

## **C** Syntax

**Topics :** <u>C</u> **Written on** <u>April 12, 2023</u>

Look at below an example of C syntax with a simple program that adds two numbers.

```
#include <stdio.h>
int main() {
    int num1, num2, sum;
    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);
    sum = num1 + num2;
    printf("The sum of %d and %d is %d", num1, num2, sum);
    return 0;
}
```

In this program, we're using several elements of C syntax:

- **#include** <**stdio**.h>: This is a preprocessor directive that tells the compiler to include the standard input/output library, which provides functions like printf and scanf.
- int main(): This is the main function of our program, which is where the program starts executing. int specifies the return type of the function, and () indicates that the function takes no arguments.
- int num1, num2, sum; These are variable declarations. We're declaring three variables of type int: num1, num2, and sum.
- printf("Enter two numbers: ");: This is a function call that prints the specified string to the console.
- scanf("%d %d", &num1, &num2);: This is a function call that reads two integers from the
  console and stores them in the variables num1 and num2. The & symbol before each variable
  name is the address-of operator, which gives the memory address of the variable.
- sum = num1 + num2;: This is an assignment statement that calculates the sum of num1 and num2 and stores the result in the variable sum.
- printf("The sum of %d and %d is %d", num1, num2, sum);: This is a function call that prints a formatted string to the console. The %d placeholders are replaced with the values of num1, num2, and sum.
- return 0;: This statement terminates the main function and returns the value 0 to the operating system, indicating that the program completed successfully.

© Copyright **Aryatechno**. All Rights Reserved. Written tutorials and materials by <u>Aryatechno</u>