

Connecting to Grace using VS Code

(by Henry Tran)

Video Instruction: <https://youtu.be/hUJHwmtY020>

Reference:

Peter Currie's instructions: [Youtube Video](#) and [Google Docs](#)

We greatly appreciate the work done by Peter Currie (CS Student), who created the original video and documentation illustrating how to connect to grace using VS Code correctly. It has made a big difference as students love VS Code and encourage the use of VS Code in classes like CMSC216. Peter, thank you :)

Step 1: Download VS Code

<https://code.visualstudio.com/Download>

Step 2: Install VS Code Extensions

C/C+ by Microsoft:

<https://marketplace.visualstudio.com/items?itemName=ms-vscode.cpptools>

SFTP by Naticzskunk:

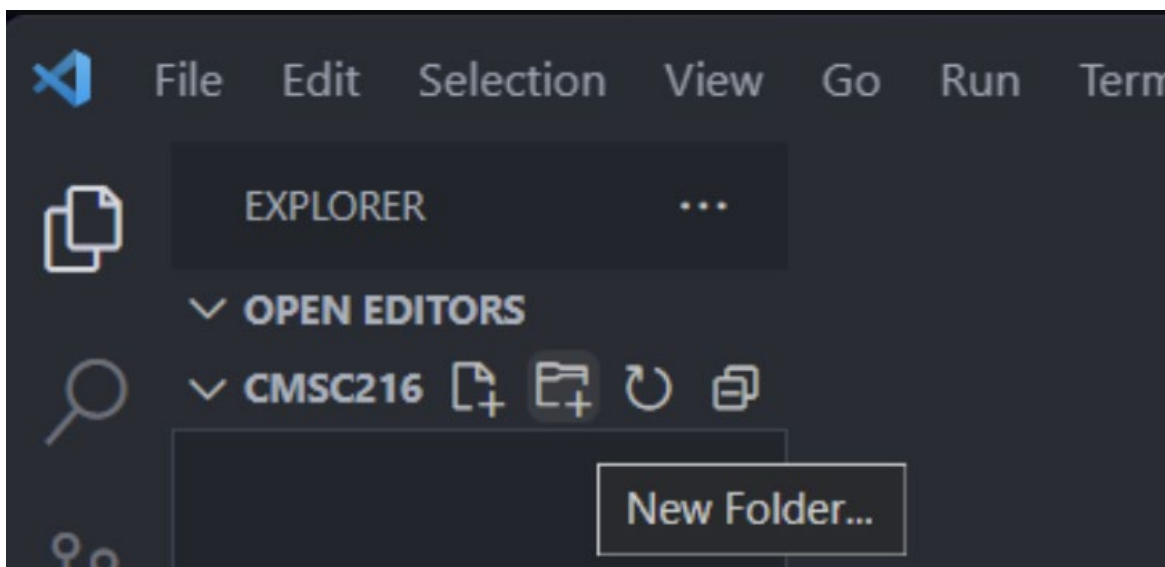
<https://marketplace.visualstudio.com/items?itemName=Naticzskunk.sftp>

Step 3: Set up The Workplace

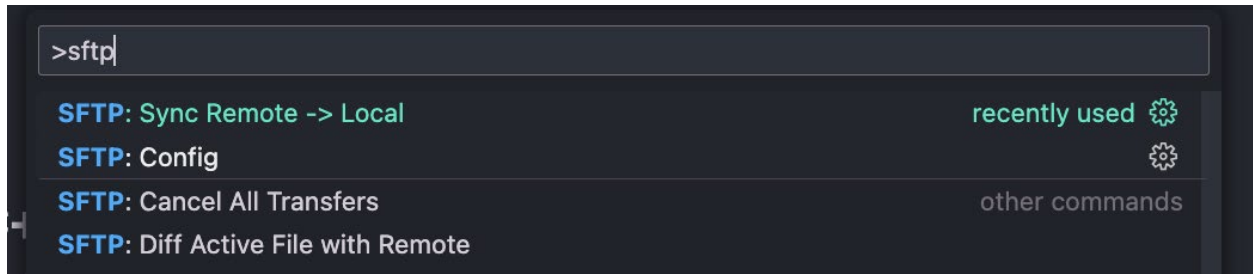
3.1: Create a CMSC216 folder on your computer

3.2: Select **File -> Open Folder** or press **Ctrl + O** (MacOS) **Ctrl + K Ctrl + O** (Windows), then select your CMSC216 folder

3.3: Create 2 new folders "216" and "216public" (Make sure the folders are not inside each other)



3.4: Select **View -> Command Palette** or press **Command+Shift+P** (MacOS) **Ctrl+Shift+P** (Windows), then type “sftp” and click on “**SFTP: Config**”

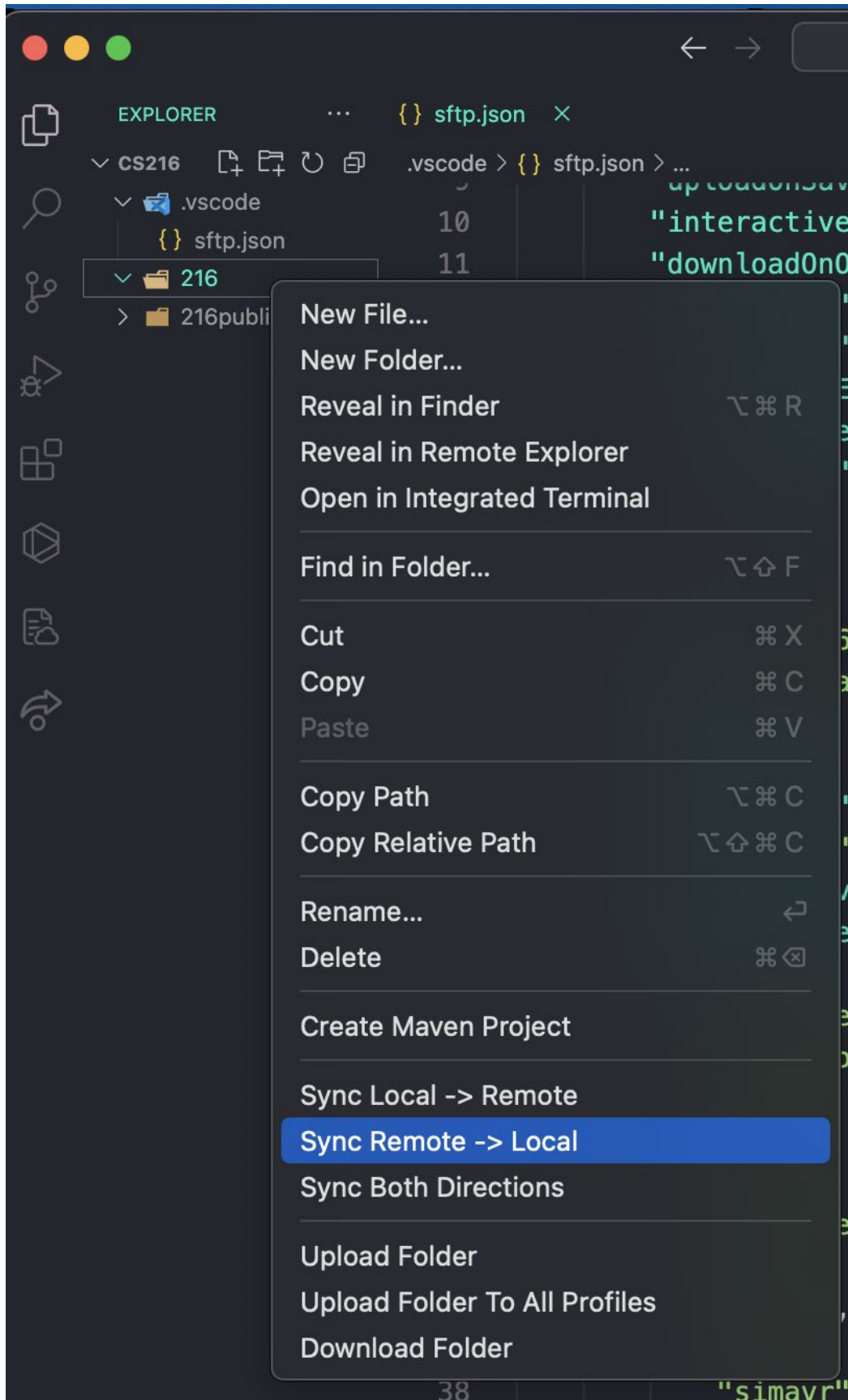


3.5: Delete everything inside **sftp.json** file, and copy/paste the following code

(USERNAME is your directory ID (e.g., ktran58), and your section should replace 020X)

```
[
  {
    "name": "216",
    "host": "grace.umd.edu",
    "protocol": "sftp",
    "username": "USERNAME",
    "remotePath": "/afs/glue.umd.edu/class/spring2024/cmsc/216/0201/student-cmsc216-
020X/USERNAME",
    "context": "216/",
    "uploadOnSave": true,
    "interactiveAuth": true,
    "downloadOnOpen": false,
    "syncOption": {
      "delete": true,
      "ignoreExisting": false,
      "skipCreate": false,
      "update": false
    }
  },
  {
    "name": "216public",
    "host": "grace.umd.edu",
    "protocol": "sftp",
    "username": "USERNAME",
    "remotePath": "/afs/glue.umd.edu/class/spring2024/cmsc/216/0201/public",
    "context": "216public/",
    "uploadOnSave": false,
    "interactiveAuth": true,
    "ignore": [
      ".htaccess",
      "arduino*",
      "avr*",
      "gdb*",
      "bin",
      "include",
      "lib*",
      "share",
      "old",
      "simavr"
    ]
  }
]
```

3.6: Right-click on 216 folder and select “Sync Remote -> Local”



3.7: Enter your directory ID password, then select an option for Duo 2FA

IMPORTANT: If you have done the previous steps but your 216 folder has not been updated, log into the Grace server then run the following commands

```
cd ~/216
```

```
realpath .
```

The result should match the remotePath

"/afs/glue.umd.edu/class/spring2024/cmssc/216/0201/student-cmssc216-020X/USERNAME", if not, copy that result and change the remotePath for 216

Note: **Sync Remote -> Local** will download the files from 216 on Grace and place them into 216 on your local computer

Sync Local -> Remote will upload your files from your local computer 216 folder to 216 on Grace

You can **Sync Remote -> Local** from 216public to your local 216public, feel free to make changes or run code in 216public on your computer, but you don't have permission to do so on Grace.

Step 4: Testing

4.1: Open Terminal on VSCode by selecting **View -> Terminal** or pressing **control + `** (MacOS) **Ctrl + `** (Windows)

4.2: Log into Grace using the command **ssh -X yourDirectoryID@grace.umd.edu**

4.3: Create a new file on your local 216, then **Sync Local -> Remote**, and check if the file is in 216 on the Grace server.

*Note: Since we set "uploadOnSave": true, when you save any changes (**Ctrl + S**), it will automatically sync from local to remote.*

4.4: Create a new file on Grace, then **Sync Remote -> Local**, and check if the file is in 216 on your local computer.

Note: At this point, you should be able to use VS Code as a text editor and compile your code on the Grace server. If you want to run C code on your computer, follow the optional steps. From my experience, I'd recommend using the Grace environment to test your code since it supports better and you don't have to install anything else.

(OPTIONAL)

Step 5 (Windows users only): Install Windows Subsystem for Linux

<https://learn.microsoft.com/en-us/windows/wsl/install>

Step 6: Install gcc and gdb on your computer

For Windows: Run the following commands one by one

```
sudo apt update
```

```
sudo apt upgrade
```

```
sudo apt install gcc gdb
```

For MacOS: First, install Homebrew: <https://brew.sh/>. Then run the following command

```
brew install gcc gdb
```

Some other useful features on VSCode:

<https://www.wearedevelopers.com/magazine/vs-code-productivity#:~:text=Instead%2C%20VS%20Code%20will%20automatically,Saves%20file%20after%20every%20second.>