

Visual Studio Code 如何编写运行 C、C++ 程序?

Serendipity 也关注了该问题

<https://www.zhihu.com/question/30315894/answer/3089595368>

已关注

编辑回答

邀请回答

好问题 317

14 条评论

分享 ...



东四王丁

1 人赞同了该回答

[Configure Visual Studio Code for Microsoft C++](#)[Get Started with C++ and Mingw-w64 in Visual Studio Code](#)[C++ programming with Visual Studio Code](#)

第一个[code.visualstudio.com/d...](#)内就是介绍了，先安装扩展，接着就可以安装mingw

1、扩展

[C/C++ extension for VS Code](#)

2、MSYS2

MSYS2安装好MSYS2以后，就可以使用命令行安装mingw

3、安装mingw-w⁶⁴

安装: `pacman -S --needed base-devel mingw-w64-x86_64-toolchain`

选择这直接默认就好，也就是直接按Enter回车

```
10696@DESKTOP-9L473HP UCRT64 ~
$ pacman -S --needed base-devel mingw-w64-x86_64-toolchain
:: There are 19 members in group mingw-w64-x86_64-toolchain:
:: Repository mingw64
 1) mingw-w64-x86_64-binutils 2) mingw-w64-x86_64-crt-git 3) mingw-w64-x86_64-gcc
 4) mingw-w64-x86_64-gcc-ada 5) mingw-w64-x86_64-gcc-fortran
 6) mingw-w64-x86_64-gcc-libgfortran 7) mingw-w64-x86_64-gcc-libs 8) mingw-w64-x86_64-gcc-objc
 9) mingw-w64-x86_64-gdb 10) mingw-w64-x86_64-gdb-multiarch 11) mingw-w64-x86_64-headers-git
12) mingw-w64-x86_64-libgccjit 13) mingw-w64-x86_64-libmangle-git
14) mingw-w64-x86_64-libwinpthread-git 15) mingw-w64-x86_64-make 16) mingw-w64-x86_64-pkgconf
17) mingw-w64-x86_64-tools-git 18) mingw-w64-x86_64-winpthreads-git
19) mingw-w64-x86_64-winstorecompat-git

Enter a selection (default=all):
resolving dependencies...
looking for conflicting packages...
```

知乎 @东四王丁

```
10696@DESKTOP-9L473HP UCRT64 ~
$ pacman -S --needed base-devel mingw-w64-x86_64-toolchain
:: There are 19 members in group mingw-w64-x86_64-toolchain:
:: Repository mingw64
 1) mingw-w64-x86_64-binutils 2) mingw-w64-x86_64-crt-git 3) mingw-w64-x86_64-gcc
 4) mingw-w64-x86_64-gcc-ada 5) mingw-w64-x86_64-gcc-fortran
 6) mingw-w64-x86_64-gcc-libgfortran 7) mingw-w64-x86_64-gcc-libs 8) mingw-w64-x86_64-gcc-objc
 9) mingw-w64-x86_64-gdb 10) mingw-w64-x86_64-gdb-multiarch 11) mingw-w64-x86_64-headers-git
12) mingw-w64-x86_64-libgccjit 13) mingw-w64-x86_64-libmangle-git
14) mingw-w64-x86_64-libwinpthread-git 15) mingw-w64-x86_64-make 16) mingw-w64-x86_64-pkgconf
17) mingw-w64-x86_64-tools-git 18) mingw-w64-x86_64-winpthreads-git
19) mingw-w64-x86_64-winstorecompat-git

Enter a selection (default=all):
resolving dependencies...
looking for conflicting packages...
.....
```

赞同 1

添加评论

分享

收藏

设置

收起 ^

win10的配置页面即可配置

Your PC is being monitored and protected.

This page has a few new settings

Some settings from Control Panel have moved here, and you can now share your PC info so it's easier to share

Related settings

BitLocker settings

Copy

Rename this PC

Device Manager

Edition	Windows 10 Home
Version	22H2

Remote desktop

System protection

Advanced system settings

Rename this PC (advanced)

 [Get help](#) [Give feedback](#)

知乎 @东四王丁

配置系统环境变量，双击PATH即可的，最下面两行

知乎 @东四王丁

5、验证

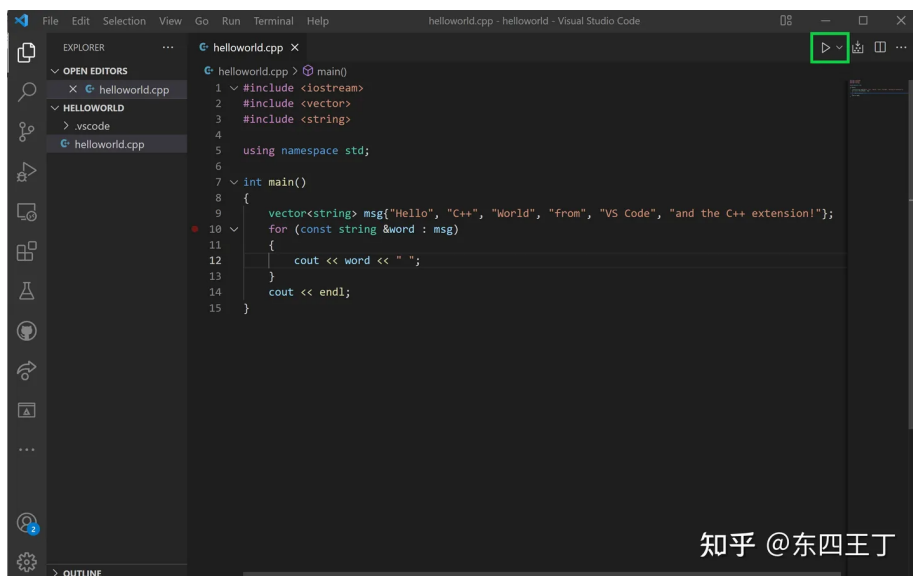
Windows+R输入cmd运行，然后gcc --version和g++ --version就可以验证的

```
C:\Users\10696>gcc --version
gcc (Rev6, Built by MSYS2 project) 13.1.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

C:\Users\10696>g++ --version
g++ (Rev6, Built by MSYS2 project) 13.1.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

知乎@东四王丁

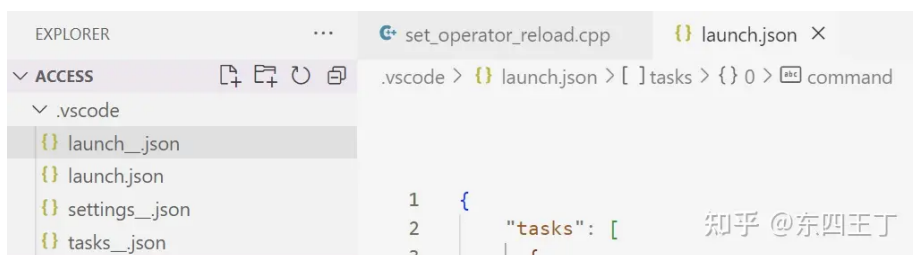
6、写codes



知乎 @东四王丁

7、产生相应的配置档案

.vscode/tasks.json^Q，编译运行使用的，两个路径的呢cwd和command



知乎 @东四王丁

```
{
  "tasks": [
    {
      "type": "cppbuild",
      "label": "C/C++: g++.exe build active file",
      "command": "F:\\kksoftware\\MSYS2\\mingw64\\bin\\g++.exe",
      "args": [
        "-fdiagnostics-color=always",
        "-g",
        "${file}",
        "-o",
        "${fileDirname}\\${fileBasenameNoExtension}.exe"
      ],
      "options": {
```

```

        "problemMatcher": ["$gcc"],
        "group": {
            "kind": "build",
            "isDefault": true
        },
        "detail": "Task generated by Debugger."
    },
    ],
    "version": "2.0.0"
}

```

.vscode/launch.json, 两个路径的呢cud和miDebuggerPath

```

{
    "version": "0.2.0",
    "configurations": [
        {
            "name": "C/C++: g++.exe build and debug active file",
            "type": "cppdbg",
            "request": "launch",
            "program": "${fileDirname}\\${fileBasenameNoExtension}.exe",
            "args": [],
            "stopAtEntry": false,
            "cwd": "F:\\kksoftware\\MSYS2\\mingw64\\bin\\",
            "environment": [],
            "externalConsole": false,
            "MIMode": "gdb",
            "miDebuggerPath": "F:\\kksoftware\\MSYS2\\mingw64\\bin\\gdb.exe",
            "setupCommands": [
                {
                    "description": "Enable pretty-printing for gdb",
                    "text": "-enable-pretty-printing",
                    "ignoreFailures": true
                }
            ],
            "preLaunchTask": "C/C++: g++.exe build active file"
        }
    ]
}

```

8、编译

编译的时候要选择相应的配置档案编译和debug

9、编译和debug的路径

编译: vscode上方: Terminal/Run Build Task

编译+运行: vscode上方: Run/Run without debugging

编译+debug: vscode上方: Run/Start debugging

编辑于 2023-06-25 20:00 · IP 属地上海

 修改

