

# C #include

The #include preprocessor directive is used to paste code of given file into current file. It is used include system-defined and user-defined header files. If included file is not found, compiler renders error.

By the use of #include directive, we provide information to the preprocessor where to look for the header files. There are two variants to use #include directive.

1. #include <filename>
2. #include "filename"

The **#include <filename>** tells the compiler to look for the directory where system header files are held. In UNIX, it is \usr\include directory.

The **#include "filename"** tells the compiler to look in the current directory from where program is running.

## #include directive example

Let's see a simple example of #include directive. In this program, we are including stdio.h file because printf() function is defined in this file.

1. #include<stdio.h>
2. int main(){
3. printf("Hello C");
4. return 0;
5. }

Output:

```
Hello C
```

## #include notes:

**Note 1:** In #include directive, comments are not recognized. So in case of #include <a//b>, a//b is treated as filename.

**Note 2:** In #include directive, backslash is considered as normal text not escape sequence. So in case of #include <a\nb>, a\nb is treated as filename.

**Note 3:** You can use only comment after filename otherwise it will give error.