GIT: Advanced Commands

GIT Default MergeTool Activity: Setting our default Merge Tool to VSCode

Brian Gorman, Author/Instructor/Trainer

Introduction

Although BASH allows us edit files via a call to VIM as the default merge tool for resolving conflicts, using it is a very tricky operation at best. I remember the first time VIM opened up for a message and all I could think was “How the heck do I get out of this thing!”

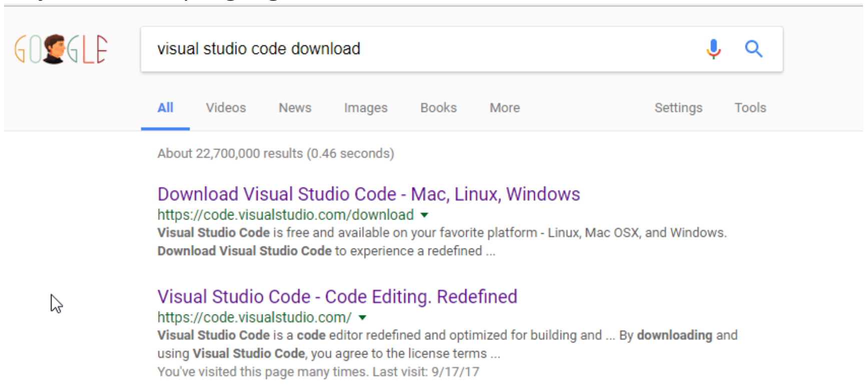
VSCode gives us an incredibly powerful mergetool that easily lets us select the source, target, or both changes, and also easily edit the changes during the merge operation. We definitely want a nice tool like this in place to make our lives easier.

Let’s gets started!

Step 1: Get Visual Studio Code Setup [if you don't already have it]

a) Download and install Visual Studio Code onto our machine.
Go To: https://code.visualstudio.com/download

Or just do a simple google search for Visual Studio Code:


visual studio code download

All Videos News Images Books More Settings Tools

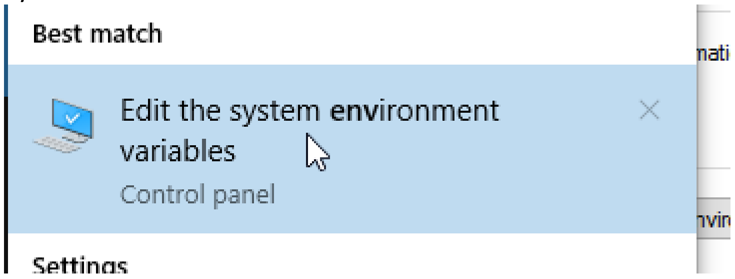
About 22,700,000 results (0.46 seconds)

Download Visual Studio Code - Mac, Linux, Windows
https://code.visualstudio.com/download
Visual Studio Code is free and available on your favorite platform - Linux, Mac OSX, and Windows. **Download Visual Studio Code** to experience a redefined ...

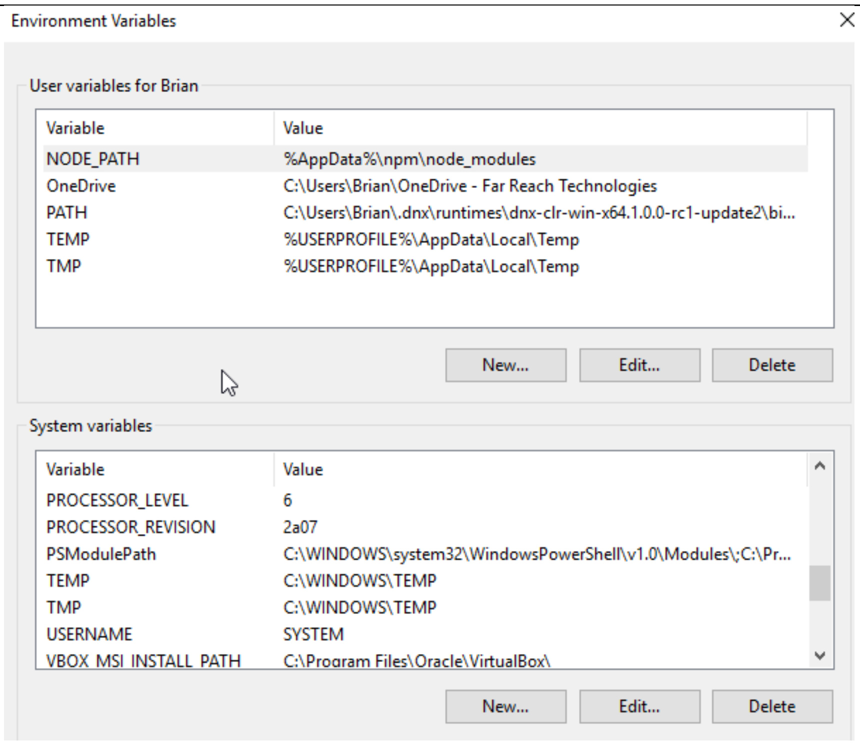
Visual Studio Code - Code Editing. Redefined
https://code.visualstudio.com/
Visual Studio Code is a **code** editor redefined and optimized for building and ... By **downloading** and using **Visual Studio Code**, you agree to the license terms ...
You've visited this page many times. Last visit: 9/17/17

Install the application, start it up and make sure it works.
More information about the application can be found here:
https://code.visualstudio.com/docs

b) If on a windows machine, make sure code is in your PATH variable.
We don't have to do this, but it will make things a lot easier. We'll be able to reference the executable directly, rather than having to code the entire path to the executable.
Go to the start menu and type "Environment Variables" Then select "Edit System Environment variables"



Notes



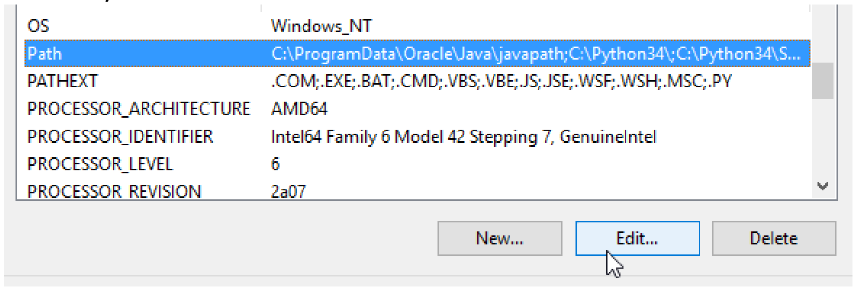
User variables for Brian

Variable	Value
NODE_PATH	%AppData%\npm\node_modules
OneDrive	C:\Users\Brian\OneDrive - Far Reach Technologies
PATH	C:\Users\Brian\.dnx\runtimes\dnx-clr-win-x64.1.0.0-rc1-update2\bi...
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

System variables

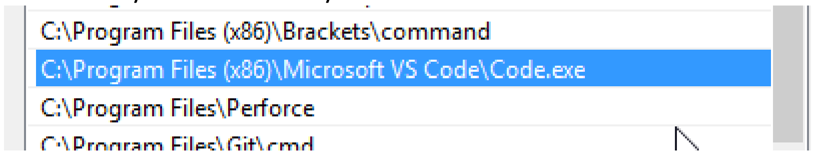
Variable	Value
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	2a07
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
TEMP	C:\WINDOWS\TEMP
TMP	C:\WINDOWS\TEMP
USERNAME	SYSTEM
VBOX_MSI_INSTALL_PATH	C:\Program Files\Oracle\VirtualBox\

Under 'System Variables' find "PATH"



OS	Windows_NT
Path	C:\ProgramData\Oracle\Java\javapath;C:\Python34\;C:\Python34\S...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC;.PY
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 42 Stepping 7, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	2a07

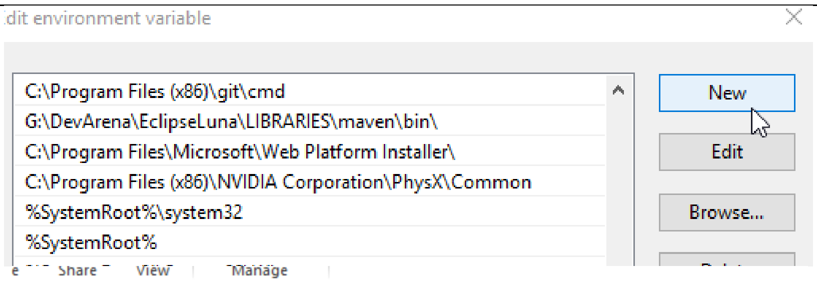
Select Edit.
If you are on an older version of windows, you may need to parse the string in a text editor. It's much easier now in Windows 10:
Make sure you have an entry for CODE:



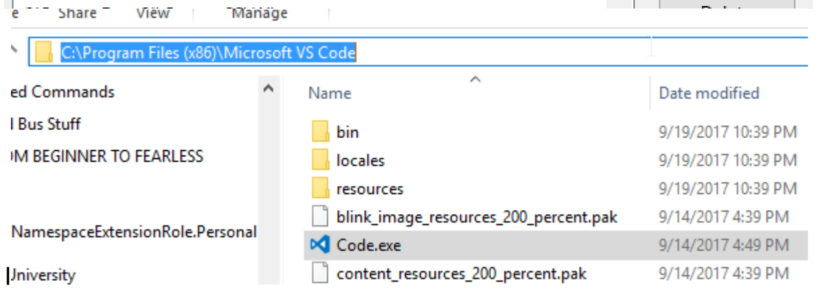
C:\Program Files (x86)\Brackets\command
C:\Program Files (x86)\Microsoft VS Code\Code.exe
C:\Program Files\Perforce
C:\Program Files\Git\cmd

If you do not, then you will want to add one using the "NEW…" button:

First find the path to your executable:



```
C:\Program Files (x86)\git\cmd
G:\DevArena\EclipseLuna\LIBRARIES\maven\bin\
C:\Program Files\Microsoft\Web Platform Installer\
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
%SystemRoot%\system32
%SystemRoot%
```



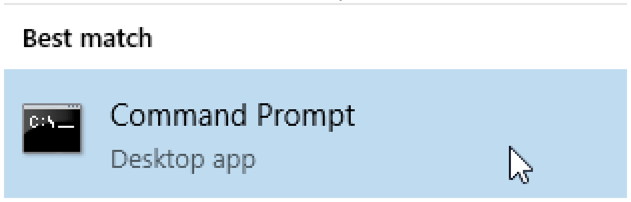
```
C:\Program Files (x86)\Microsoft VS Code
```

Name	Date modified
bin	9/19/2017 10:39 PM
locales	9/19/2017 10:39 PM
resources	9/19/2017 10:39 PM
blink_image_resources_200_percent.pak	9/14/2017 4:39 PM
Code.exe	9/14/2017 4:49 PM
content_resources_200_percent.pak	9/14/2017 4:39 PM

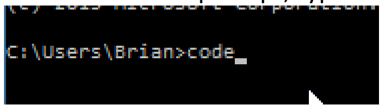
Then place that path into your Environment variables.

c) Test that it works:

Go to the start menu and type 'cmd'
Then select "Command Prompt"



In the command prompt, type "code":



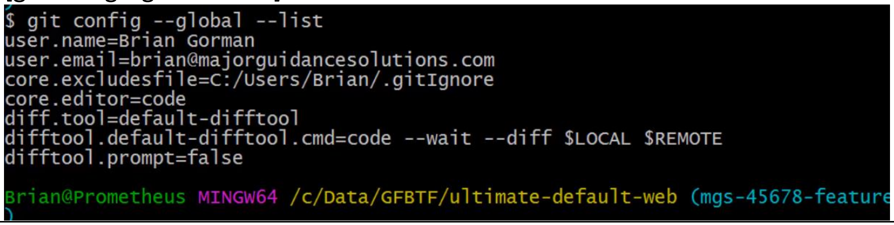
```
C:\Users\Brian>code_
```

If VS Code launches, you are all set. If not, you can either try again or just use the full path from above when setting values in the git config for difftool.

Step 2: Go to any repository

a) Make sure you are on any working repository. It doesn't even have to be up to date. We are not going to be affecting anything.

b) Take a quick look at config file settings:
[git config --global --list]



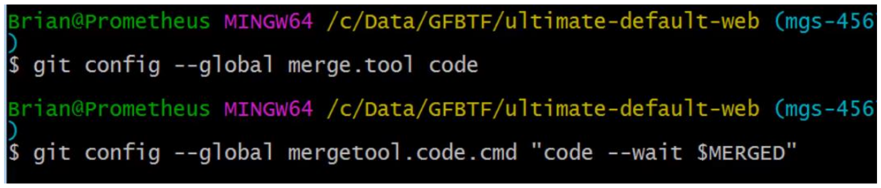
```
$ git config --global --list
user.name=Brian Gorman
user.email=brian@majorguidancesolutions.com
core.excludesfile=C:/Users/Brian/.gitIgnore
core.editor=code
diff.tool=default-difftool
difftool.default-difftool.cmd=code --wait --diff $LOCAL $REMOTE
difftool.prompt=false

Brian@Prometheus MINGW64 /c/Data/GFBTF/ultimate-default-web (mgs-45678-feature
```

No settings are in place for the merge tool at this point.

c) Add the entry for the merge tool

[git config --global merge.tool code]
[git config –global mergetool.code.cmd "code --wait $MERGED"]

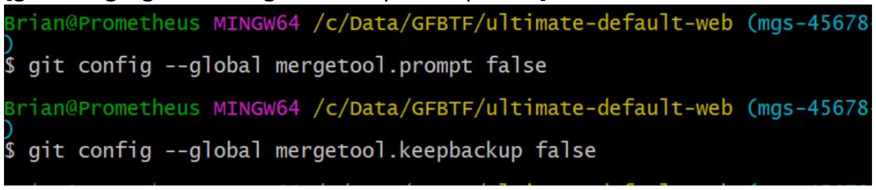


```
Brian@Prometheus MINGW64 /c/Data/GFBTF/ultimate-default-web (mgs-456
)
$ git config --global merge.tool code

Brian@Prometheus MINGW64 /c/Data/GFBTF/ultimate-default-web (mgs-456
)
$ git config --global mergetool.code.cmd "code --wait $MERGED"
```

d) Add settings to avoid prompting and keeping backups

[git config –global mergetool.prompt false]
[git config --global mergetool.keepbackup false]

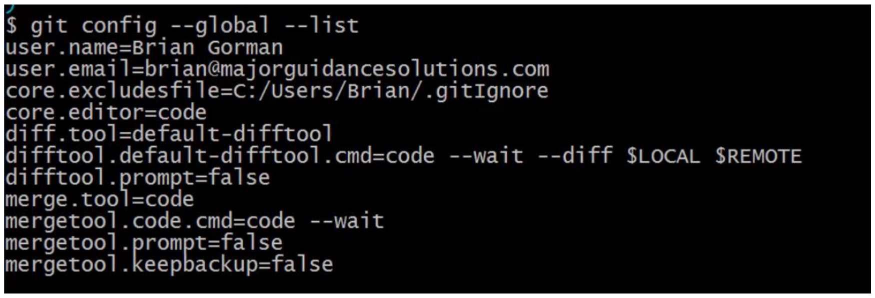


```
Brian@Prometheus MINGW64 /c/Data/GFBTF/ultimate-default-web (mgs-45678
)
$ git config --global mergetool.prompt false

Brian@Prometheus MINGW64 /c/Data/GFBTF/ultimate-default-web (mgs-45678
)
$ git config --global mergetool.keepbackup false
```

e) Review the config settings:

[git config --global --list]

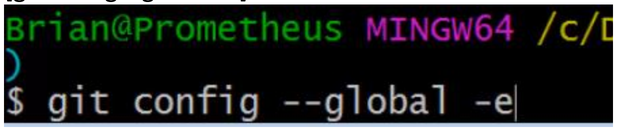


```
$ git config --global --list
user.name=Brian Gorman
user.email=brian@majorguidancesolutions.com
core.excludesfile=C:/Users/Brian/.gitIgnore
core.editor=code
diff.tool=default-difftool
difftool.default-difftool.cmd=code --wait --diff $LOCAL $REMOTE
difftool.prompt=false
merge.tool=code
mergetool.code.cmd=code --wait
mergetool.prompt=false
mergetool.keepbackup=false
```

--looks like the code command didn't work as expected

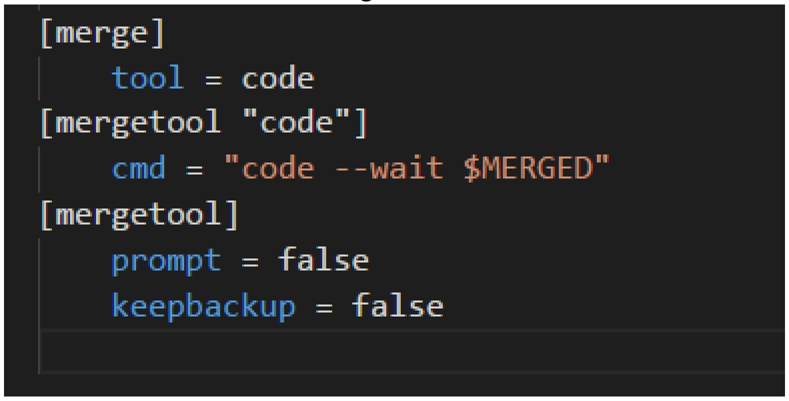
f) Enter the editor and setup the command for mergetool as expected manually

[git config --global -e]



```
Brian@Prometheus MINGW64 /c/D
)
$ git config --global -e
```

Enter “$MERGED” into the mergetool.code.cmd line

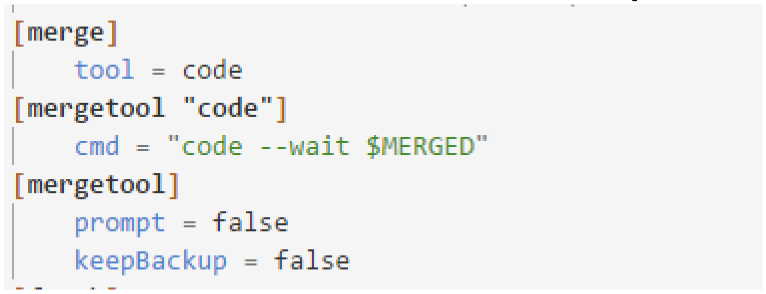


```
[merge]
    tool = code
[mergetool "code"]
    cmd = "code --wait $MERGED"
[mergetool]
    prompt = false
    keepbackup = false
```

Step 3: Verify mergetool is setup

a) To see that the mergetool is working, we’ll need a conflict to resolve. When we get to that, if something doesn’t work, come back to this activity and make sure that everything is setup as expected. For now, let’s just take a quick look:
[git config --global -e]

Make sure have the entries as above or as shown here [same values]



```
[merge]
    tool = code
[mergetool "code"]
    cmd = "code --wait $MERGED"
[mergetool]
    prompt = false
    keepBackup = false
```

This concludes our GIT Default MergeTool Activity.

Closing Thoughts

Setting VSCode [or another tool] to use for the default merge tool gives us a nice way to start working with a more user-friendly option [rather than VIM]. Again, VIM is perfectly fine if you like that tool, and so using a tool like VSCode is entirely optional, but highly recommended, especially for merge operations. Take a few minutes to make some notes about the various commands we've learned about in this activity, and practice using them.	Notes