# Command Line Arguments in C

The arguments passed from command line are called command line arguments. These arguments are handled by main() function.

To support command line argument, you need to change the structure of main() function as given below.

1. **int** main(**int** argc, **char** \*argv[] )

Here, **argc** counts the number of arguments. It counts the file name as the first argument.

The **argv[]** contains the total number of arguments. The first argument is the file name always.

## **Example**

Let's see the example of command line arguments where we are passing one argument with file name.

#include <stdio.h>

**void** main(**int** argc, **char** \*argv[] )

 {

   printf("Program name is: %s\n", argv[0]);

**if**(argc < 2){

      printf("No argument passed through command line.\n");

   }

**else**{

      printf("First argument is: %s\n", argv[1]);

   }

}

Run this program as follows in Linux:

1. ./program hello

Run this program as follows in Windows from command line:

1. program.exe hello

Output:

Program name is: program

First argument is: hello

If you pass many arguments, it will print only one.

1. ./program hello c how r u

Output:

Program name is: program

First argument is: hello

But if you pass many arguments within double quote, all arguments will be treated as a single argument only.

1. ./program "hello c how r u"

Output:

Program name is: program

First argument is: hello c how r u

You can write your program to print all the arguments. In this program, we are printing only argv[1], that is why it is printing only one argument.