Making GTK Functional in Geany on a Raspberry Pi

Geany Compile or Build Fault Report:

gtk/gtk.h: No such file of directory

```
gcc -Wall -c "geany_gtk_setup.c" (in directory: /home/pi/Documents/C_Programs)
geany_gtk_setup.c:14:10: fatal error: gtk/gtk.h: No such file or directory
 #include <gtk/gtk.h>
          ^~~~~~~~~~~
compilation terminated.
Compilation failed.
```

Raspberry Pi and Geany

- **The newer Raspberry Pi OS:**
  - https://www.raspberrypi.org/downloads/

  - GEANY v 1.33 pre-loaded
    - Code Name: "Gordon"

  - Decent little programmer
    - Versatile
    - Free
    - Works

Geany "C" or "C++" Programming

- Geany can be used to <u>write</u> code for both:
  - ▫ "C" and "C++" files.
  - ▫ Other languages as well.

- Geany can be used to <u>compile</u>   code for both:
  - ▫ "C" and "C++" files.
  - ▫ Other languages as well.

- Geany can be used to <u>build</u>   code for both:
  - ▫ "C" and "C++" files.
  - ▫ Other languages as well.

Geany – GTK Compile or Build Fault

- Out of the box.. when trying to "Compile" or "Build" a C (.c) or C++ (.cpp) type file which includes the following #include statement:

#include <gtk/gtk.h>

- The following
fatal error occurs:

```
gcc -Wall -c "geany_gtk_setup.c" (in directory: /home/pi/Documents/C_Programs)
geany_gtk_setup.c:14:10: fatal error: gtk/gtk.h: No such file or directory
 #include <gtk/gtk.h>
          ^~~~~~~~~~~
compilation terminated.
Compilation failed.
```

C Language Example
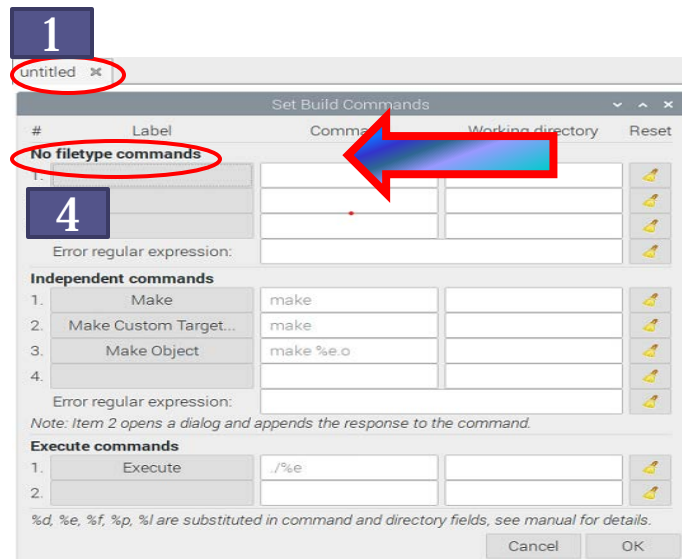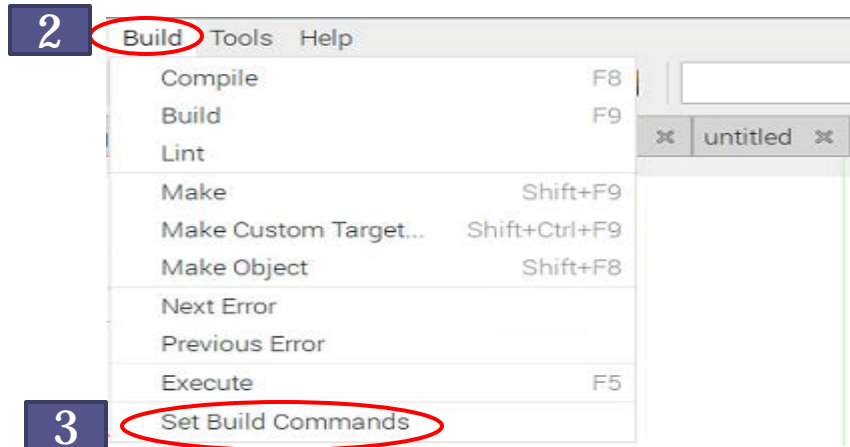

- **Compile / Build Error:**

```
gcc -Wall -c "geany_gtk_setup.c" (in directory: /home/pi/Documents/C_Programs)
geany_gtk_setup.c:14:10: fatal error: gtk/gtk.h: No such file or directory
 #include <gtk/gtk.h>
          ^~~~~~~~~~~
compilation terminated.
Compilation failed.
```

C++ Language Example

- **Compile / Build Error:**
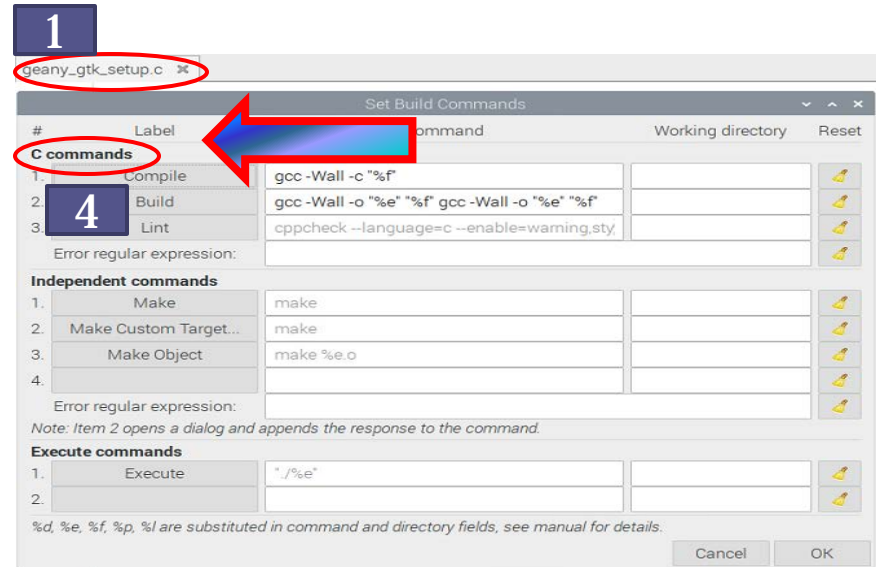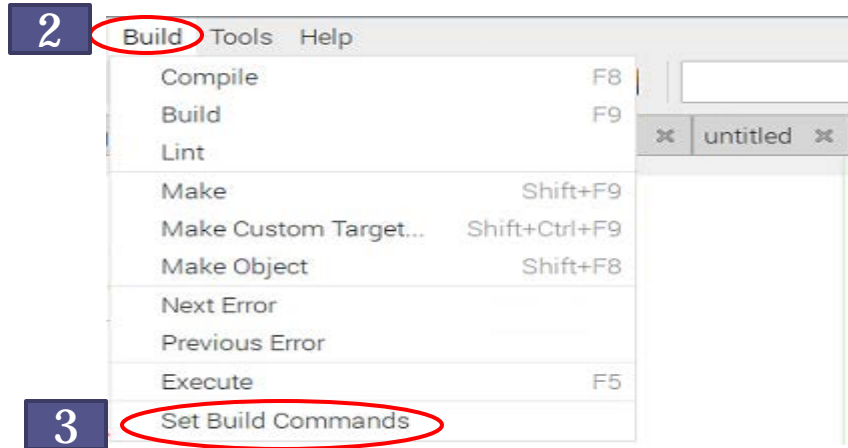
There is a Fix to this Problem

- **The key to resolving this issue:**

- Active window (1) is blank, or NOT a C/C++ file type so in this case you CANNOT access the File Type Commands (4) you <u>need</u> to change.

Load a C type of file in the programming window
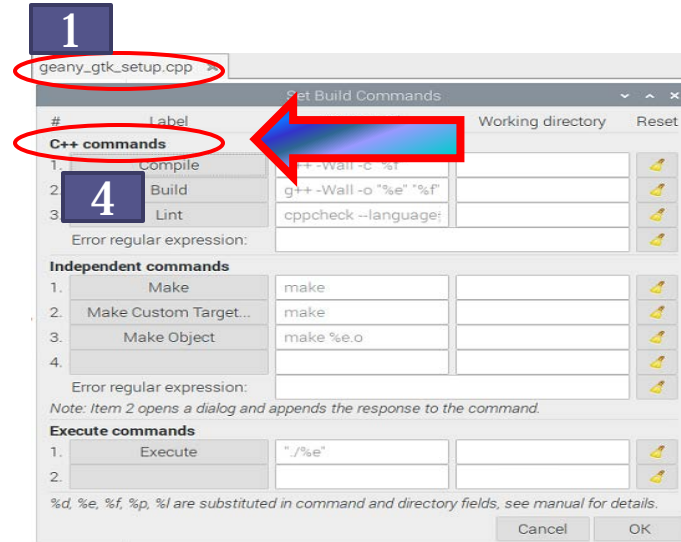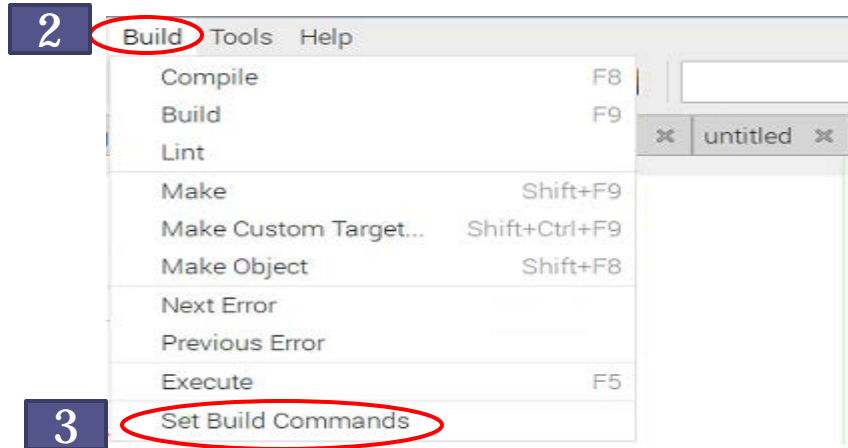
- **Load a "C" type file (3):**

- **Active window (1) is NOW a C file type so in this case you CAN access the C Commands (4) you <u>need</u> to change.**

Load a C++ type of file in the programming window

- Load a "C++" type file (3):

- Active window (1) is NOW a C++ file type so in this case you CAN access the C++ Commands (4) you <u>need</u> to change.

Make the following changes to BOTH File Type:

- **To the "Set Build Commands" for both of the "File Types" add some parameters to the default entries:**

- **3.0 for GTK 3.0   and   2.0 for GTK 2.0**

- **Compile:**
  - ▫ Add the following: `pkg-config --cflags gtk+-3.0`
    - • From: gcc -Wall -c "%f"
    - • To: gcc -Wall -c "%f" `pkg-config --cflags gtk+-3.0`

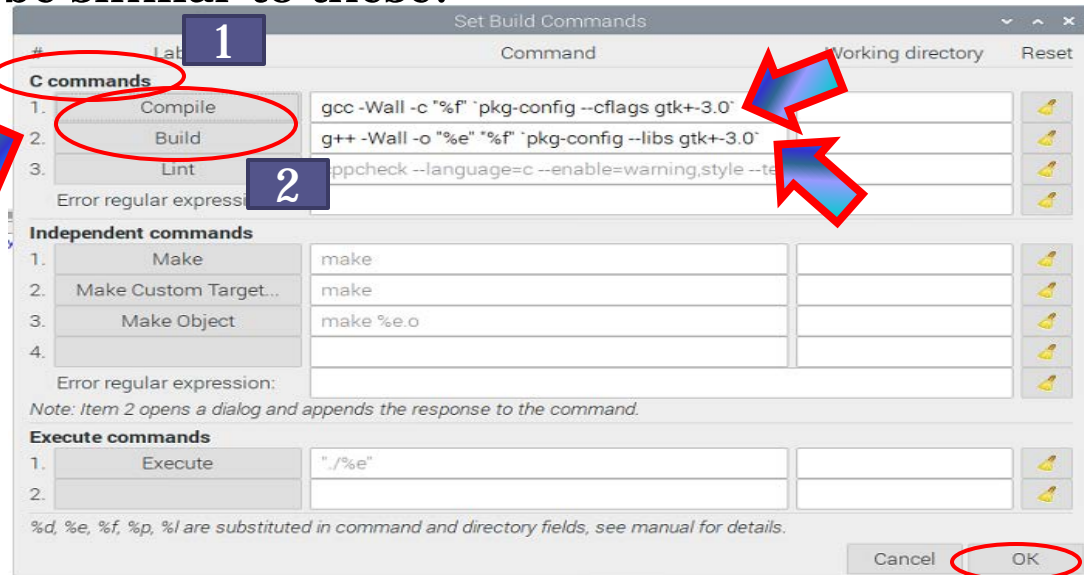- **Build:**
  - ▫ Add the following: `pkg-config --libs gtk+-3.0`
    - • From: gcc -Wall -o "%e" "%f"
    - • To: gcc -Wall -o "%e" "%f" `pkg-config --libs gtk+-3.0`

Entries Results (stretch right side of box):

- **The "Set Build Commands" for <u>**each**</u> of the two "File Types" should now be similar to these:**

- **Do this for each C and for C++**

- **For each instance click on OK to save entries.**
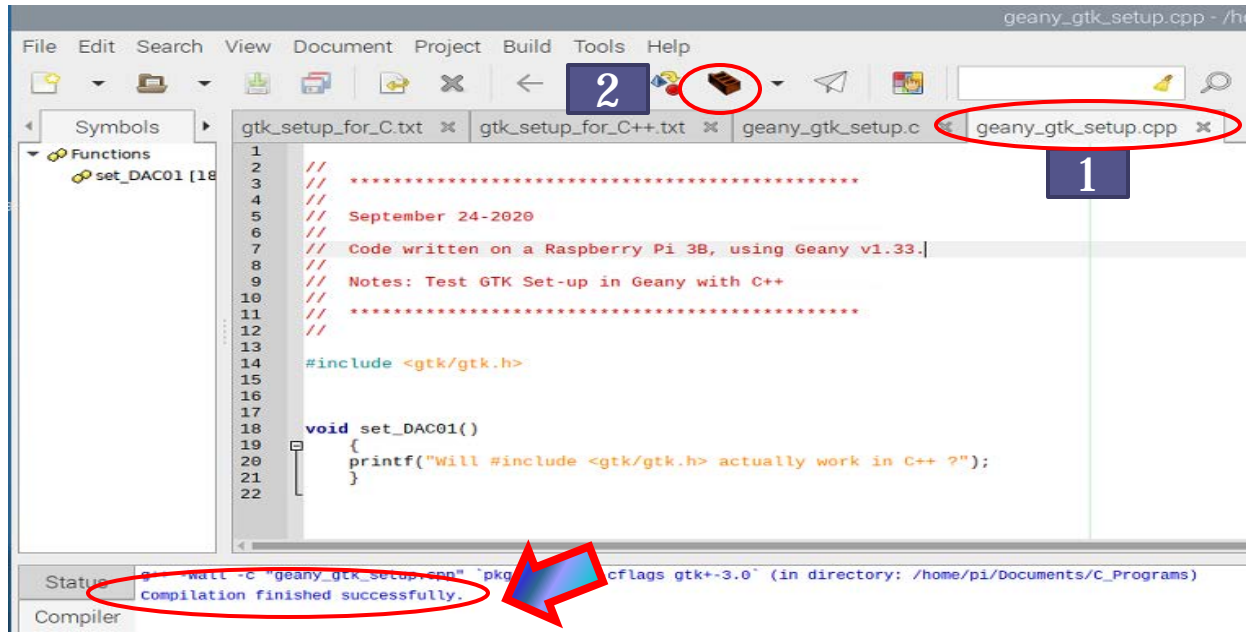
Test: C - Compile / Build Results:

- **Try compiling or building a simple C file:**

Test: C++ - Compile / Build Results:

- **Try compiling or building a simple C++ file:**

Synopsys:

Adding the entries as shown to both the COMPILE and the BUILD parameters, for each type of file types resolved the

#include <gtk/gtk.h>

Issues that I had encountered when trying to compile or build in C or C++.

Give this a try, and hopefully you'll have the same success I had.

Thanks for Watching:

Please LIKE this video and feel free to add your comments / suggestions.

and

SUBSCRIBE to my Channel.

YouTube Video: https://youtu.be/29OYRhWEdAQ