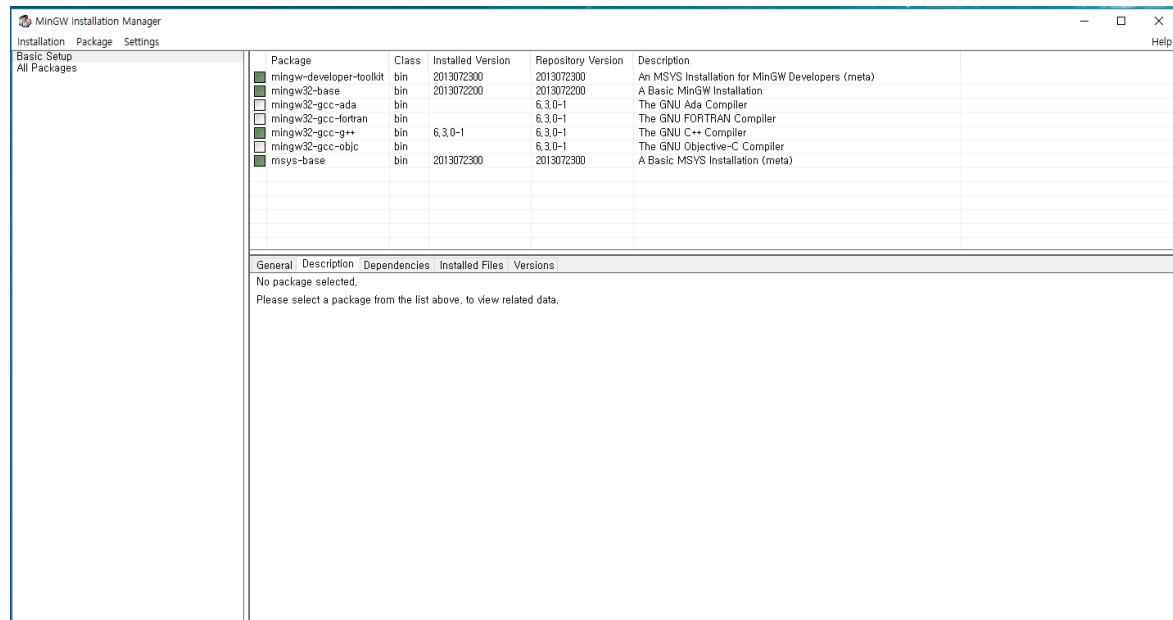


visual studio code 에서 c/c++ 개발환경 설정하기(windows 10)

1. VSCode 설치 (<https://code.visualstudio.com/download>)

(선택사항 : system installer)

2. MinGW 설치(<https://sourceforge.net/projects/mingw/>) - [MinGW.org GCC-6.3.0-1]



선택 후, 설치 (참고: <https://webnautes.tistory.com/1196>)

(선택사항: 옵션 ...for all users)

3. PATH에 경로 추가

- 하단 윈도우 검색창에서 '시스템 환경 변수 편집' 검색 후 실행
- 하단의 '환경 변수'를 선택하고,
- 시스템 환경변수의 'path'를 더블클릭'해서 '새로 만들기'를 선택
- 'C:\MinGW\bin'을 추가

4. cmd(명령 프롬프트)를 실행하여 PATH가 잘 추가 되었는지 확인.

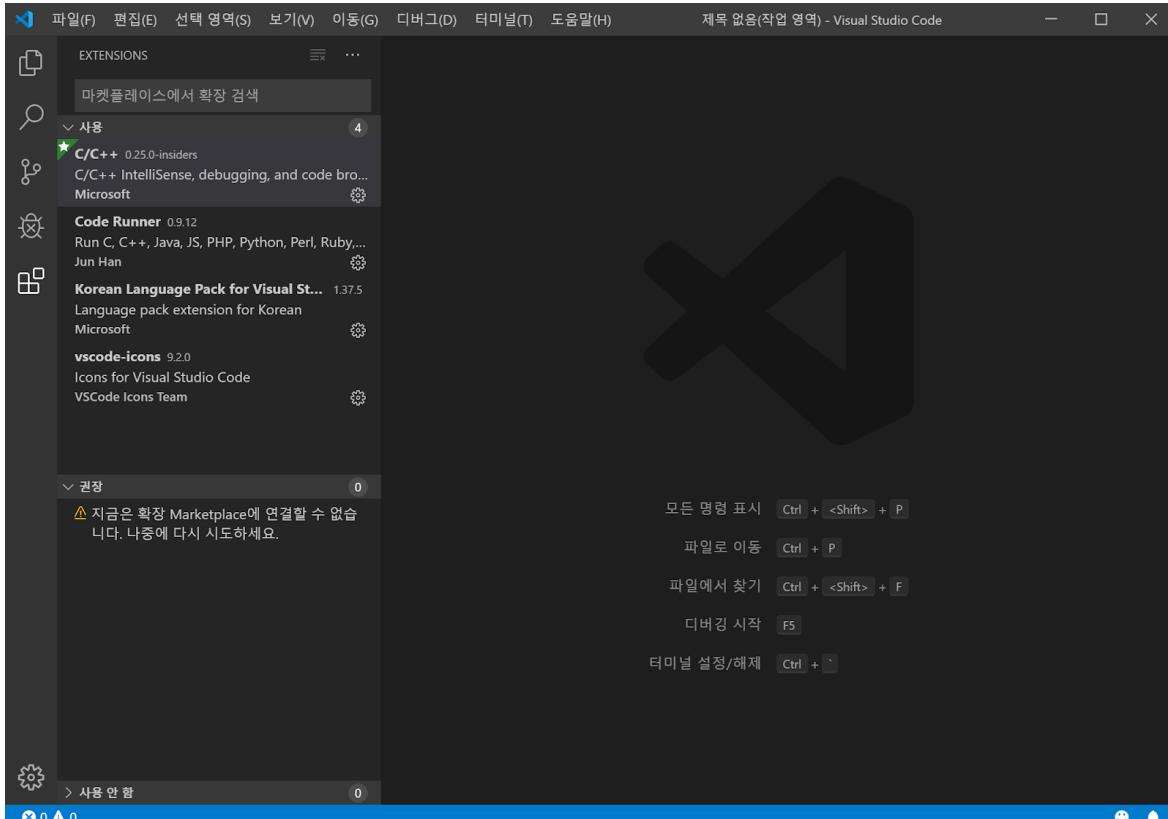
명령어 **gcc -v**
 g++ -v

5. 시스템 재시작(reboot)

6. VSCode를 실행

7. 확장 설치(Extensions, ctrl + shift + x), 왼쪽탭 제일 아래 버튼을 눌러서 검색 후 설치

- (선택) vscode-icons (VSCode Icons Team)
- (필수) C/C++ (Microsoft)
- (필수) Code Runner (Jun Han)
- (선택) Korean Language Pack for Visual Studio Code (Microsoft)



8. 탐색기(ctrl + shift + e), 폴더 열기를 눌러 폴더를 생성/선택

9. 간단한 소스파일(or 과제폴더 불러오기) 작성 후, 저장

1. main.c
2. suv.c
3. sub.h

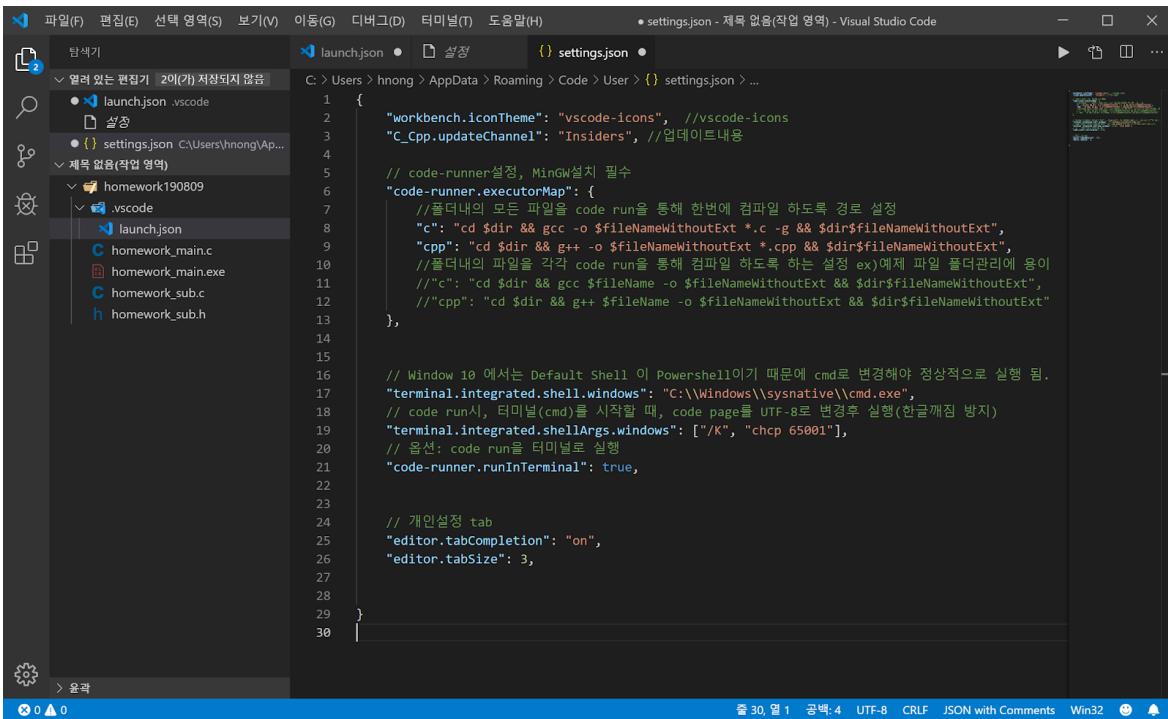
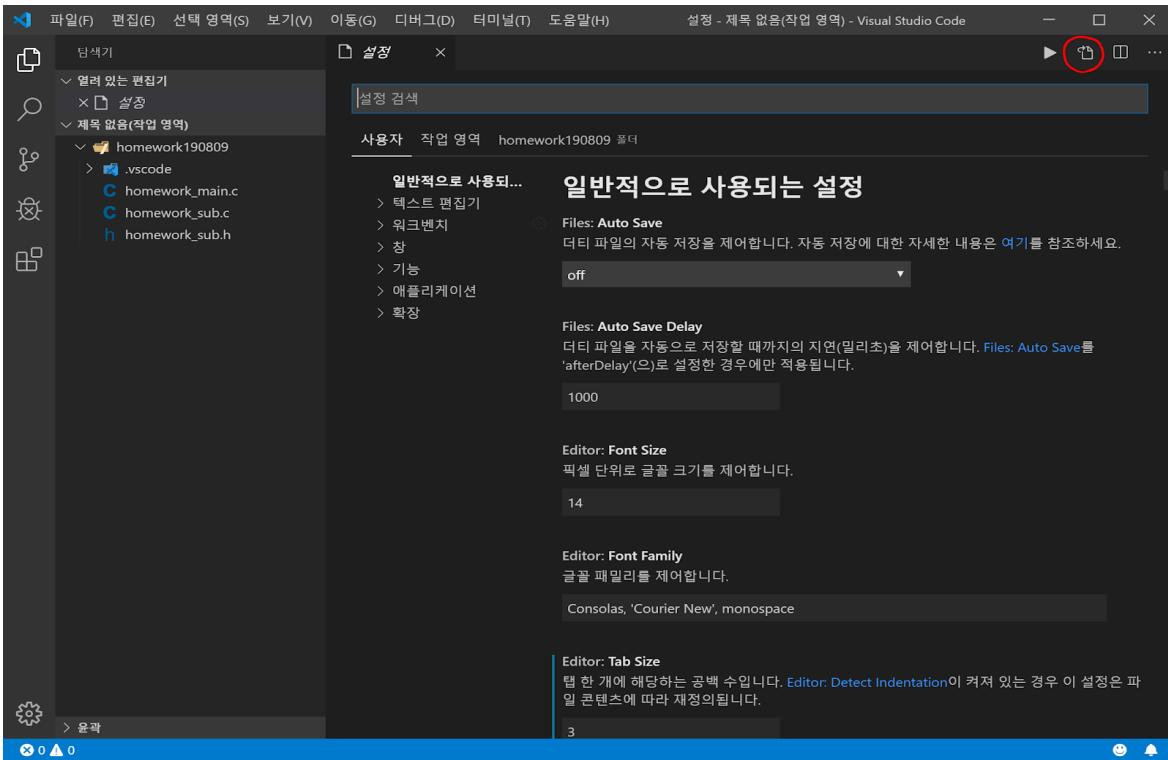
10. code runner를 활용한 code run

1. 'ctrl + ,를 눌러 설정창을 연 후에,
2. 우측 상단의 문서나 아이콘을 클릭하여 settings.json 파일을 열고,
3. 아래내용을 복사, 붙여넣기를 한 후에 '{ }' 내에 저장(ctrl + s)

```
// code-runner설정, MingW설치 필수
"code-runner.executorMap": {
    //폴더내의 모든 파일을 code run을 통해 한번에 컴파일 하도록 경로 설정
    "c": "cd $dir && gcc -o $fileNameWithoutExt *.c -g && $dir$fileNameWithoutExt",
    "cpp": "cd $dir && g++ -o $fileNameWithoutExt *.cpp && $dir$fileNameWithoutExt",
    //폴더내의 파일을 각각 code run을 통해 컴파일 하도록 하는 설정 ex)예제 파일 폴더관리에 용이
    //"c": "cd $dir && gcc $fileName -o $fileNameWithoutExt && $dir$fileNameWithoutExt",
    //"cpp": "cd $dir && g++ $fileName -o $fileNameWithoutExt && $dir$fileNameWithoutExt"
},
```

```
// Window 10에서는 Default Shell 이 Powershell이기 때문에 cmd로 변경해야 정상적으로 실행 됨.
"terminal.integrated.shell.windows": "C:\\Windows\\sysnative\\cmd.exe",
// code run시, 터미널(cmd)를 시작할 때, code page를 UTF-8로 변경후 실행(한글깨짐 방지)
"terminal.integrated.shellArgs.windows": ["/K", "chcp 65001"],
// 옵션: code run을 터미널로 실행
```

"code-runner.runInTerminal": true,



//주석처리 변경을 통해 작업 폴더마다 다르게 설정 가능

11. 작성해둔 소스파일을 우클릭해 code run을 실행한후 .exe 파일이 생성되었는지 확인

12. Debug setting(*주의)디버깅 실행시 콘솔의 한글이 깨져서 나타남/unicode 호환문제

1. F5를 누르거나 상단 메뉴의 디버그 > 구성추가 선택
2. C++ (GDB/LLDB) 선택

3. 아래내용을 복사, 붙여넣기를 한 후에 저장(crtl + s)code runner

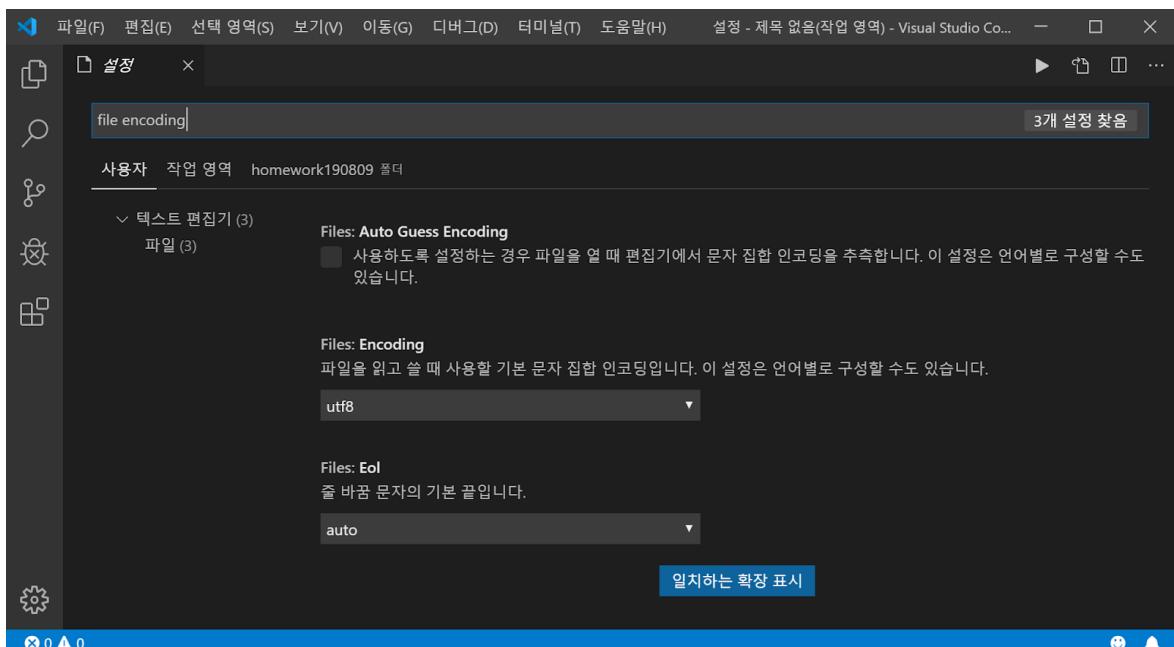
```
{  
    // IntelliSense를 사용하여 가능한 특성에 대해 알아보세요.  
    // 기존 특성에 대한 설명을 보려면 가리킵니다.  
    // 자세한 내용을 보려면 https://go.microsoft.com/fwlink/?linkid=830387을(를) 방문하세요.  
    "version": "0.2.0",  
    "configurations": [  
        {  
            "name": "(gdb) Launch",  
            "type": "cppdbg",  
            "request": "launch",  
            "program": "${fileDirname}/${fileBasenameNoExtension}.exe",  
  
            "args": [],  
            "stopAtEntry": false,  
            "cwd": "${workspaceFolder}",  
            "environment": [],  
            "externalConsole": true,  
            "MIMode": "gdb",  
            "miDebuggerPath": "C:/MinGW/bin/gdb.exe",  
            "setupCommands": [  
                {  
  
                    "description": "Enable pretty-printing for gdb",  
                    "text": "-enable-pretty-printing",  
                    "ignoreFailures": true  
                }  
            ]  
        }  
    ]  
}
```

```

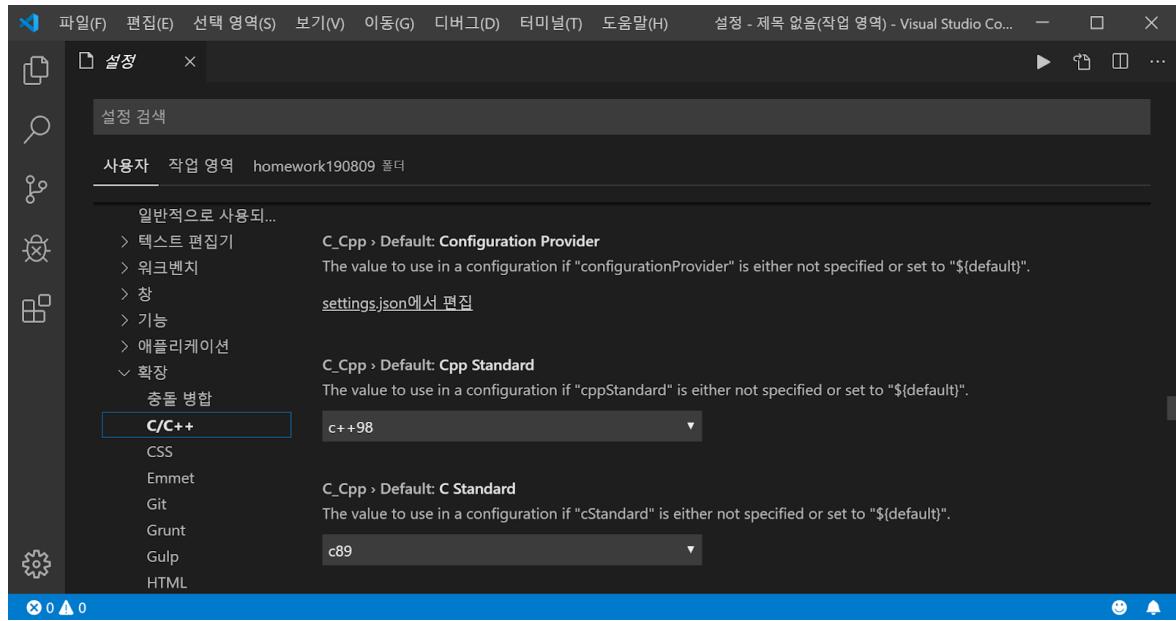
{
    "version": "0.2.0",
    "configurations": [
        {
            "name": "(gdb) Launch",
            "type": "cppdbg",
            "request": "launch",
            "program": "${fileDirname}/${fileBasenameNoExtension}.exe",
            "args": [],
            "stopAtEntry": false,
            "cwd": "${workspaceFolder}",
            "environment": [],
            "externalConsole": true,
            "MIMode": "gdb",
            "miDebuggerPath": "C:/MinGW/bin/gdb.exe",
            "setupCommands": [
                {
                    "description": "Enable pretty-printing for gdb",
                    "text": "-enable-pretty-printing",
                    "ignoreFailures": true
                }
            ]
        }
    ]
}

```

- vscode에서는 폴더별로 간편하게 프로젝트를 관리할 수 있다
- 셋팅은 .vscode 폴더를 복사해서 붙여넣거나 셋팅시 기존의 설정 내용을 복사/붙여넣기
- setting.json 파일은 이동금지, 필요, 용도에따라 주석처리 변경으로 변경해서 사용
- Debug 세팅(launch.json)은 하지 않아도 된다
- Debug 시, 콘솔에서 한글 깨짐을 방지하려면 코딩소스 파일의 인코딩을 변경한다./사진



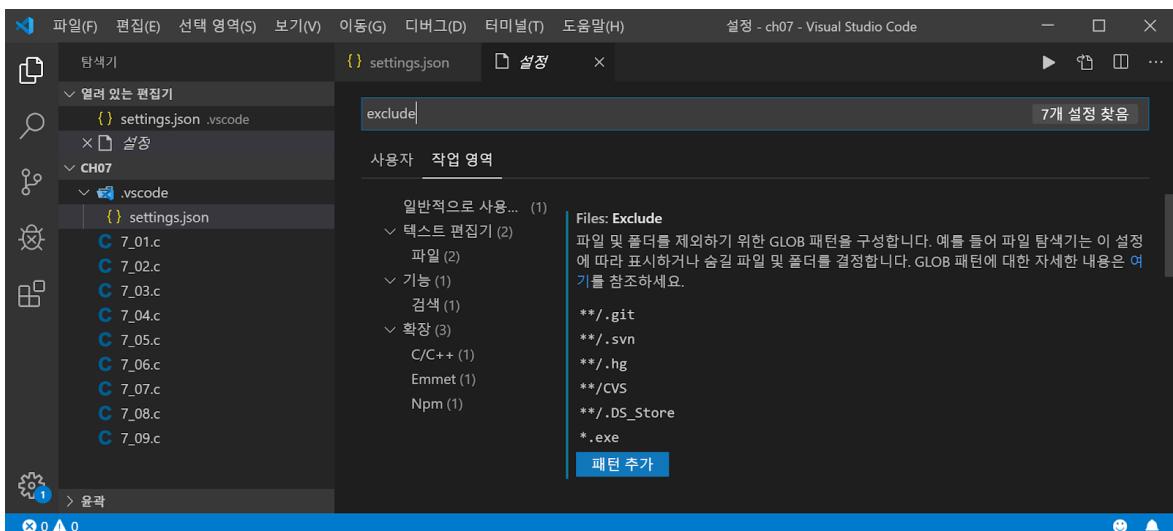
- C/C++의 표준 변경방법



**예제 연습시 실행파일이 보기 불편할 때, 작업영역 설정을 통해 전체 설정을 바꾸지 않고 폴더 내에서만 적용되도록 설정할 수 있다.



**설정([ctrl] + [,])의 작업 영역 탭에서 탐색기에서 관련 설정을 검색 후, 'exe'가 확장자인 파일을 보이지 않도록 설정하거나, settings.json을 작성



**vscode 폴더 내에 settings.json이 생성되었는지 확인

