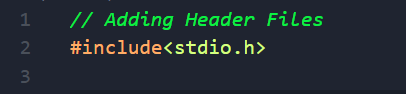
3. Managing Input and Output Operations

**Introduction to Input and Output Operations**

Reading, processing and writing are the three essential components in a computer program. Most of the programs that we write take input of data and display output of processed data. Like for example the printf function is used to display the output and the scanf. In C we include a header file (.h files) in the start to handle few functions. We include the stdio.h file (Standard Input and Output) to handle input and output by adding this line in the starting of the program.



This line above tells the compiler to search for the file called stdio.h which contains input and output function.

The functions found in stdio.h are as follows:

**Reading a Character**

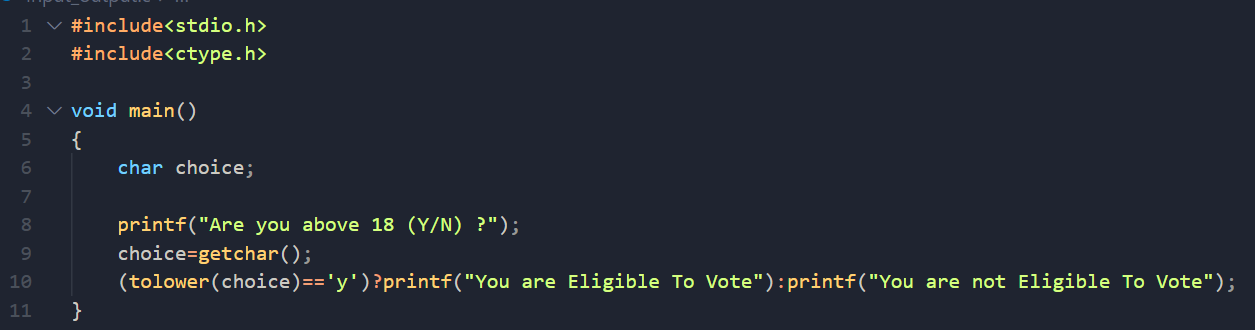
We can read data from the user by using the functions mentioned above that is as follows:

* Scanf
* Getc
* Gets
* Getchar

First, we will be seeing the getchar function. The syntax for using the getchar function is as follows:

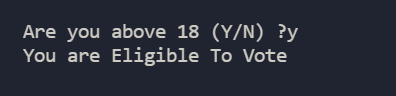
**<variable name>=getchar();**

To demonstrate the getchar function we will be writing a program that will display a yes/no question and accordingly display the output.

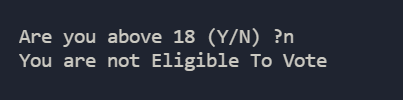


The output of the code is as follows:

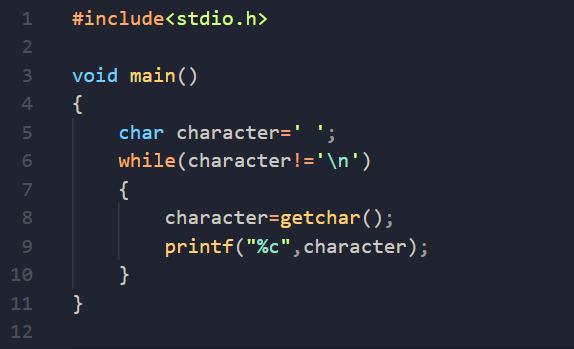
If choice is Y (Yes)



If choice is N (No)



We can get more than one character using the getchar function by calling it successively withing a while loop with condition. The condition is **character!=‘\n’**. The program is as follows:



**Writing a Character**

We can write data or show output to the screen by using the output functions like:

* Printf
* Puts
* Putc
* Putchar

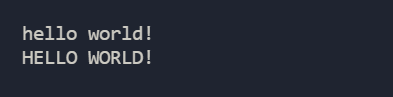
We will be seeing the putchar function. Like the getchar function the putchar function is used to display the output to the screen. The syntax for using the putchar function is as follows:

**putchar(<variable name>);**

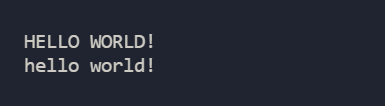
To demonstrate the putchar function we will be writing a program to swap the case of the given word that is if the letter is in upper case, it will convert it into lower case and vice versa.



The output of the above code is as follows:



From Lowercase to Uppercase



From Uppercase to Lowercase

**Formatted Input**

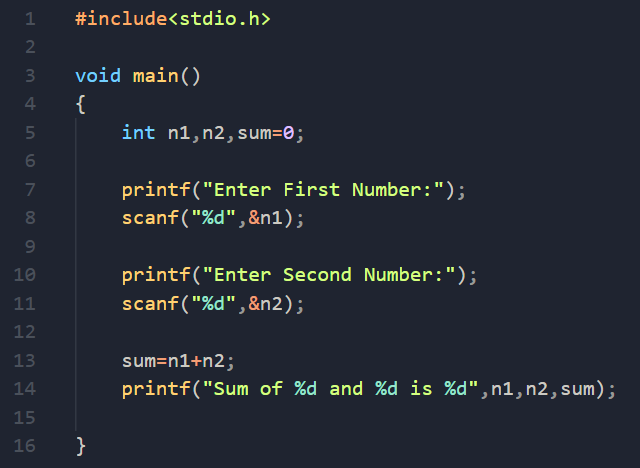
Formatted input refers to input data that has been arranged in a particular format. For example:

John, 12.23, 123

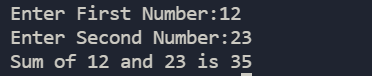
We can use the scanf() function for taking formatted input from the user. The full form of scanf is (scan format), the syntax for using the scanf function is:

**scanf (<Format Specifier>, &<variable>);**

Let’s write a program to take input of two integers and display the sum of it.



Output of the above code is as follows:



**Formatted Output**

Formatted Output refers to displaying an output in a particular format. For Example: John,34.55, 75 etc.

We can use the printf() function to display the output in a particular format. The full form of printf is print format. The syntax to use the printf() function is:

**printf (<Display Text or Format Specifier>, <variable>);**

In all the above examples we have used the printf() function to display the output in a particular format.