Manual for Shortcuts

Kang SK

October 28, 2020

Contents

Introduction

Shortcuts are one of the best friends for a programmer, a writer, or basically anyone who works with a computer. However, multiple environments have their own unique keybindings, making users hard to distinguish between them. Of course, a user may fix his/her working environment, especially the working OS, to further fullfil the needs. However, this may not be possible, or even if possible, a very tiresome job for some people, including myself.

Before going any further, I admit I used the Windows OS for quite a long time, and I still need to go back and forth from Linux to Windows. Additionally, I admit, therefore I am more comfortable with the shortcuts on the Windows OS. Although yes, I can get used to multiple sets of shortcuts, I don't really think this is necessary, if I can simply modify the settings instead. This was the motivation for making this repository in the first place, and here it is, the manual for it.

Some might argue that this is a "sacrilegious" act, ignoring the design choices of bindings. Some might argue that I am limiting myself from being able to use any other computer environment than my own. I agree with some parts of the idea, but frankly saying, *I don't care what others say.*

Therefore, I state here the objective of this manual. First of all, this manual and all the shortcuts in it is based on **my** perspective, and is aiming for **my** convenience. Modifying shortcuts of the Windows OS is much harder compared to changing settings on a Linux environment, therefore this manual is closer to the Windows' shortcuts. This will result a Debian environment as convenient as possible for pre-Window users.

1 Debian Shortcuts

2 Terminal Shortcuts

3 VS Code Shortcuts

The basic philosophy of the shortcut design is as follows:

- Ctrl is the main function key.
- Up to two Ctrl keybindings make up functionality; the first key stands for the group, the second for the actual function.
- Ctrl + W manages functions related to windows.
- Ctrl + G manages functions related to git.
- Ctrl + B manages functions related to bookmarks.
- Alt is for alternation, which is used to alternate text.
- Ctrl + special keys are gotos.

Toggle Fullscreen	Ctrl + W, F
Toggle Sidebar	Ctrl + W, B
Split Workspace	Ctrl + W, V
Spilt Terminal	Ctrl + W, V
Close Window	Ctrl + W
Switch to Header(C/C++)	Ctrl + W, H
Diff Active and Saved	Ctrl + W, D
Fold All (Shrink)	Ctrl + W, S
Expand All	Ctrl + W, E

Figure 1: Keybindings for Windows

Git Fetch	Ctrl + G, F
Git Pull	Ctrl + G, Ctrl + P
Git Commit	Ctrl + G, C
Git Push	Ctrl + G, P
Git Sync	Ctrl + G, S
Git Merge	Ctrl + G, M
Git Undo Commit	Ctrl + G, U
Git Create Tag	Ctrl + G, T
Git Delete Tag	Ctrl + G, Ctrl + T

Figure 2: Keybindings for Git

Toggle Bookmark	Ctrl + B, B
Goto Prev Bookmark	Ctrl + B, H
Goto Next Bookmark	Ctrl + B, L

Figure 3: Keybindings for Bookmarks

Delete Line	Alt + X
Comment Line	Alt + C
Indent Line	Alt +]
Auto Fix	Alt + F
Next Match (Multi)	Alt + D
Trigger Suggest	Alt + I

Figure 4: Keybindings for Alternation

Goto Bracket	Ctrl +]
Goto Last Edit	Ctrl + E
Goto Definition	Ctrl + D
Goto Implementation	Ctrl + I

Figure 5: Keybindings for Gotos