

Visual Studio Code Setup

This is an extended guide on how to setup Visual Studio Code to use Intellisense for HPL3 and SOMA. Visual Studio Code is a lightweight notepad-esque version of Microsoft's full IDE Visual Studio. As Visual Studio Code is still under active development, some steps of this guide may need to be altered, so feel free to do so as the need arises.

Note: Even though Visual Studio Code is a cross-platform program, HPL3 is designed to be modded on a Windows machine. As such, the instructions in this guide assume that you are on a Windows OS. (Linux users who have successfully gotten the HPL3 modding tools may use this guide as a base to get Intellisense working on their platform.)

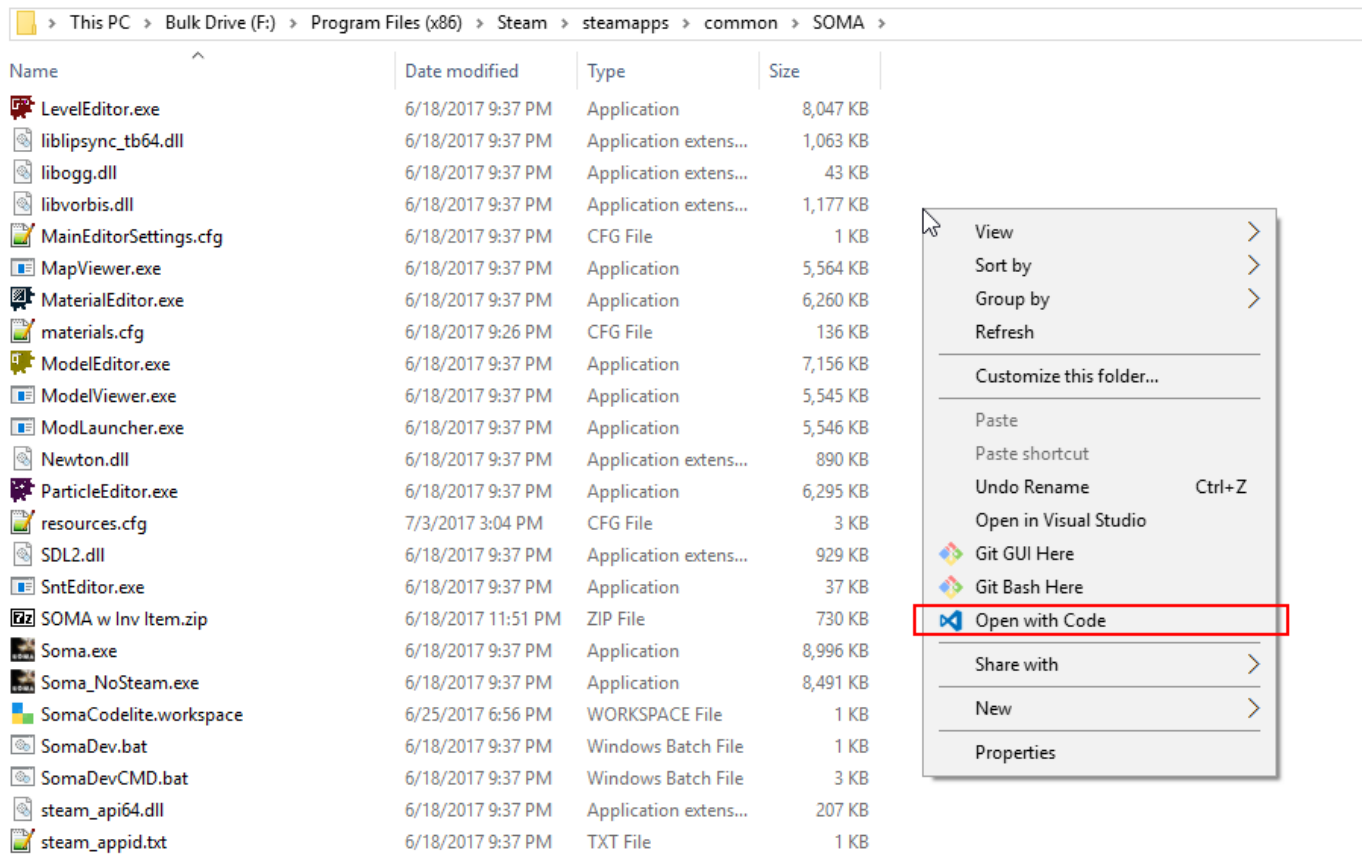
Install VS Code

You will need to download and install VS Code. You can do so from Microsoft's [VS Code website](#). Simply choose your desired platform and download the installer. Follow the instructions given in the installer to install VS Code on your computer.

Open SOMA Folder in VS Code

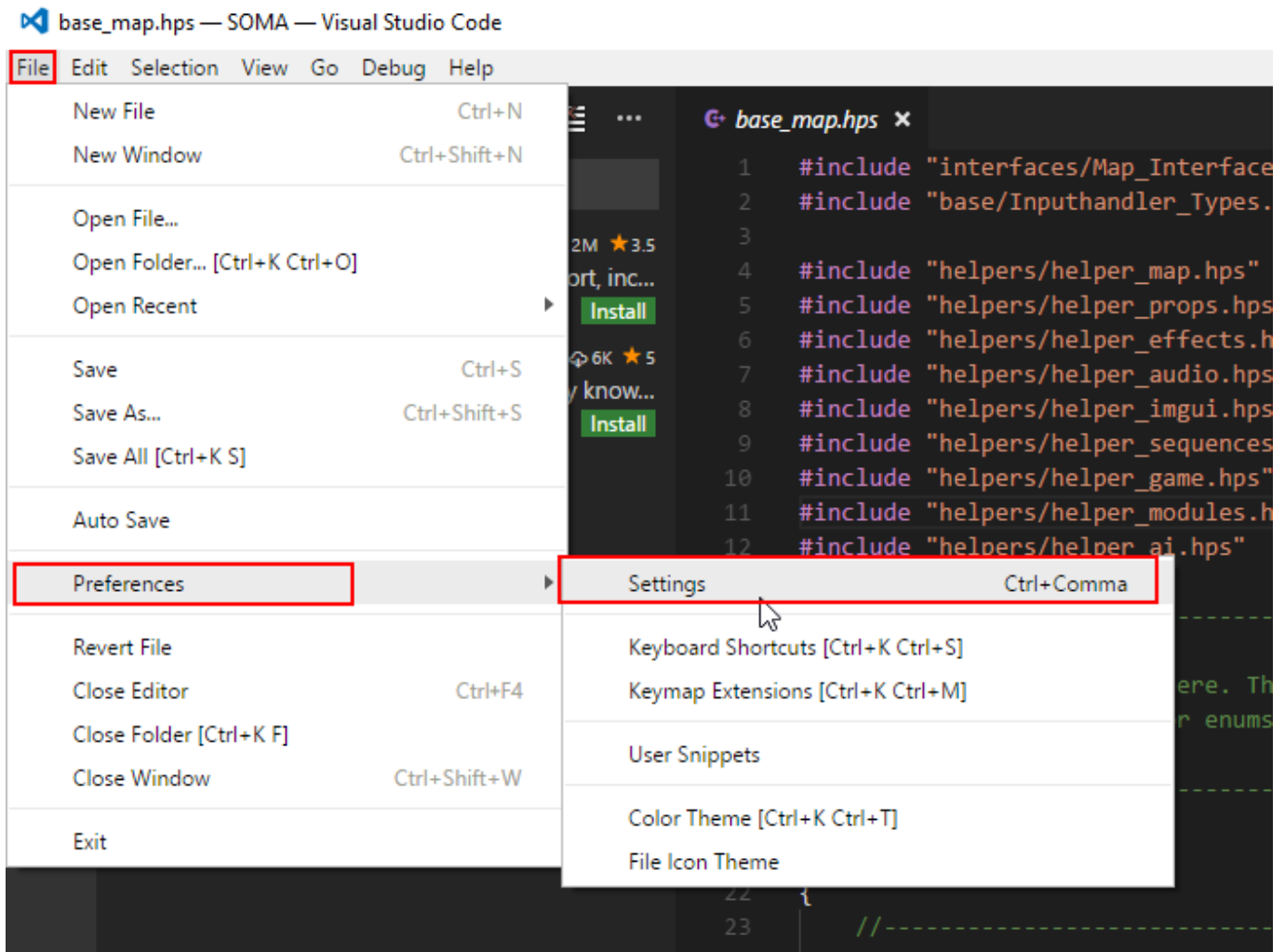
VS Code has the capability of opening an entire folder into its internal directory tree. Doing this gives you the benefit to be able to quickly and easily navigate to files in that folder and in subsequent folders. It also grants added benefits for inter-file relationships, such as C++ linking capabilities.

To do this, simply navigate to your SOMA installation folder and right-click on the window background. (i.e. Don't click on any file or folder within the SOMA folder.) Then select the "Open with Code" option from the drop down menu:



Associate HPS files with C++

The first thing to do is to associate HPL3 script files with VS Code's C++ syntax highlighting engine. To do so, you need to open VS Code's preferences file:

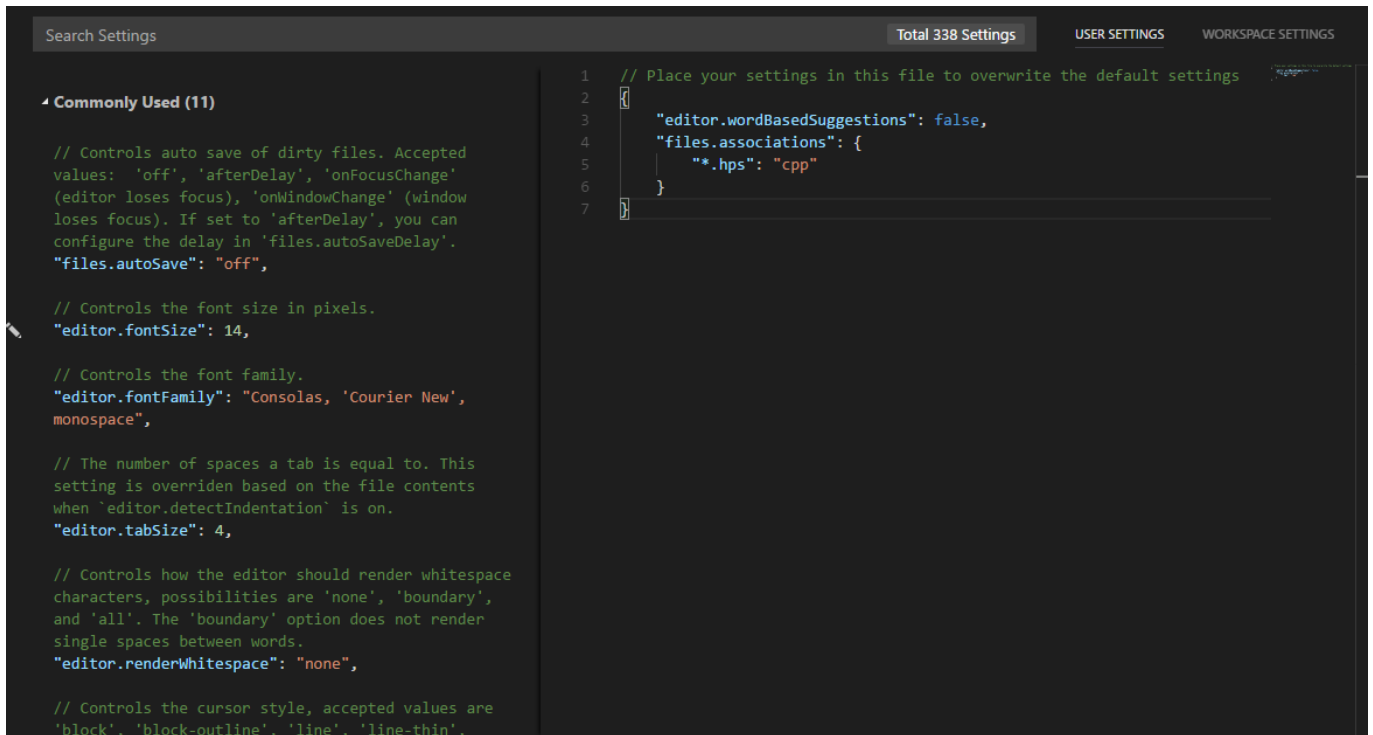


In VS Code, settings are saved in a JSON file. The window on the left is the VS Code default settings, while the window on the right is a JSON file for creating and saving user configurations.

To associate .hps files with C++, simply copy the following text into the user preferences window (on the right):

```
"files.associations": {  
  "*.hps": "cpp"  
}
```

Afterwards, your preferences file should look something like this:



```
Search Settings Total 338 Settings USER SETTINGS WORKSPACE SETTINGS

Commonly Used (11)

// Controls auto save of dirty files. Accepted values: 'off', 'afterDelay', 'onFocusChange' (editor loses focus), 'onWindowChange' (window loses focus). If set to 'afterDelay', you can configure the delay in 'files.autoSaveDelay'.
"files.autoSave": "off",

// Controls the font size in pixels.
"editor.fontSize": 14,

// Controls the font family.
"editor.fontFamily": "Consolas, 'Courier New', monospace",

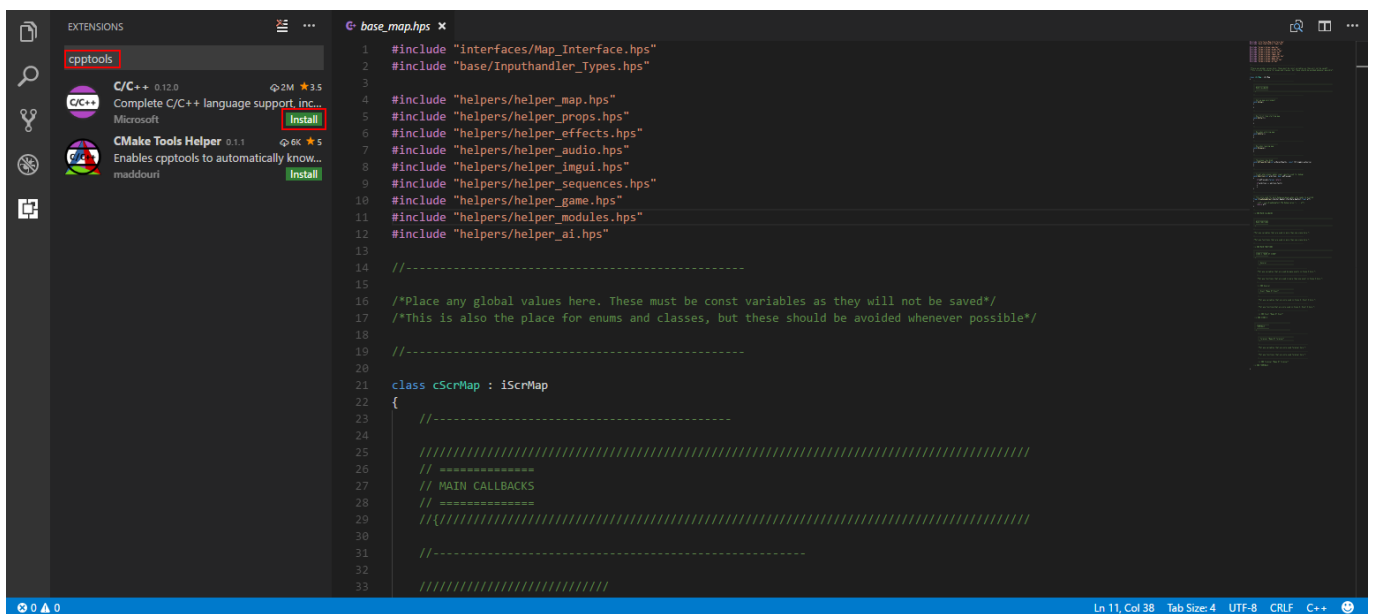
// The number of spaces a tab is equal to. This setting is overridden based on the file contents when 'editor.detectIndentation' is on.
"editor.tabSize": 4,

// Controls how the editor should render whitespace characters, possibilities are 'none', 'boundary', and 'all'. The 'boundary' option does not render single spaces between words.
"editor.renderWhitespace": "none",

// Controls the cursor style, accepted values are 'block', 'block-outline', 'line', 'line-thin',
```

Install CPPTools extension

Next, you will need to get the C/C++ extension for VS Studio. To do this, go to the Extensions tab on the left side of the window. In the search bar, search for “C/C++”. The extension you want should appear at the top of the list (it's authored by Microsoft). After you find it, click the green “Install” button:



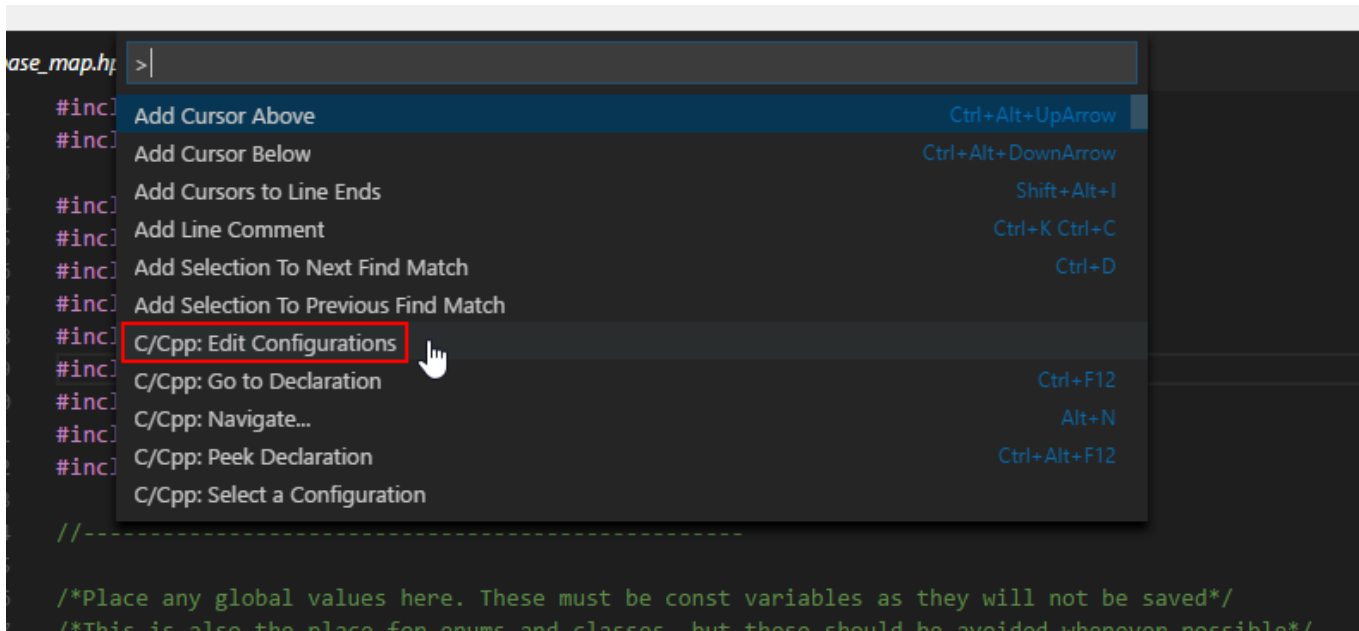
```
EXTENSIONS
cpptools
C/C++ 0.12.0 3M 3.5
Complete C/C++ language support, inc...
Microsoft Install
CMake Tools Helper 0.1.1 6K 5
Enables cpptools to automatically know...
maddouri Install

base_map.hps x
1 #include "interfaces/Map_Interface.hps"
2 #include "base/Inpuhandler_Types.hps"
3
4 #include "helpers/helper_map.hps"
5 #include "helpers/helper_props.hps"
6 #include "helpers/helper_effects.hps"
7 #include "helpers/helper_audio.hps"
8 #include "helpers/helper_imgui.hps"
9 #include "helpers/helper_sequences.hps"
10 #include "helpers/helper_game.hps"
11 #include "helpers/helper_modules.hps"
12 #include "helpers/helper_ai.hps"
13
14 //-----
15
16 /*Place any global values here. These must be const variables as they will not be saved*/
17 /*This is also the place for enums and classes, but these should be avoided whenever possible*/
18
19 //-----
20
21 class cScrMap : iScrMap
22 {
23     //-----
24     // *****
25     // MAIN CALLBACKS
26     // *****
27     //\\//-----
28     //-----
29     //-----
30     //-----
31     //-----
32     //-----
33     //-----
```

(After you install it, the window will need to be reloaded twice. The first time is after installing the extension, and the second time is after the extension downloads and installs its needed dependencies. Simply click on the “Reload” button as it appears and VS Code will restart itself for you, retaining your open files.)

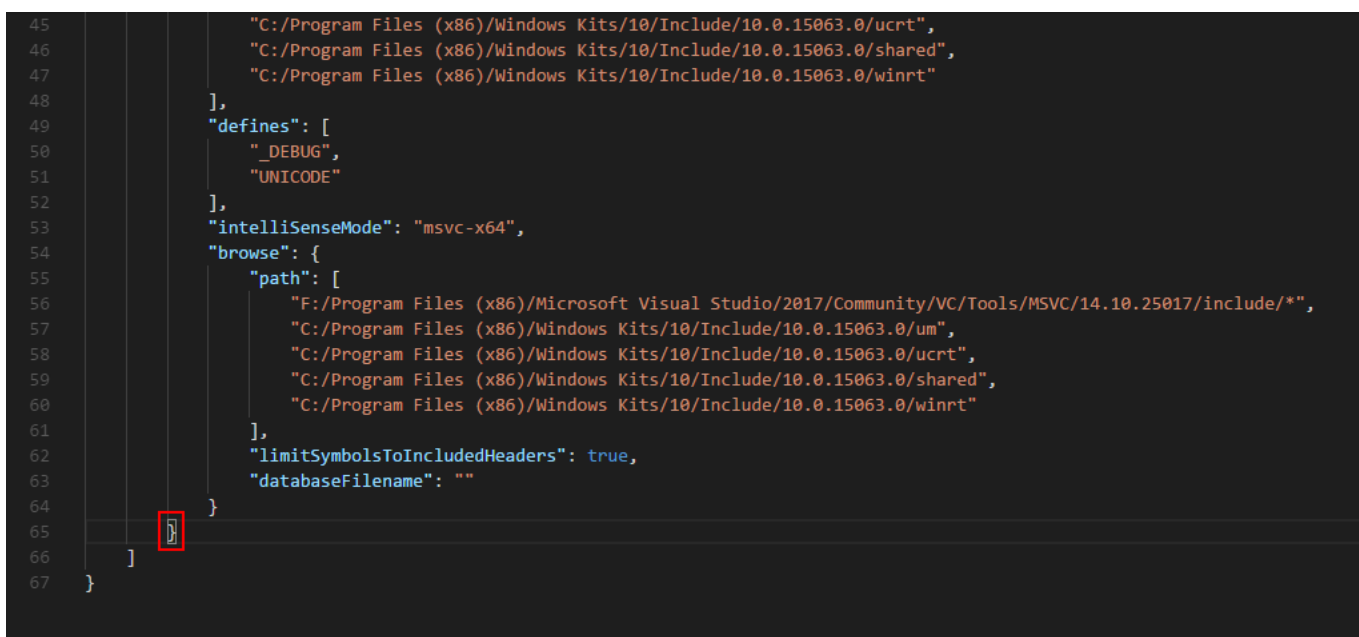
Configure The HPL3 Profile

Now you need to configure the extension to use the HPL3 script files for its Intellisense. To do this, open the Command Palette by pressing "Ctrl-Shift-P". From there, find and select the "C/Cpp: Edit Configurations" option:



On this page, you will see a lot of existing JSON text. This was automatically generated by the extension, and there's no need to change it. Instead, we are going to add a new profile for the Intellisense to use in our projects.

Scroll to the bottom and put the cursor after the closing bracket of the last profile. (It's the third bracket from the bottom.)



After that bracket, add a comma, add a new line (press Enter), then paste the following text:

```
{
  "name": "HPL3",
  "includePath": [
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/agents",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/areas",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/base",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/critters",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/custom_depth",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/custom_depth/helper_custom_depth_im
gui",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/effects",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/helpers",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/interfaces",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/modules",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/player",
    "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/props",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/utilities"
  ],
  "intelliSenseMode": "clang-x64",
  "browse":{
    "path":[
      "F:/Program Files (x86)/Steam/steamapps/common/SOMA",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/agents",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/areas",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/base",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/critters",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/custom_depth",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/custom_depth/helper_custom_depth_im
gui",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/effects",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/helpers",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/interfaces",
      "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/modules",
```

```

    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/player",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/props",
    "F:/Program Files
(x86)/Steam/steamapps/common/SOMA/script/utilities"
  ]
}
}

```

(Change the path in each one to the folder where your installation of SOMA is located, obviously.)

Afterwards, the text should look like this (the new text is highlighted):

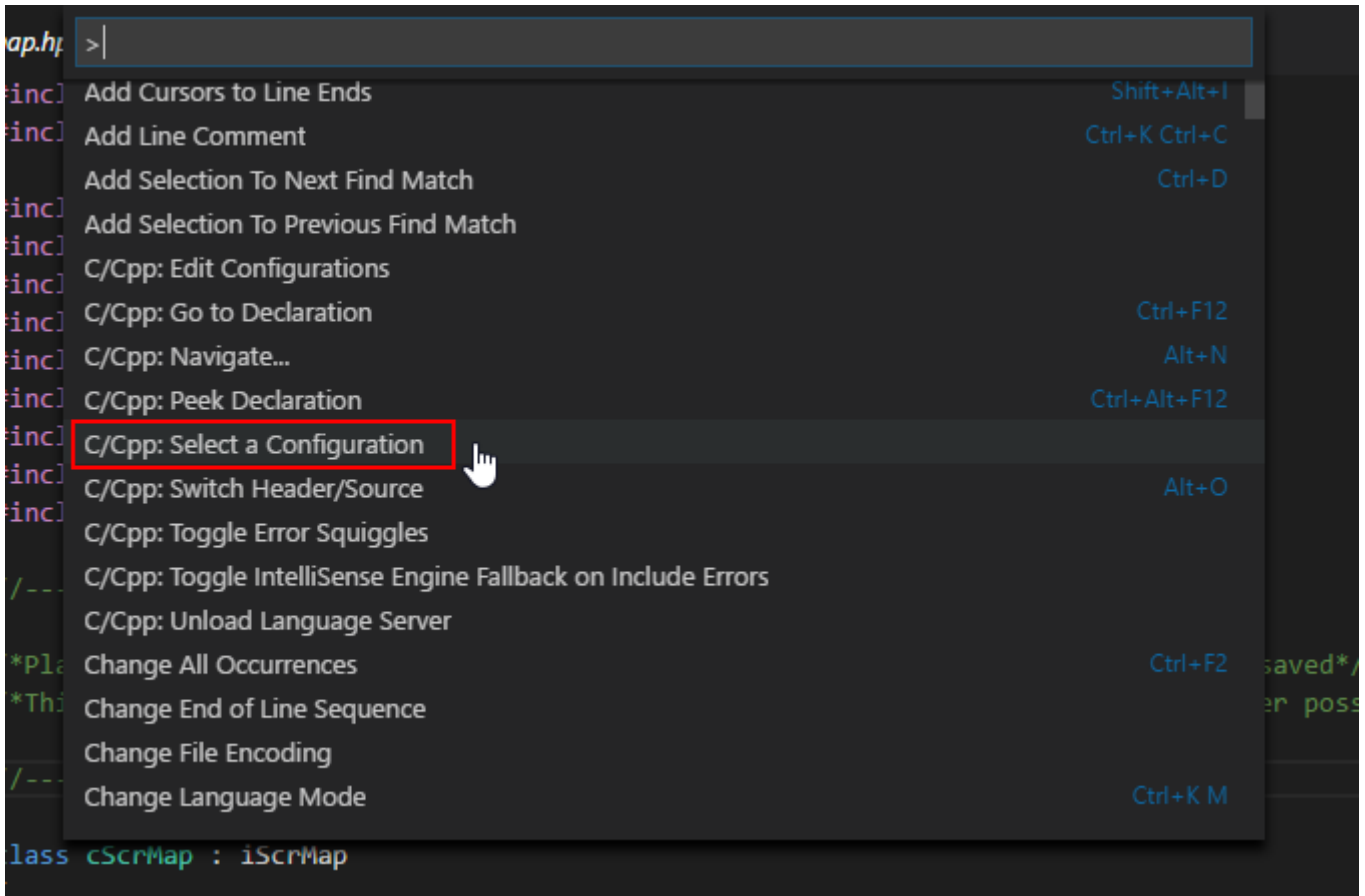
```

58     "C:/Program Files (x86)/Windows Kits/10/Include/10.0.15063.0/ucrt",
59     "C:/Program Files (x86)/Windows Kits/10/Include/10.0.15063.0/shared",
60     "C:/Program Files (x86)/Windows Kits/10/Include/10.0.15063.0/winrt"
61   ],
62   "limitSymbolsToIncludedHeaders": true,
63   "databaseFilename": ""
64 },
65 },
66 {
67   "name": "HPL3",
68   "includePath": [
69     "F:/Program Files (x86)/Steam/steamapps/common/SOMA",
70     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/agents",
71     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/areas",
72     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/base",
73     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/critters",
74     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/custom_depth",
75     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/custom_depth/helper_custom_depth_imgui",
76     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/effects",
77     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/helpers",
78     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/interfaces",
79     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/modules",
80     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/player",
81     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/props",
82     "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/utilities"
83   ],
84   "intelliSenseMode": "clang-x64",
85   "browse": {
86     "path": [
87       "F:/Program Files (x86)/Steam/steamapps/common/SOMA",
88       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/agents",
89       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/areas",
90       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/base",
91       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/critters",
92       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/custom_depth",
93       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/custom_depth/helper_custom_depth_imgui",
94       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/effects",
95       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/helpers",
96       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/interfaces",
97       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/modules",
98       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/player",
99       "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/props",
100      "F:/Program Files (x86)/Steam/steamapps/common/SOMA/script/utilities"
101    ]
102  }
103 }
104 ]
105 }

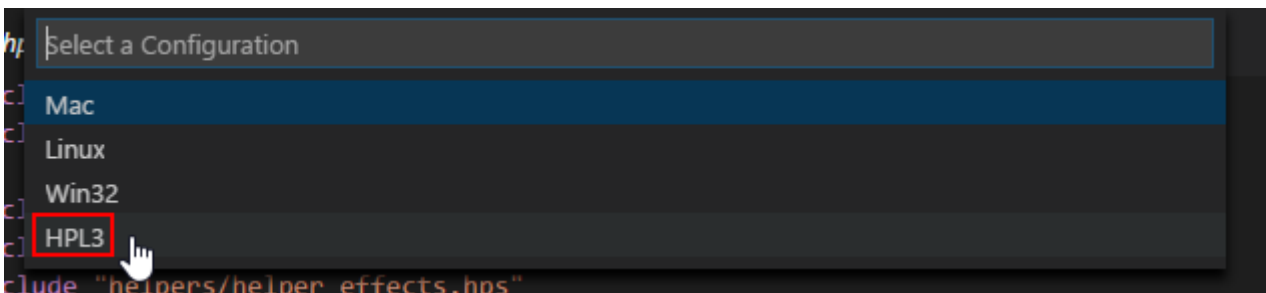
```

(If your text has any red squiggles in it, then you copied the text into the wrong place. Compare your text to the image above to make sure it looks the same.)

Now your profile is ready to go. Save and close the file, then go back to your map script. Open the Command Palette again (“Ctrl-Shift-P”) and navigate to “C/Cpp: Select a Configuration”:



Under the drop-down menu, select “HPL3”:



And that's it! You're done! Go ahead and test it out in your map script:


```
////////////////////////////////////
// Set up map environment
void Setup()
{
    Player_
}
//-----
//-----
//-----
// Run first
void OnStar
{
}
//-----
```

- Player_AddBodyForce void Player_AddBodyForce(con.. i
- Player_AddBodyForceAwayFromEntity
- Player_AddContinuosBodyForce
- Player_AddHealth
- Player_CancelJump
- Player_ChangeStateToCustomControls
- Player_ChangeStateToNormal
- Player_DecCrawlCount
- Player_DisableCameraLock
- Player_EnableCameraLock
- Player_EquipTool
- Player_ExitTerminal

From: <https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link: https://wiki.frictionalgames.com/hpl3/community/scripting/visual_studio_code_setup?rev=1514665172

Last update: **2017/12/30 20:19**

