

# WHY DO WE NEED MAKEFILE?

[http://www.tutorialspoint.com/makefile/why\\_makefile.htm](http://www.tutorialspoint.com/makefile/why_makefile.htm)

Copyright © tutorialspoint.com

Compiling the source code files can be tiring, especially when you have to include several source files and type the compiling command every time you need to compile. Makefiles are the solution to simplify this task.

Makefiles are special format files that help build and manage the projects automatically.

For example, let's assume we have the following source files.

- main.cpp
- hello.cpp
- factorial.cpp
- functions.h

## main.cpp

```
#include <iostream>

using namespace std;

#include "functions.h"

int main(){
    print_hello();
    cout << endl;
    cout << "The factorial of 5 is " << factorial(5) << endl;
    return 0;
}
```

## hello.cpp

```
#include <iostream>

using namespace std;

#include "functions.h"

void print_hello(){
    cout << "Hello World!";
}
```

## factorial.cpp

```
#include "functions.h"

int factorial(int n){
    if(n!=1){
        return(n * factorial(n-1));
    }
    else return 1;
}
```

## functions.h

```
void print_hello();
int factorial(int n);
```

The trivial way to compile the files and obtain an executable, is by running the command –

```
gcc main.cpp hello.cpp factorial.cpp -o hello
```

This command generates *hello* binary. In this example we have only four files and we know the sequence of the function calls. Hence, it is feasible to type the above command and prepare a final binary. However, for a large project where we have thousands of source code files, it becomes difficult to maintain the binary builds.

The make command allows you to manage large programs or groups of programs. As you begin to write large programs, you notice that re-compiling large programs takes longer time than re-compiling short programs. Moreover, you notice that you usually only work on a small section of the program *such as a single function*, and much of the remaining program is unchanged.

~~In the subsequent section, we see how~~ to prepare a makefile for our project.

Loading [MathJax]/jax/output/HTML-CSS/jax.js