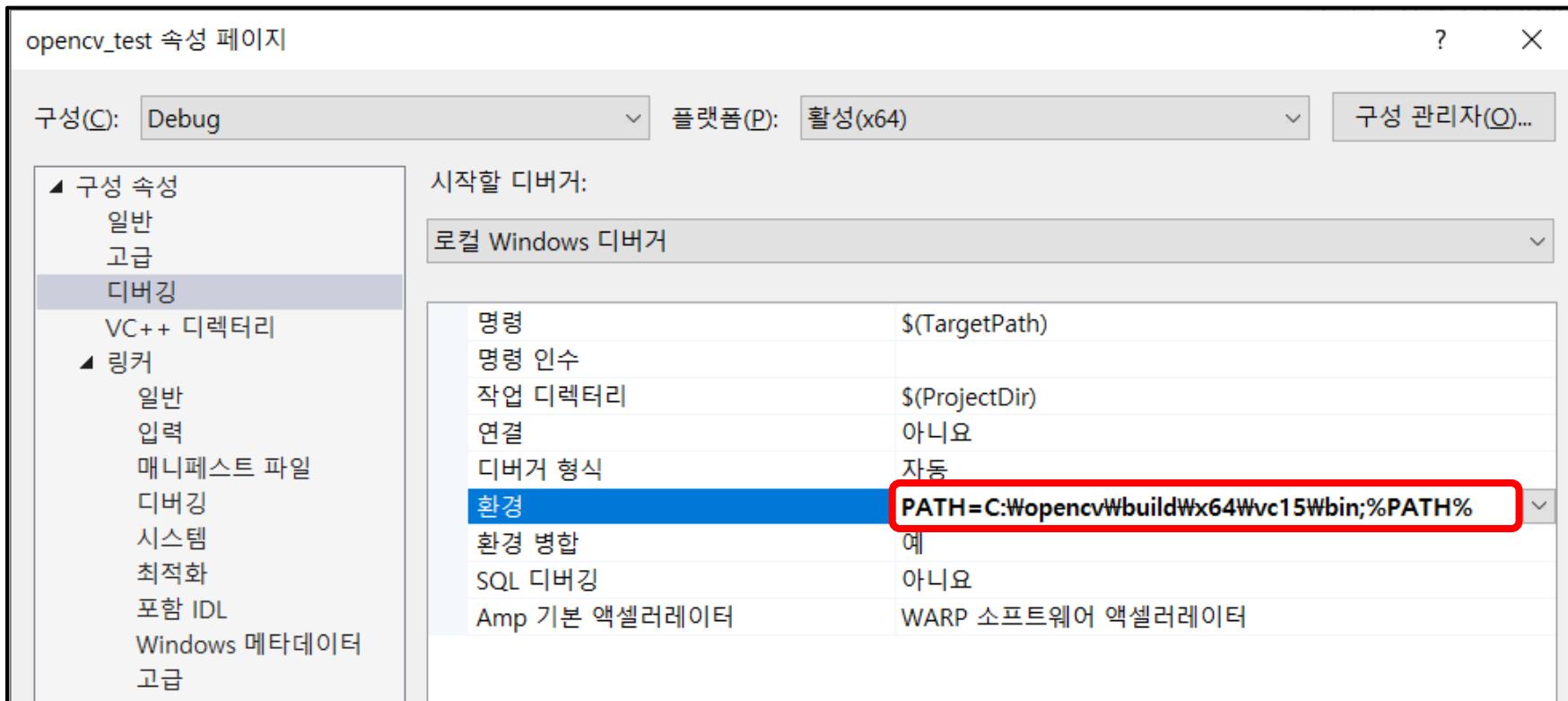
# 4 VS Project 속성 설정: 추가종속성 설정

- 링커 > 입력 > 추가종속성에 다음 줄을 추가하고 “적용”  
`opencv_world430d.lib` (Debug용)  
`opencv_world430.lib` (Release용)
- 주의) 위 두개의 라이브러리 파일을 같은 구성에 모두 추가하면 안됨!**



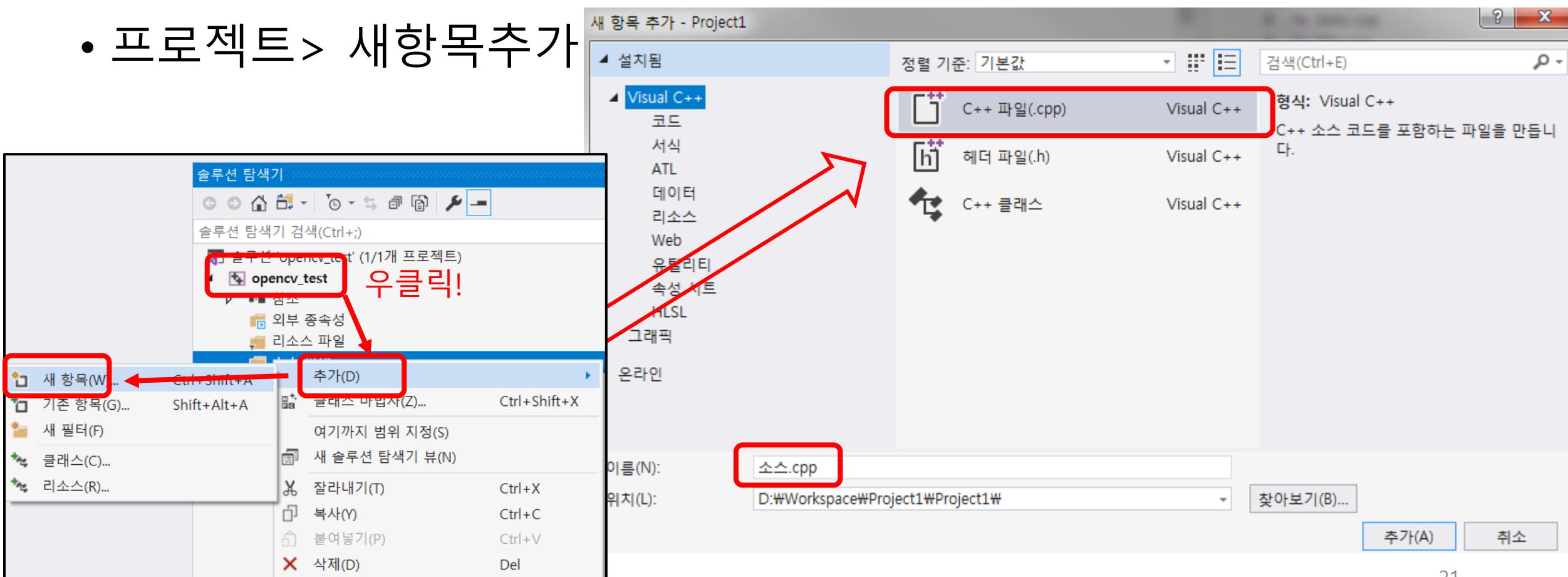
# VS Project 속성 설정: 디버깅 환경 설정

- 구성 속성 > 디버깅 > 환경에 다음 줄을 추가하고 “적용”  
PATH=C:\opencv\build\x64\vc15\bin;%PATH%



# 확인하기 : New 소스 file 생성 (소스.cpp)

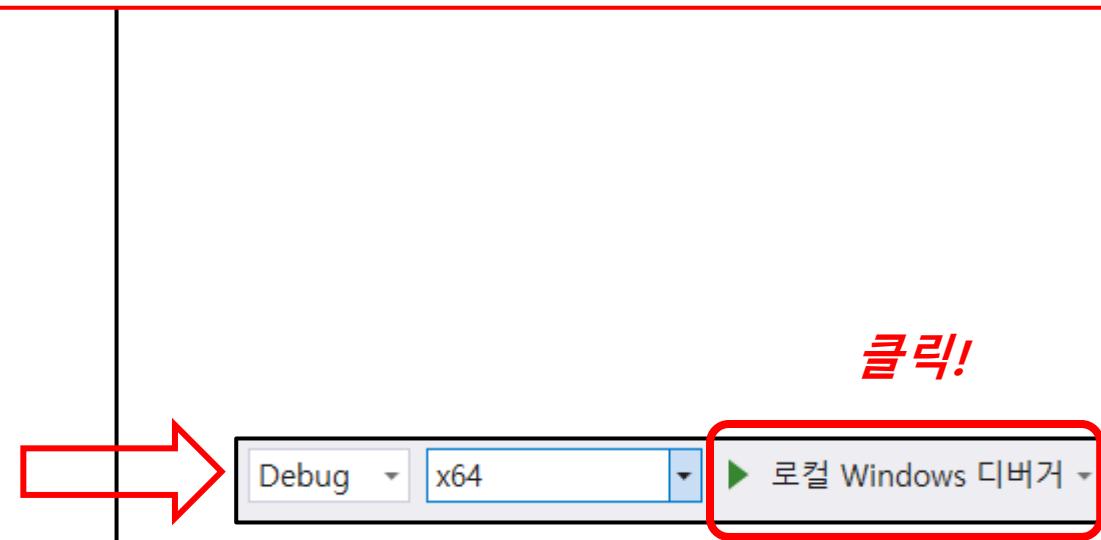
- 솔루션 탐색기 > 프로젝트 선택 후 마우스 우클릭 > 추가 > 새항목  
or
- 프로젝트 > 새항목추가



# 확인하기: 예제 코드 cpp 파일 입력 > 컴파일 및 실행

```
src.cpp* x
opencv_test
1 #include <iostream>
2 #include <opencv2/core/core.hpp>
3 #include <opencv2/highgui/highgui.hpp>
4
5 using namespace cv;
6 using namespace std;
7
8 int main(int argc, const char* argv[]) {
9     Mat pic = imread("/Users/jeonghoon/Desktop/lena.png");
10    if (pic.empty()) {
11        cerr << "read fail!" << endl;
12        exit(-1);
13    }
14    imshow("Pic", pic);
15    waitKey(0);
16
17    return 0;
18 }
```

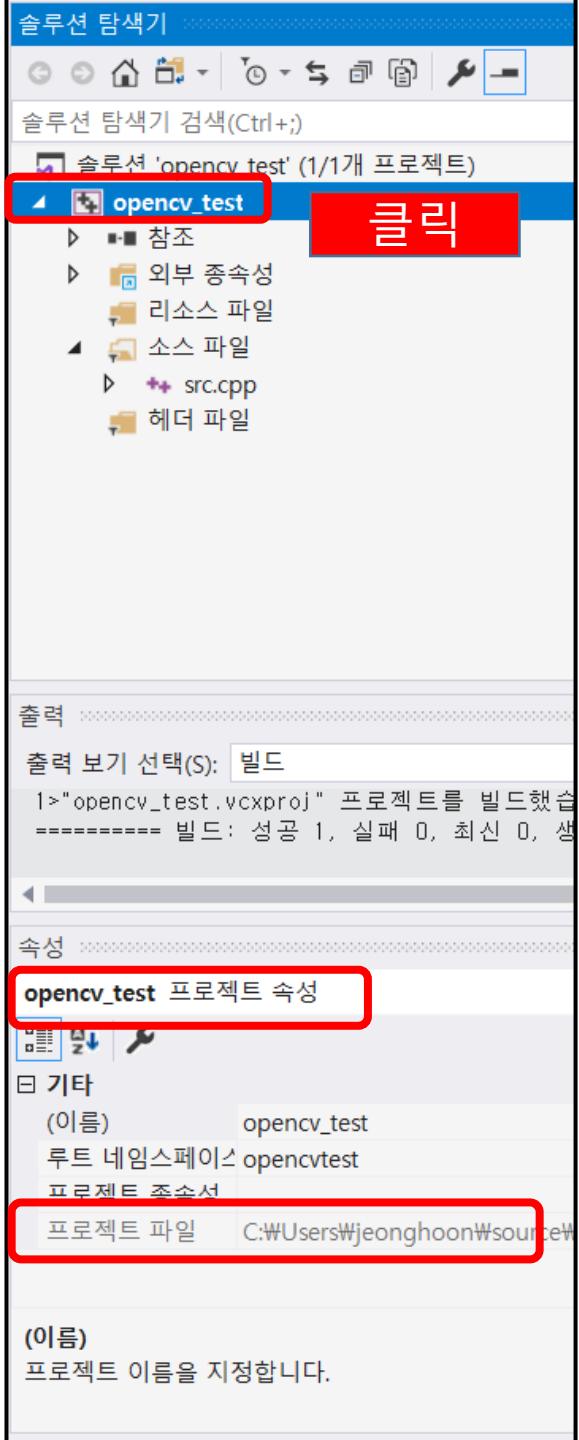
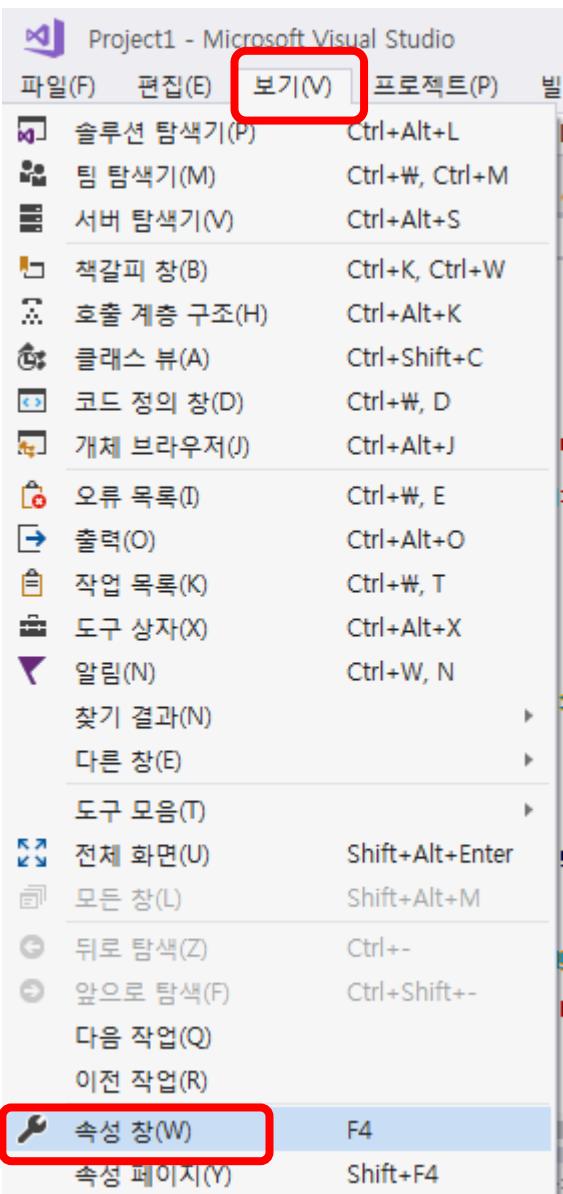
데스크탑에 있는 'lena.png'라는 예시 이미지 파일의 절대경로임.  
(예시 코드에서 이 부분은, 각자 환경에 맞게 변경 가능)  
현재 프로젝트 폴더에 이미지 파일이 있는 경우에는, 파일 이름만 적으면 됨



# \*현재 프로젝트의 생성 경로 확인 방법

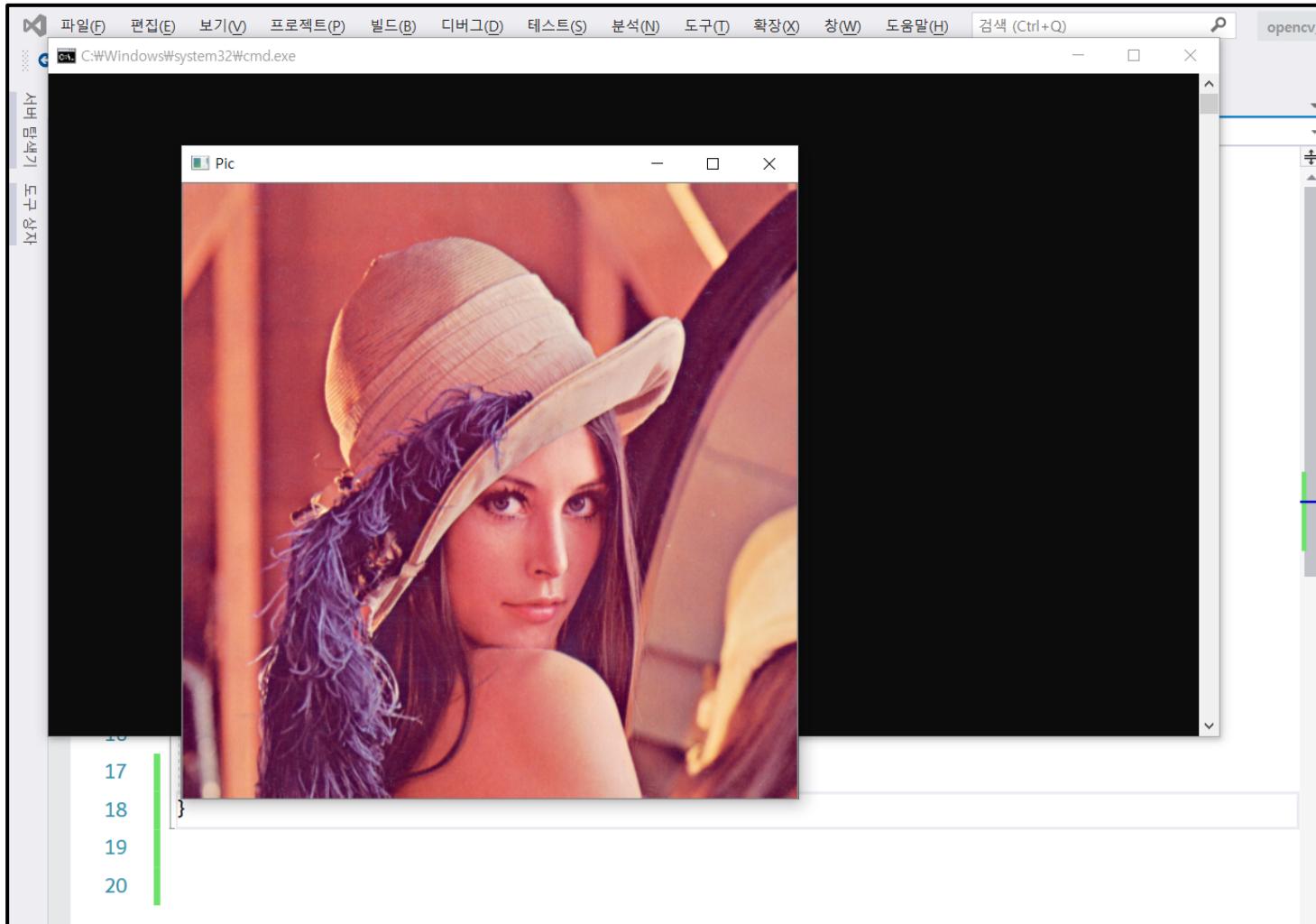
1. 보기 메뉴 > 속성 창 (or F4)
  2. 솔루션 탐색기 > 프로젝트명  
클릭
- 프로젝트 속성 창에서 “프로젝트 파일” 항목에 경로 확인 가능

이 경로에 lena.png 파일을 복사해 두면  
코드에서 이미지 파일 참조시 이름만으로 가능



# 확인하기: 예제 코드 실행 결과

아래 화면과 같이, 이미지 파일이 정상적으로 출력되면 문제없이 설치가 완료된 것임.



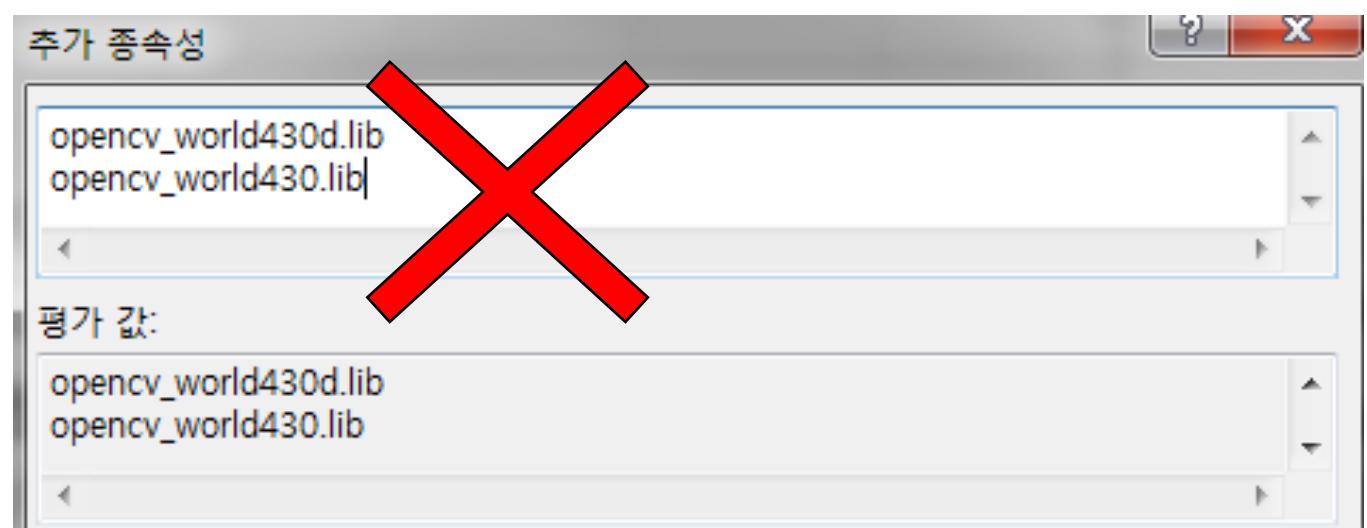
# OpenCV 컴파일, 실행 오류 해결하기

## Cases

Windows 10,  
Visual Studio 2019,  
OpenCV version 4.3.0,  
Debug 구성, 64 bit 플랫폼(x64)을  
기준으로

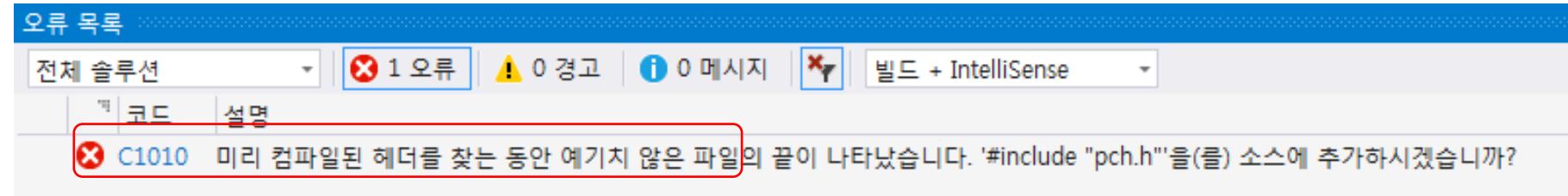
# \* OpenCV가 Visual Studio에서 컴파일 또는 실행이 안될 때 흔한 이유

- Debug mode에서 프로젝트 속성에서 지정된 OpenCV의 lib 파일명에 **d**글자가 포함되어 있는지 확인할 것  
(예: opencv\_world430**d**.lib)
- 프로젝트의 folder 명 혹은 경로 중에 한글이 포함되지 않도록 할 것
- Debug mode를 Release mode로 바꾸어 볼 것
  - 주의) Release mode에서 Property를 설정해야 함
- 프로젝트 속성에서 Release 용 lib 파일(예: *opencv\_world430.dll*)과 debug용 lib 파일(예:*opencv\_world430d.dll*) 두가지 모두가 속성의 **링커 > 입력 > 추가 종속성**에 지정된 경우 → 현재의 모드에 맞게 하나만 추가종속성에 지정해야 함.

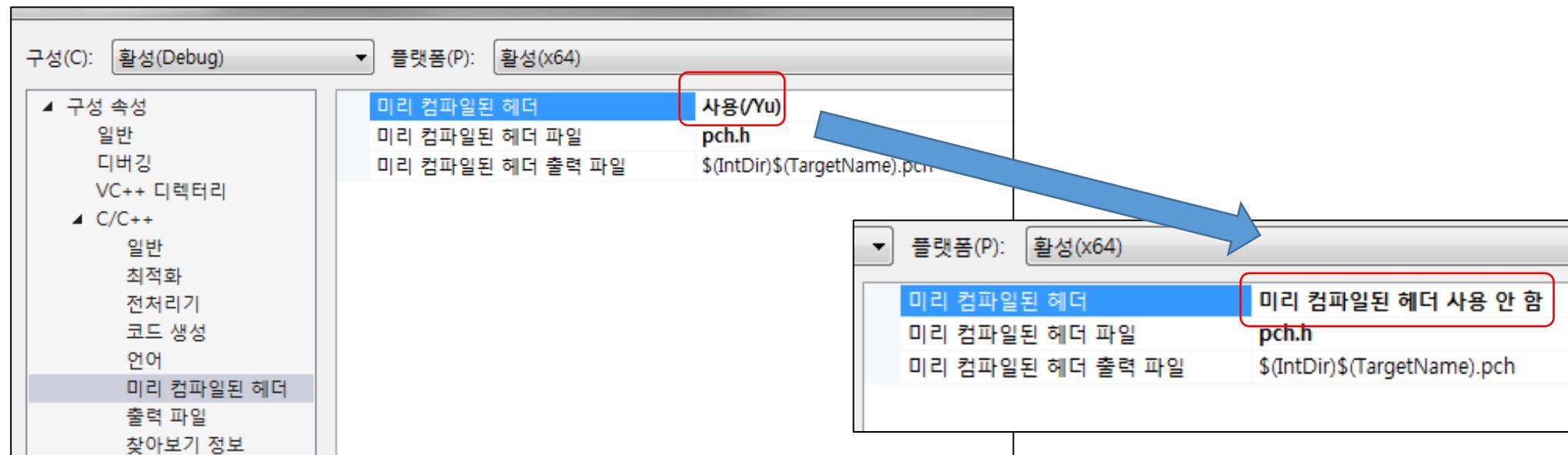


# \* 컴파일시 “미리 컴파일된 헤더파일 (precompiled header)”을 찾는 오류

오류 메시지:



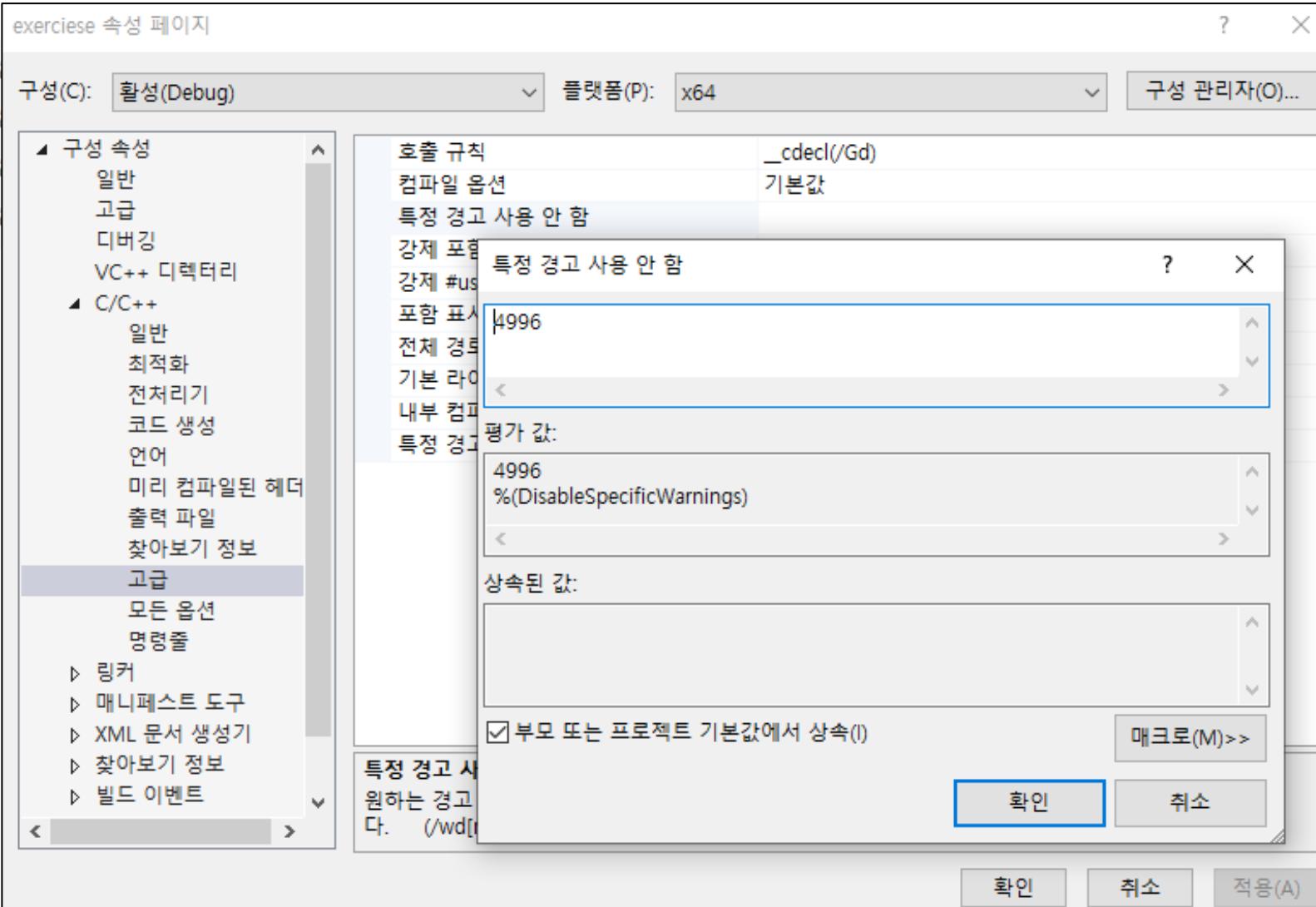
- **프로젝트 속성 > C/C++ > 미리 컴파일된 헤더 > 미리컴파일된 헤더 사용 안함으로 설정 변경**



# \* 컴파일시 sprintf 및 sscanf 함수 사용 오류

- C4996 'sprintf': This function or variable may be uns...
- C4996 'sprintf': This function or variable may be uns...
- C4996 'sprintf': This function or variable may be uns...
- C4996 'sprintf': This function or variable may be uns...

- C4996 Error 무시하기  
→ 설정 > C/C++ > 고급 > 특정 경고 무시하기: 4996 오류  
무시하도록 설정



# \*실행시 “오류:DLL 파일을 찾을수 없습니다.”

1. 적합한 버전의 OpenCV DLL 파일을 찾음  
(예: build/x64/vc15/bin/opencv\_world3410d.dll)
2. 찾은 DLL 파일을 C:\Windows\System32에 복사해 넣기  
또는

PATH 환경변수에 DLL 파일이 있는 경로를 추가하기

(예: PATH = \$(PATH); C:\opencv\build\x64\vc15\bin)

# \* 실행시 이미지 화일이 Open되지 않을 때

## 1. 이미지 화일의 경로가 올바른지 확인한다.

- 파일 경로 표시는 Project 폴더의 위치를 기준으로 한 상대 경로이다.
- 예컨대, "mysample.png" 라고 적으면 현재 프로젝트와 같은 폴더에 mysample.png 파일이 반드시 있어야 한다.

## 2. 파일 확장자가 올바른지 확인한다.

\* 이미지 파일의 확장자는 jpg, jpeg, png, tif, bmp 등 매우 다양하다. 파일 이름뿐 아니라 파일 확장자까지 정확해야 파일이 정상적으로 열린다.

## 3. 경로가 잘못되어 (혹은 파일 이름이 잘못되어) 파일을 open하지 못하는 경우를 대비하여 예외 처리 코드를 삽입한다.

```
Mat img = imread("sample.png",1);
```

```
if(img.empty()) return;
```

## 4. 경로가 올바름에도 불구하고 열리지 않으면 파일 Size가 0은 아닌지, 파일에 대한 접근 경로가 있는지 확인한다.

# \*실행시 동영상 화일이 Open되지 않을 때

## 1. 동영상 화일의 경로가 올바른지 확인한다.

- 파일 경로 표시는 Project 폴더의 위치를 기준으로 한 상대 경로이다.
- 예컨대, "myvideo.mp4" 라고 적으면 현재 프로젝트와 같은 폴더에 myvideo.mp4 파일이 있어야 한다.

## 2. openCv 해당 버전의 ffmpeg decoder를 복사한다.

1. OpenCV의 build/x64/vc15/bin 폴더아래 아래 ffmpeg DLL 파일을 찾는다  
**"opencv\_videoio\_ffmpeg430\_64.dll"**
2. ffmpeg DLL 파일을 현재 프로젝트 폴더 또는  
C:Windows/System32 폴더 아래에 복사한다.

# 부록2: Mac OS X에서 OpenCV 사용하기

# 1. 홈브루(Homebrew)로 OpenCV 설치

- 홈브루(Homebrew)가 설치되어 있지 않다면, 먼저 **homebrew**를 설치해줍니다.
- 아래 명령어를 복사하여 터미널에 입력하면 **homebrew** 설치가 시작됩니다.

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

- 설치가 완료되면 이어서 homebrew로 **opencv**를 설치합니다.

터미널에,

```
brew update  
brew install opencv  
brew upgrade opencv
```

명령어를 차례로 입력하여 **opencv**를 설치합니다.

## 2. Xcode 설치

App store 에서 Xcode를 설치합니다.

The screenshot shows the Xcode app page on the App Store. At the top left is a back arrow icon. The main image features a hammer resting on a blue blueprint with architectural drawings and the word "XCODE" at the bottom right. To the right of the image, the app's name "Xcode" is displayed in large bold letters, followed by the text "개발자 도구" and "Apple". Below the image, the rating "2.0 ★★★☆☆" is shown, along with "771개의 평가". To the right of the rating is the rank "# 1" and the category "개발자 도구". Further to the right are two buttons: a blue "업데이트" button with a red border and a blue ellipsis "...". Below these buttons, the age rating "4+" and "연령" are listed. A note at the bottom states "macOS 10.15.2 이상이 필요합니다." (macOS 10.15.2 이상이 필요합니다.).

Xcode  
개발자 도구  
Apple

2.0 ★★★☆☆  
771개의 평가

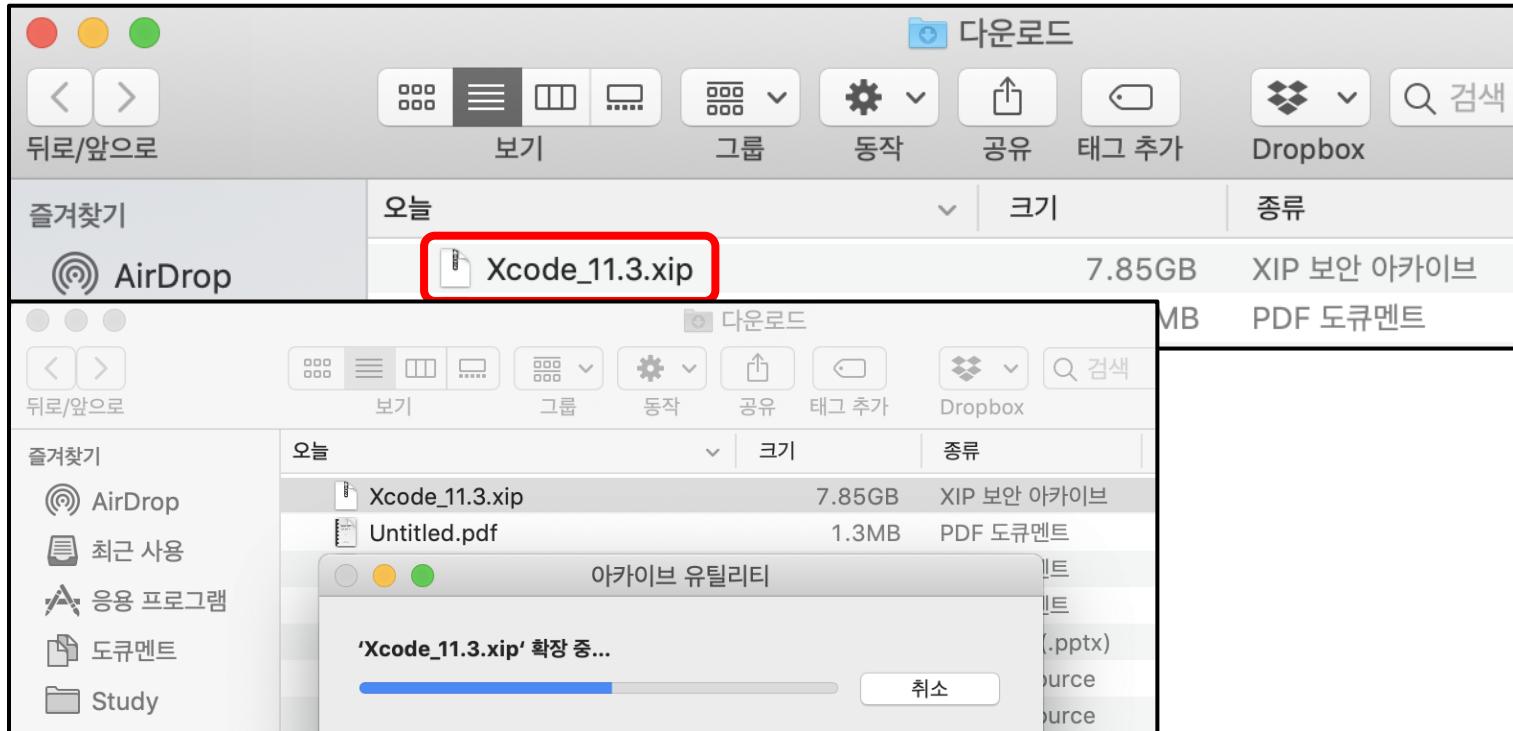
# 1  
개발자 도구

4+  
연령

macOS 10.15.2 이상이 필요합니다.

## 2. Xcode 설치

다운로드 받은 Xcode\_11.3.xip를 “더블클릭”하여 xip파일을 열어 줍니다.

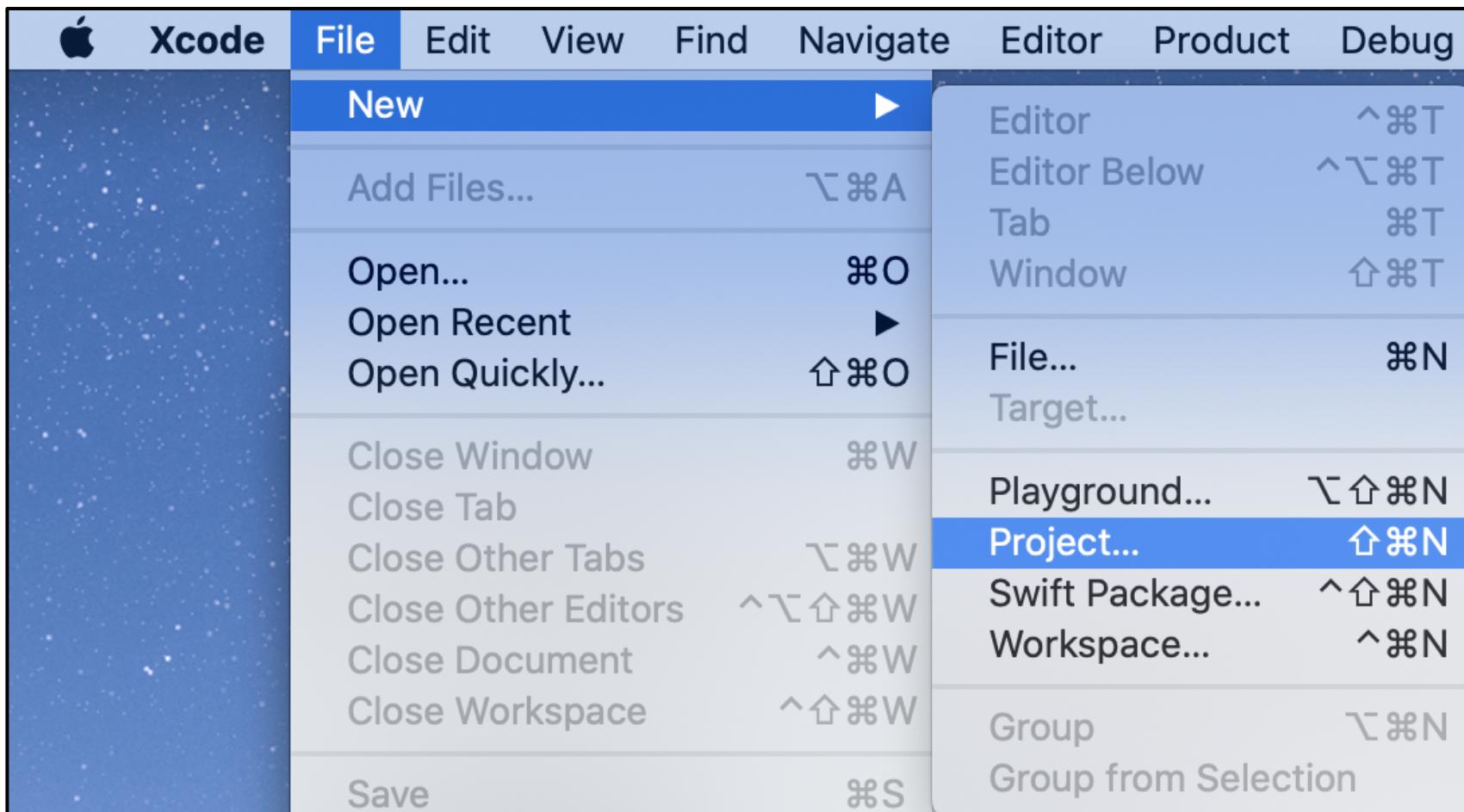


Xcode를 실행하기 위해 OS 버전  
업데이트 등 시간이 소요될 수 있  
으니 주의하세요!

위치	2019	
원격 디스크	Zoom.pkg	12.1MB 설치 프…램 패키지
네트워크	Xcode	16.17GB 응용 프로그램

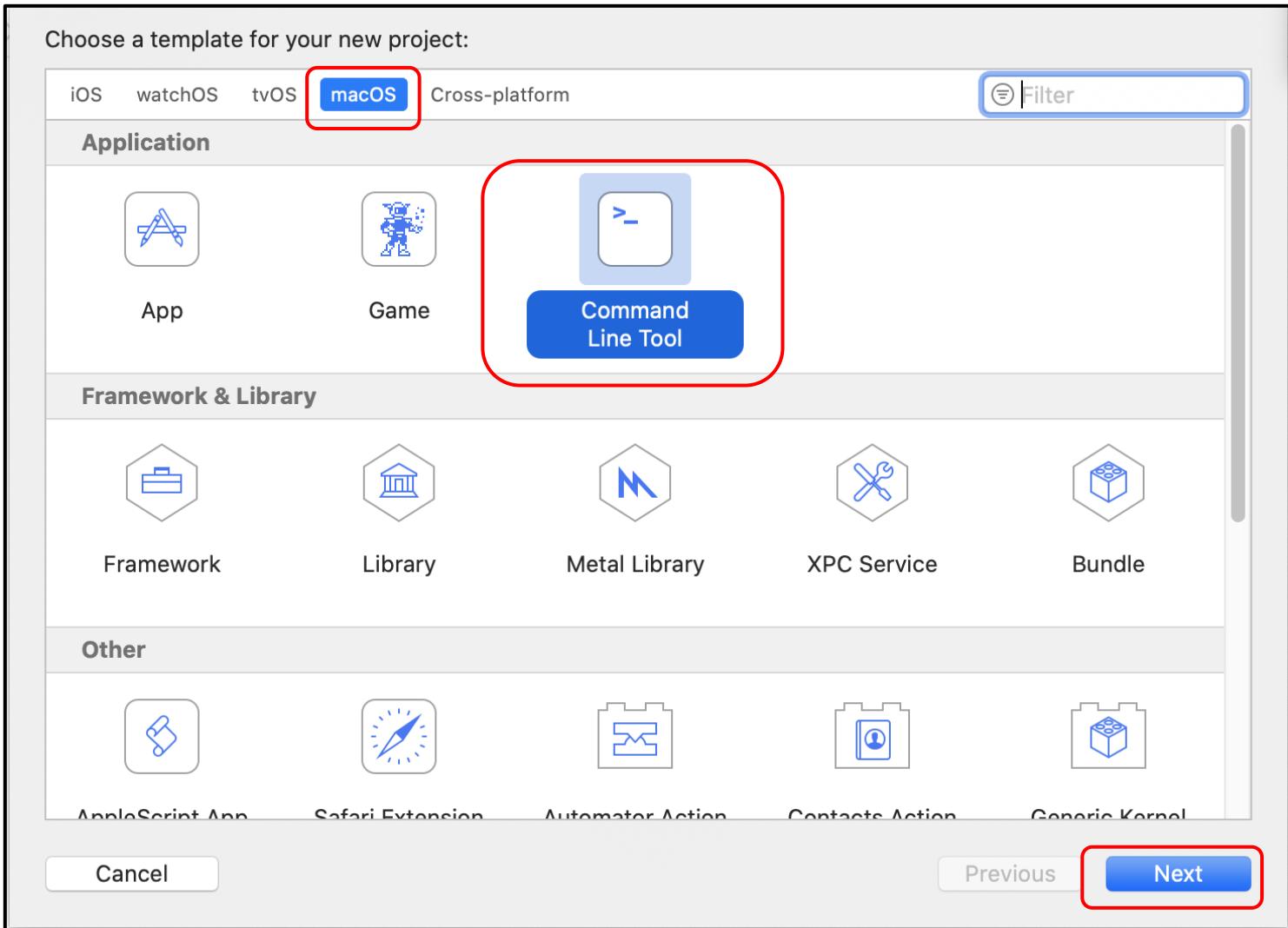
### 3. Xcode에서 프로젝트 생성

설치가 완료되면 Xcode를 실행한 후, File > New > Project를 클릭합니다.



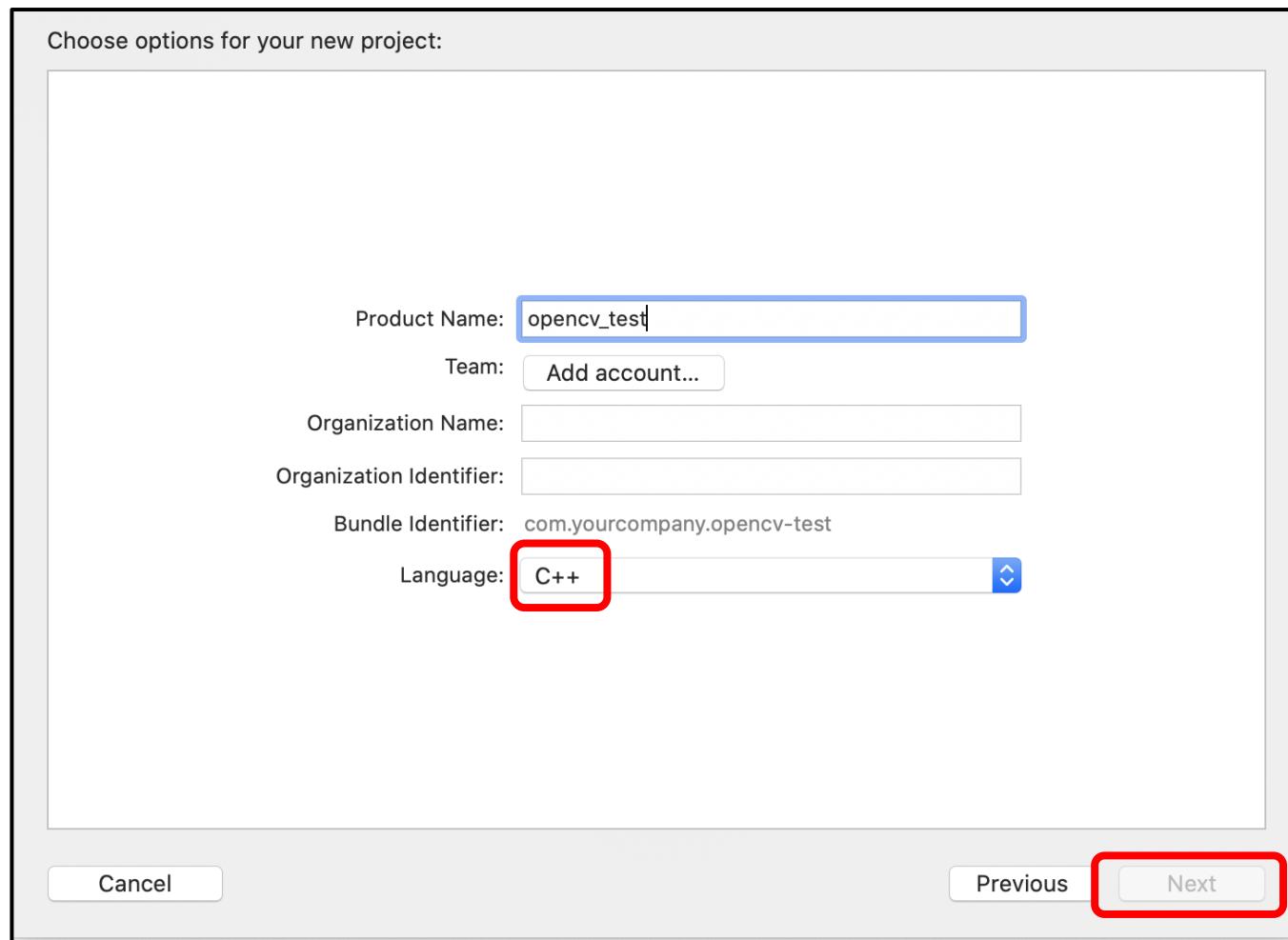
### 3. Xcode에서 프로젝트 생성

'macOS → command Line Tool → Next' 차례대로 클릭



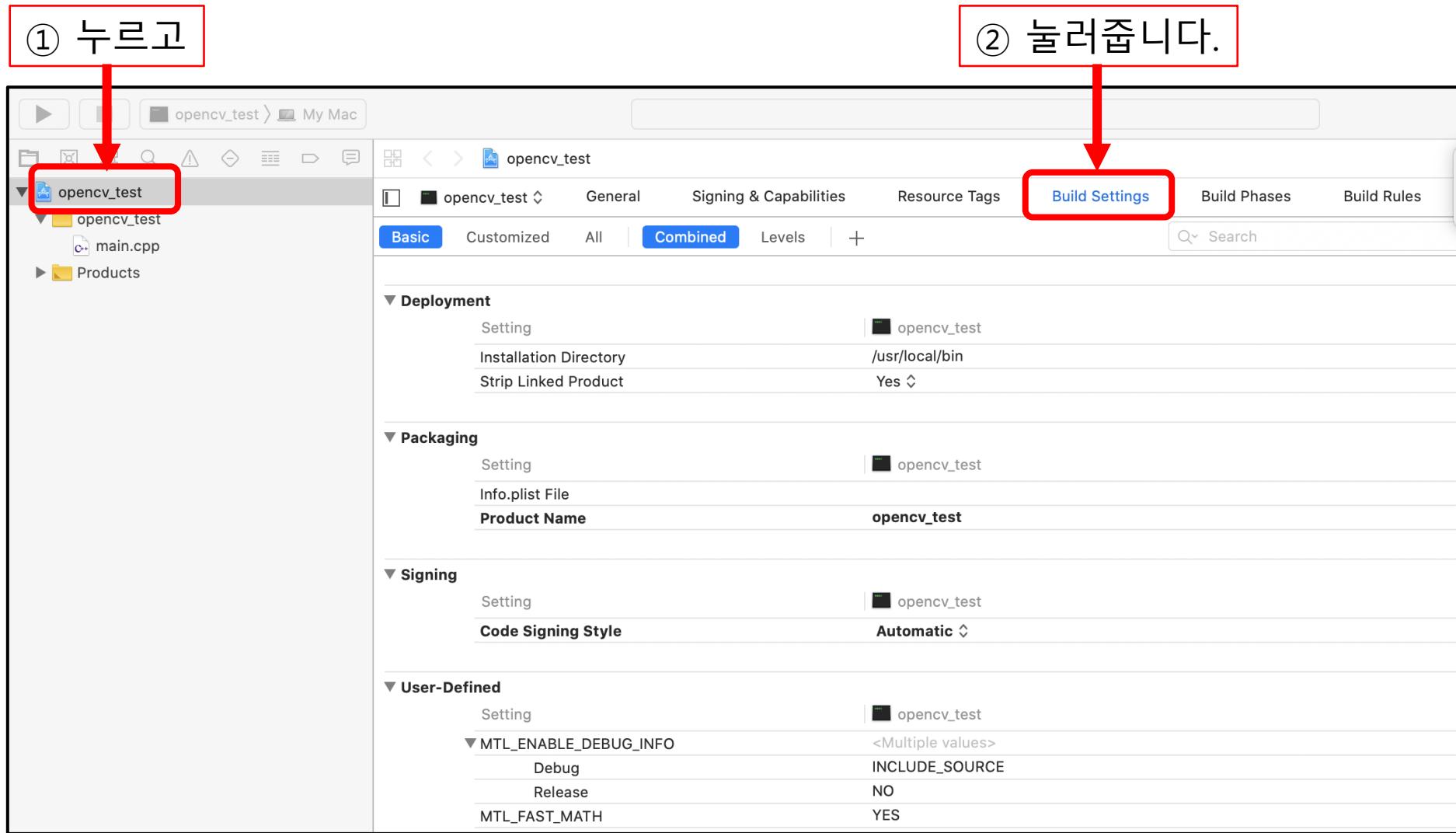
### 3. Xcode에서 프로젝트 생성, 환경 설정

원하는 'Product Name' 입력, 'Language' → 'C++'로 설정한 후 'Next' 클릭



# 4. Xcode에서 프로젝트 환경 설정

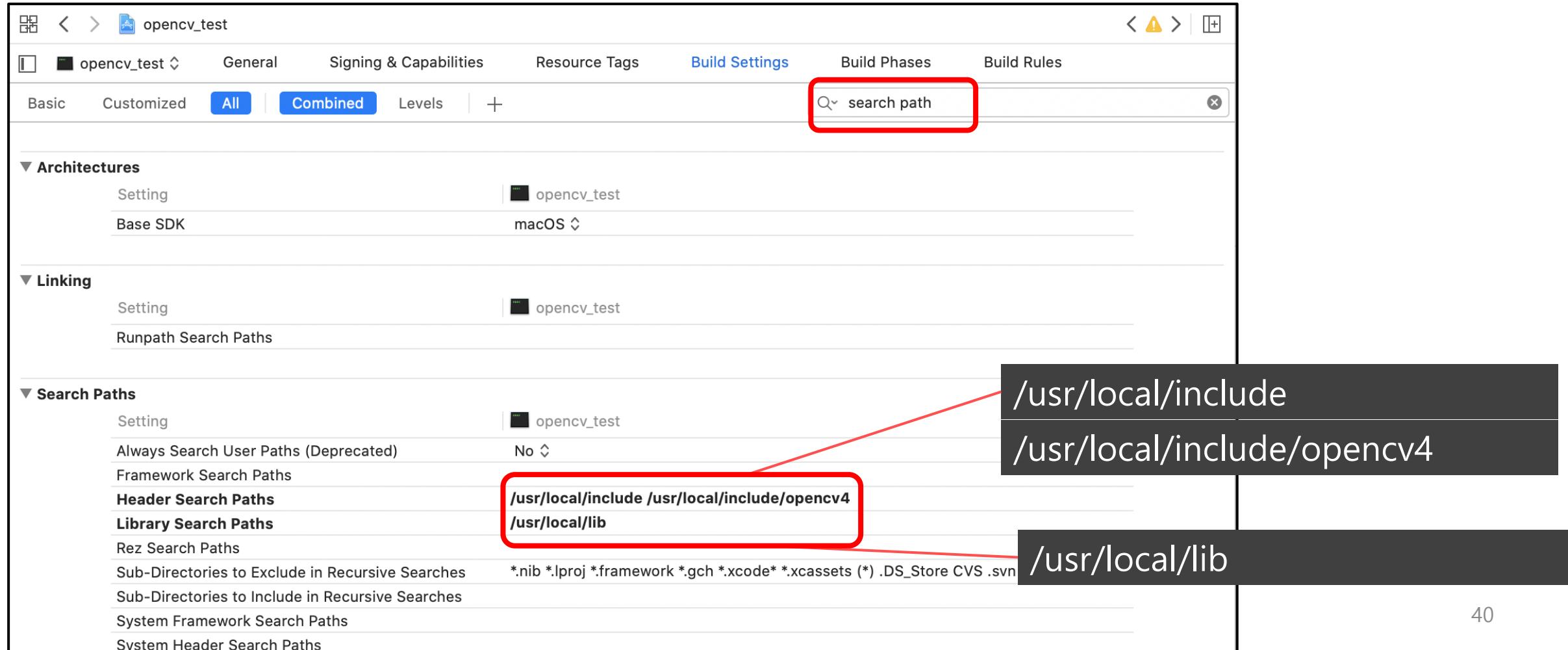
생성된 프로젝트의 환경 설정을 위해 **Build Settings** 탭을 눌러줍니다.



# 4. Xcode 에서 프로젝트 환경 설정

검색창에 'search path' 검색하여

'Search Paths → Header Search Paths, Library Search Paths'에 다음과 같이 입력



# 4. Xcode에서 프로젝트 환경 설정

- Linker Flags 설정값을 알아내기 위해 먼저 설치된 opencv 버전을 알아봅니다.  
터미널에서 아래 명령어 입력한 결과가 **opencv 버전입니다.**

```
ls /usr/local/Cellar/opencv
```

Ex) \$ ls /usr/local/Cellar/opencv

4.3.0\_5

- 다음으로 아래 명령어를 터미널에 입력한 결과를 복사해줍니다.

```
pkg-config --cflags --libs /usr/local/Cellar/opencv/(설치된 opencv 버전)/lib/pkgconfig/opencv4.pc
```

4.3.0\_5

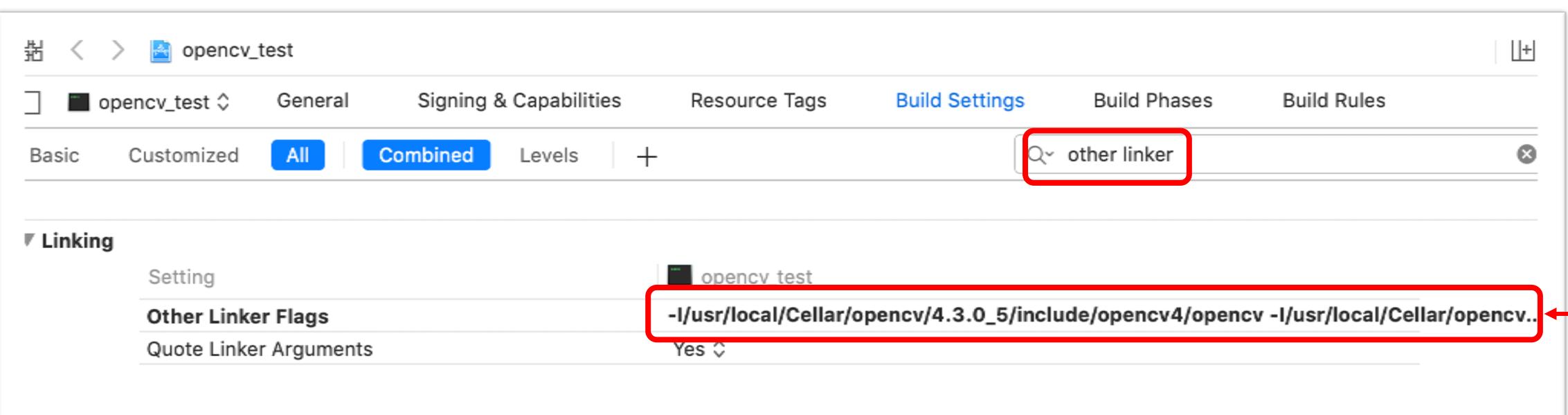
Ex) \$ pkg-config --cflags --libs /usr/local/Cellar/opencv/4.3.0\_5/lib/pkgconfig/opencv4.pc

```
-I/usr/local/Cellar/opencv/4.3.0_5/include/opencv4/opencv -I/usr/local/Cellar/opencv/4.3.0_5/include/opencv4 -L/usr/local/Cellar/opencv/4.3.0_5/lib -lopencv_gapi -lopencv_stitching -lopencv_alphamat -lopencv_aruco -lopencv_bgsegm -lopencv_bioinspired -lopencv_ccalib -lopencv_dnn_objdetect -lopencv_dnn_superres -lopencv_dpm -lopencv_highgui -lopencv_face -lopencv_freetype -lopencv_fuzzy -lopencv_hfs -lopencv_img_hash -lopencv_intensity_transform -lopencv_line_descriptor -lopencv_quality -lopencv_rapid -lopencv_reg -lopencv_rgbd -lopencv_saliency -lopencv_sfml -lopencv_stereo -lopencv_structured_light -lopencv_phase_unwrapping -lopencv_superres -lopencv_optflow -lopencv_surface_matching -lopencv_tracking -lopencv_datasets -lopencv_text -lopencv_dnn -lopencv_plot -lopencv_videostab -lopencv_videoio -lopencv_xfeatures2d -lopencv_shape -lopencv_ml -lopencv_ximgproc -lopencv_video -lopencv_xobjdetect -lopencv_objdetect -lopencv_calib3d -lopencv_imgcodecs -lopencv_features2d -lopencv_flann -lopencv_xphoto -lopencv_photo -lopencv_imgproc -lopencv_core
```

# 4. Xcode에서 프로젝트 환경 설정

다시 Xcode의 Build Settings 검색창에 'other linker' 검색하여  
**'Linking → Other Linker Flags'**에 앞서 복사한 내용 입력

```
-I/usr/local/Cellar/opencv/4.3.0_5/include/opencv -I/usr/local/Cellar/opencv/4.3.0_5/include/opencv4 -L/usr/local/Cellar/opencv/4.3.0_5/lib -lopencv_gapi -lopencv_stitching -lopencv_alphaamat -lopencv_aruco -lopencv_bgsegm -lopencv_bioinspired -lopencv_ccalib -lopencv_dnn_objdetect -lopencv_dnn_superres -lopencv_dpm -lopencv_highgui -lopencv_face -lopencv_freetype -lopencv_fuzzy -lopencv_hfs -lopencv_img_hash -lopencv_intensity_transform -lopencv_line_descriptor -lopencv_quality -lopencv_rapid -lopencv_reg -lopencv_rgbd -lopencv_saliency -lopencv_sfsm -lopencv_stereo -lopencv_structured_light -lopencv_phase_unwrapping -lopencv_superres -lopencv_optflow -lopencv_surface_matching -lopencv_tracking -lopencv_datasets -lopencv_text -lopencv_dnn -lopencv_plot -lopencv_videostab -lopencv_videoio -lopencv_xfeatures2d -lopencv_shape -lopencv_ml -lopencv_ximgproc -lopencv_video -lopencv_xobjdetect -lopencv_objdetect -lopencv_calib3d -lopencv_imgcodecs -lopencv_features2d -lopencv_flann -lopencv_xphoto -lopencv_photo -lopencv_imgproc -lopencv_core
```



# 5. 예제 코드 실행

다음과 같은 예제 코드를 입력한 후, '**command + R**' 을 눌러 실행 결과 확인

데스크탑에 있는 'lena.png'라는 예시 이미지 파일을 출력하는 코드입니다.  
예시 코드 작성해 보실 때, 이 부분은,  
본인이 가지고 있는 이미지 파일을 경로와 함께 넣어주시면 됩니다.

```
#include <iostream>
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace cv;
using namespace std;

int main(int argc, const char * argv[]) {
    Mat pic = imread("절대경로 작성");
    if(pic.empty()) {
        cerr << "read fail!" << endl;
        exit(-1);
    }
    imshow("Pic",pic);
    waitKey(0);

    return 0;
}
```



# 5. 예제 코드 실행 – 파일 상대 경로 사용

## 1) 프로젝트와 동일 폴더에 읽을 파일 넣은 경우

```
#include <iostream>
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace cv;
using namespace std;

int main(int argc, const char * argv[]){
    Mat pic = imread("lena.png");
    if(pic.empty()){
        cerr << "read fail!" << endl;
        exit(-1);
    }
    imshow("Pic", pic);
    waitKey(0);

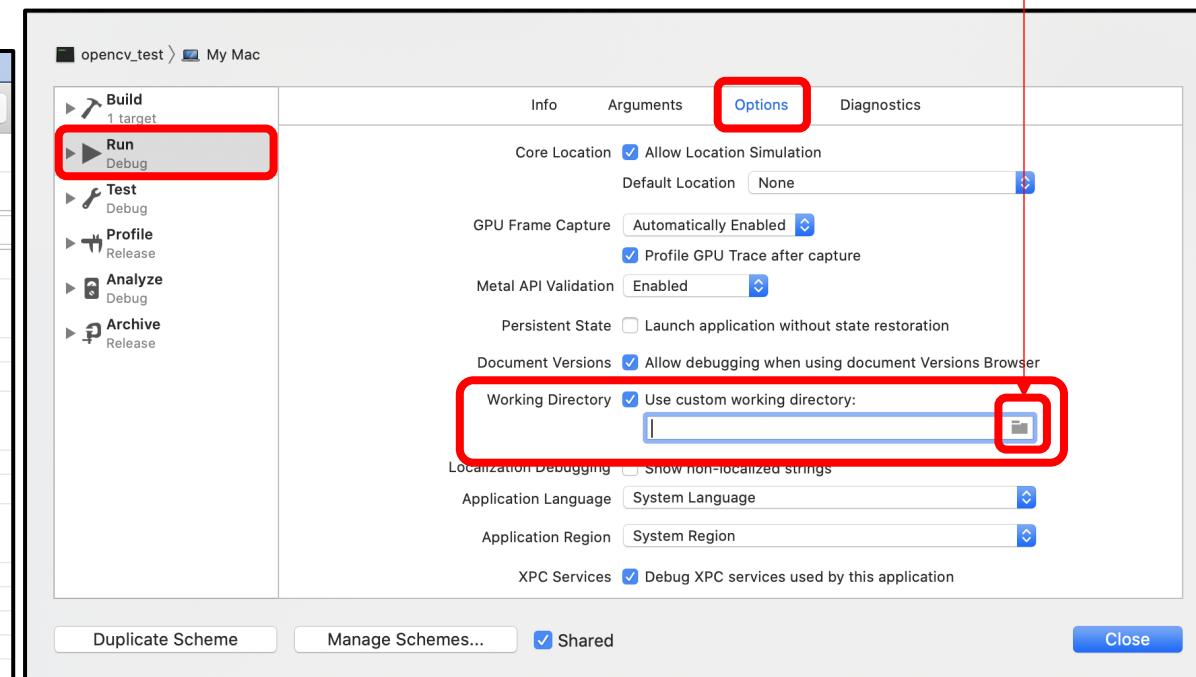
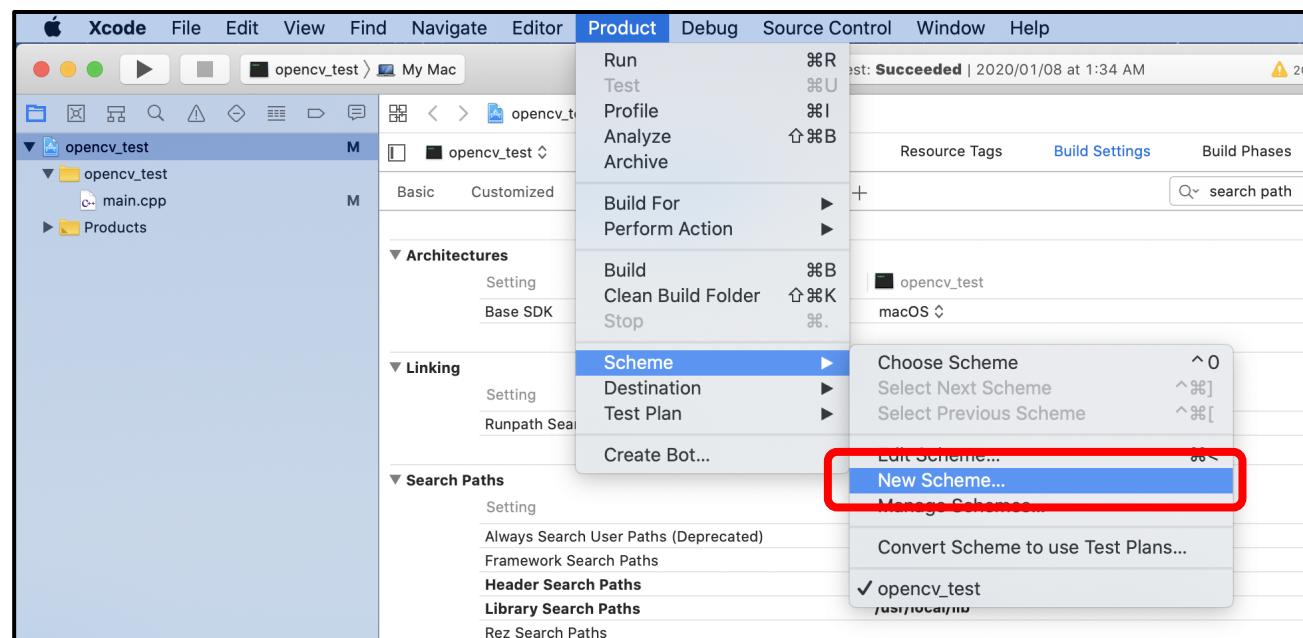
    return 0;
}
```

# 5. 예제 코드 실행 – 파일 상대 경로 사용

## 1) 프로젝트와 동일 폴더에 읽을 파일 넣은 경우

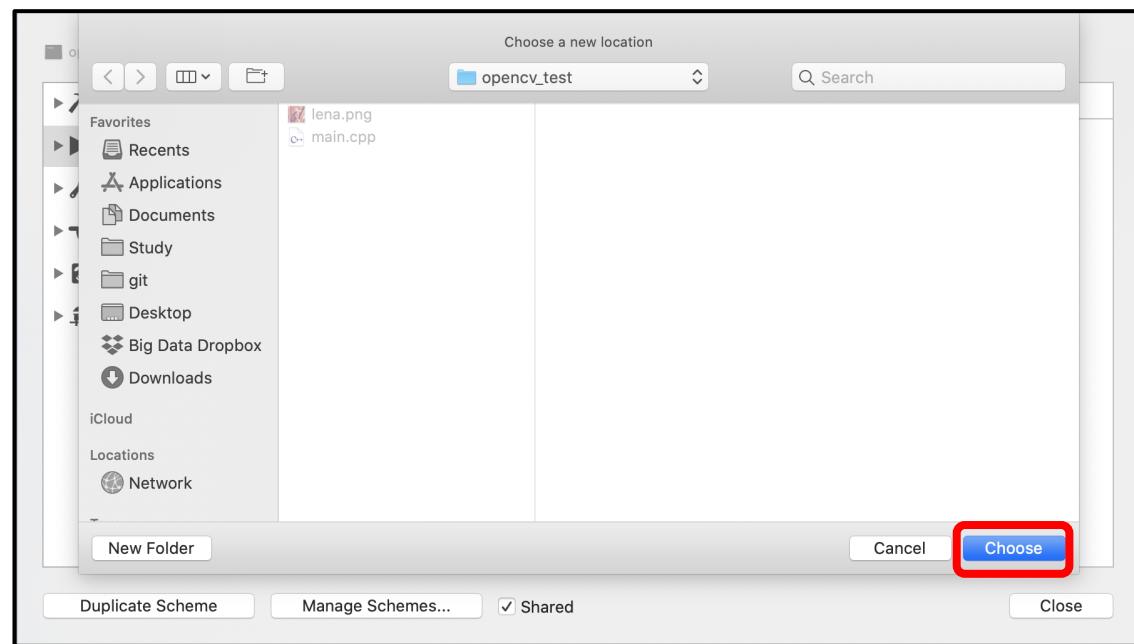
Product > Scheme > New Scheme

폴더 모양의 아이콘을 눌러서  
프로젝트가 들어 있는 폴더를 선택

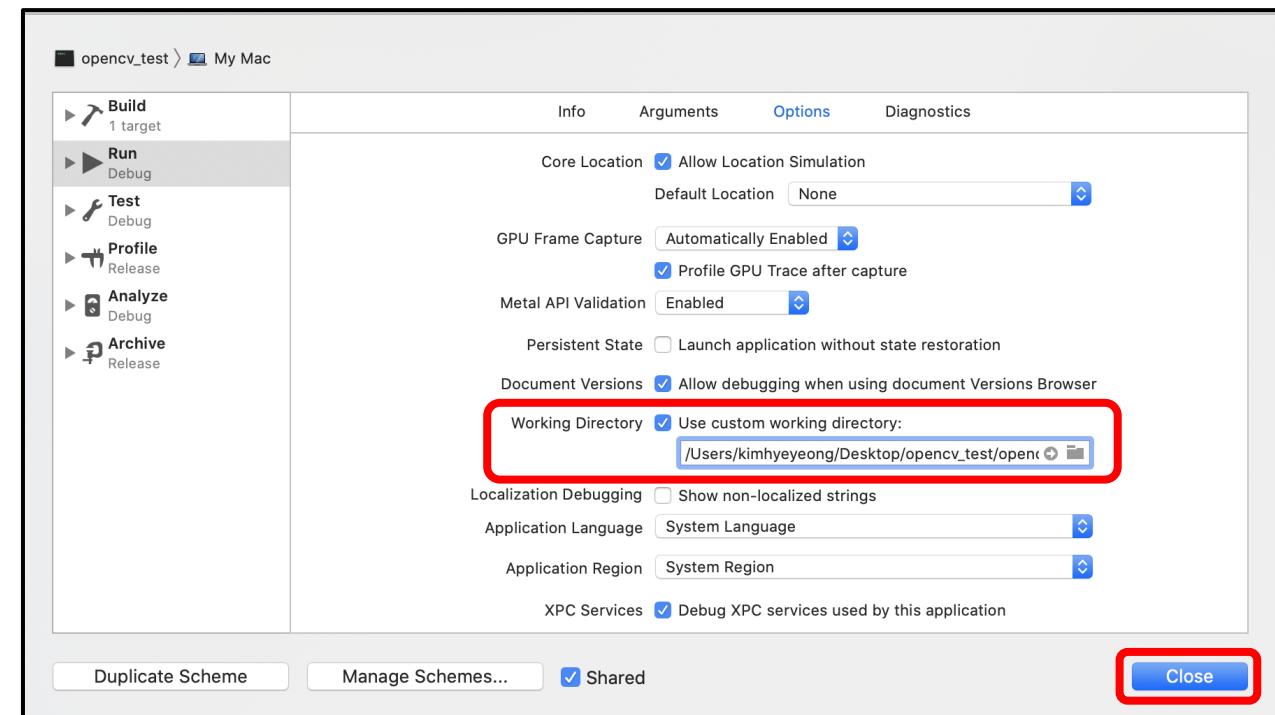


# 5. 예제 코드 실행 – 파일 상대 경로 사용

## 1) 프로젝트와 동일 폴더에 읽을 파일 넣은 경우



해당 프로젝트가 들어 있는 폴더를 선택



# 5. 예제 코드 실행 – 파일 절대 경로 사용

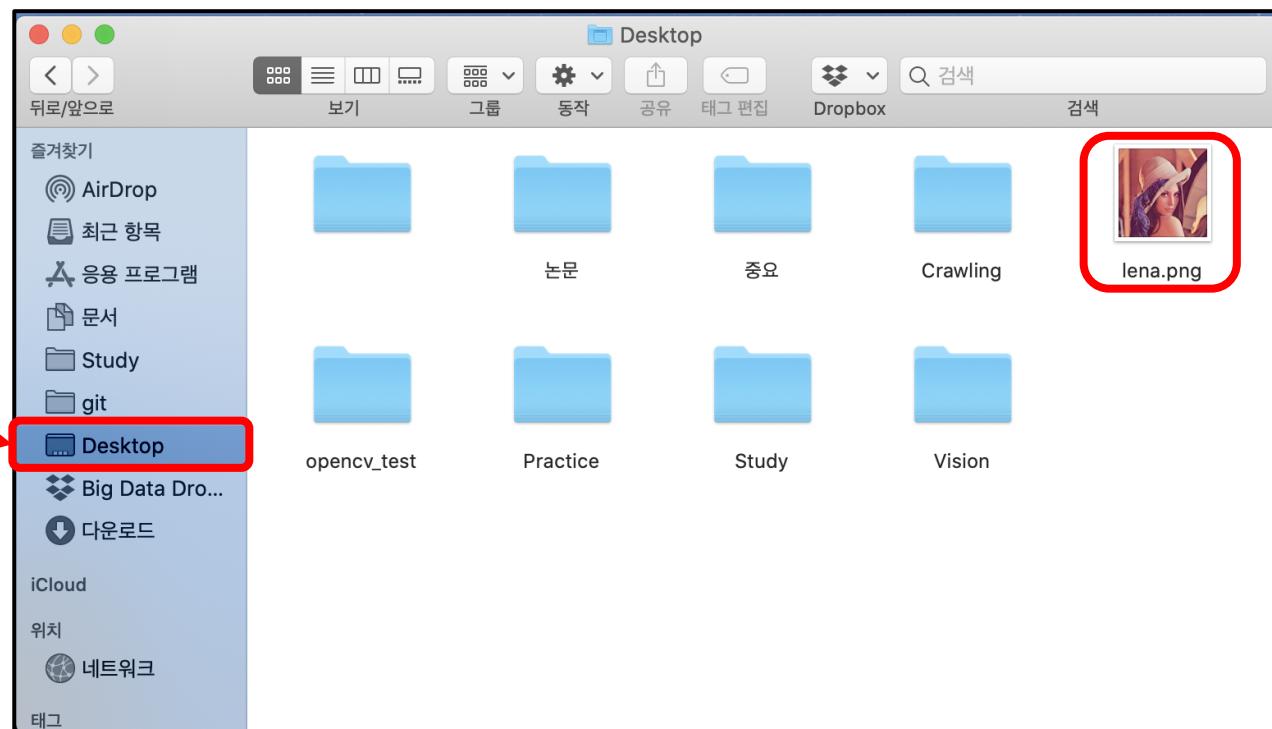
## 2) 파일의 절대 경로를 입력

```
#include <iostream>
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace cv;
using namespace std;

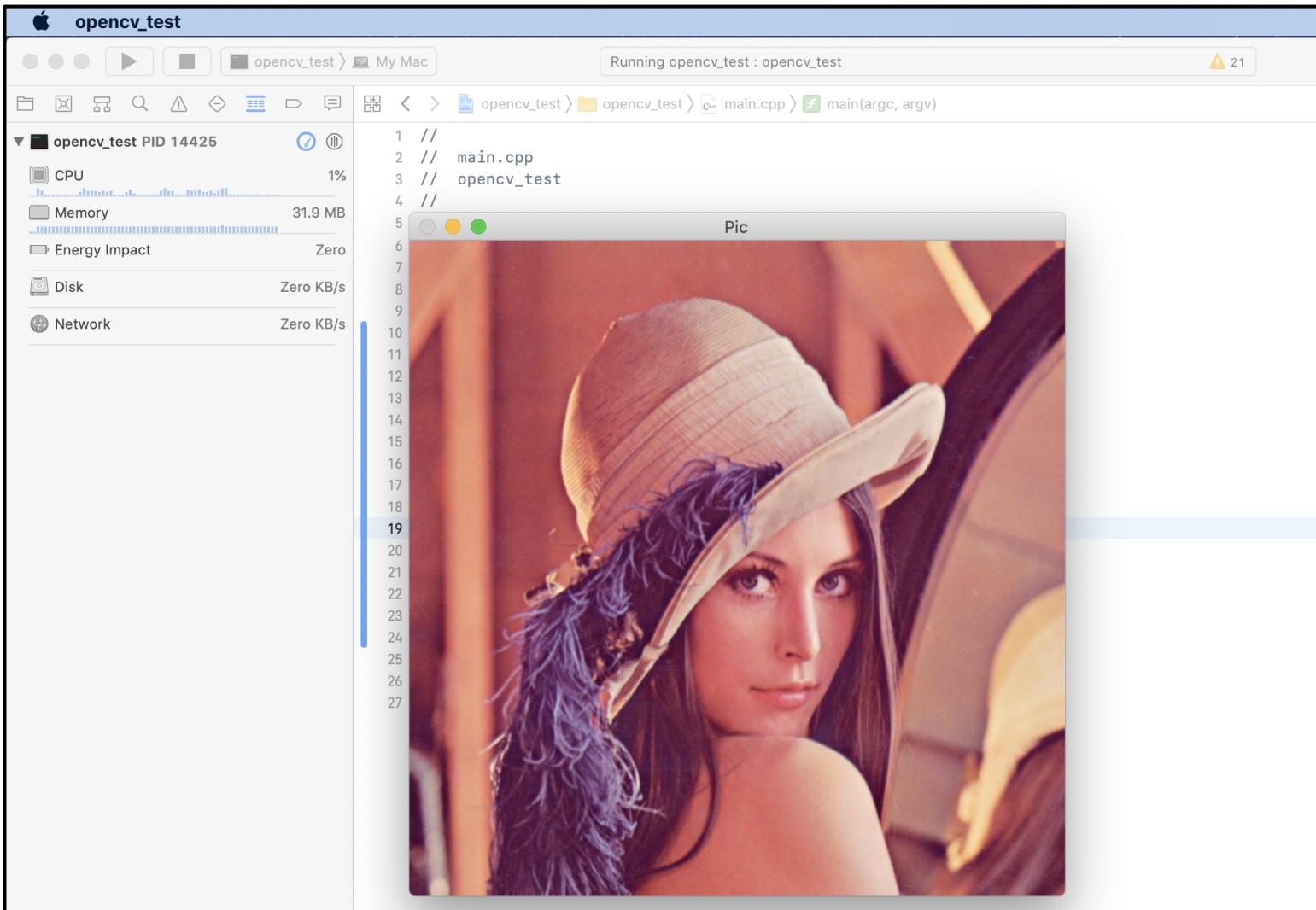
int main(int argc, const char * argv[]){
    Mat pic = imread("/Users/kimhyeyeong/Desktop/lena.png");
    if(pic.empty()){
        cerr << "read fail!" << endl;
        exit(-1);
    }
    imshow("Pic", pic);
    waitKey(0);

    return 0;
}
```



# 5. 예제 코드 실행 결과 확인

아래 화면과 같이, 이미지 파일이 정상적으로 출력되면 문제없이 설치가 완료된 것입니다.



Thank you!