# Using Vim and C toolchain

Using Vim and the C toolchain usually means:

1. Edit code with Vim.
2. Compile with GCC or Clang.
3. Run & debug the resulting program.
4. Automate builds with **make**.

## Creating a file in C, compiling using gcc and clang and running it

**Objective**

Learn how to edit, compile, run, and manage C programs using Vim, GCC/Clang, and Makefiles.

Install **Vim** if it is not already installed.









This is the content of the file hello.c:

A screen shot of a computer code

AI-generated content may be incorrect.

Compile first and run the program.

A screen shot of a computer

AI-generated content may be incorrect.

The same could be accomplished using just one line (compile and run in one step).

A black screen with white text

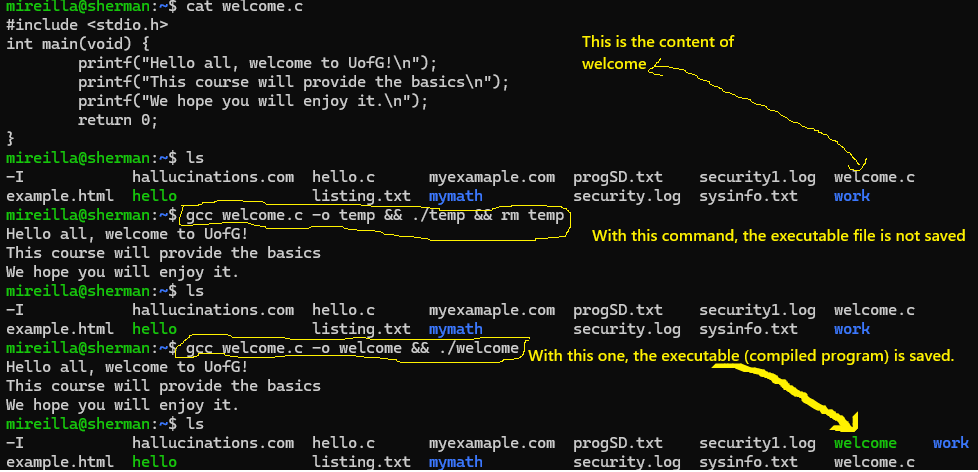
AI-generated content may be incorrect.

**gcc hello.c -o hello** compile the program.

**&&**  only run the next command if the first one succeeds.

**./hello** run the newly compiled program.

If you do not want to keep the hello executable at all, you can compile, run, and delete the executable afterwards. It means the executable is not saved.



**Clang** is a modern, open-source **C, C++, and Objective-C compiler**. It does the same job as **gcc** (the GNU Compiler Collection), that is, turning the C source code into machine code you can run.

A black background with white text

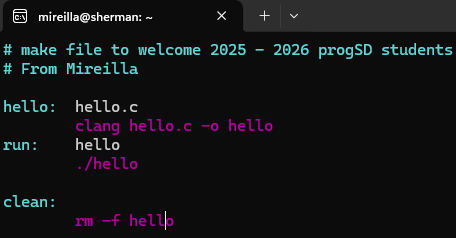
AI-generated content may be incorrect.

A black background with white text

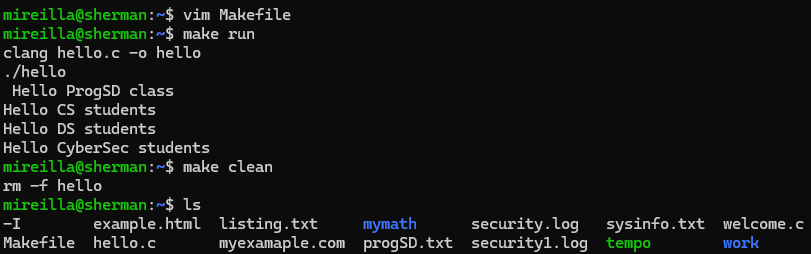
AI-generated content may be incorrect.

## Makefile

Creating a file called Makefile:



Then I will use **make run** to run it and **make clean** to delete the executable file.



Example 2

In this example, I included the command to extract the IP addresses from security1.log file.

A screen shot of a computer

AI-generated content may be incorrect.

The output when I run it.

A screen shot of a computer code

AI-generated content may be incorrect.