



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

← [\(https://www.educba.com/ternary-operator-in-c/\)](https://www.educba.com/ternary-operator-in-c/)

→ [\(https://www.educba.com/unary-operator-in-c/\)](https://www.educba.com/unary-operator-in-c/)



Introduction to Address Operator in C

The Address Operator in C also called a pointer. This address operator is denoted by "&". This symbol is called an ampersand. This & is used in a unary operator. The purpose of this address operator or pointer is used to return the address of the variable. Once we declared a pointer





(<https://www.educba.com/software-development/>)

address of the operator are known as pointers because they point to the variable in memory.

Usage:

Start Your Free Software Development Course

Web development, programming languages, Software testing & others

1. While scanning the user input we used ampersand operator.
2. While displaying the address of the variable we used ampersand operator.

Why Address Operator used in C?

Address operators commonly used for 2 purposes:

1. Conduct parameter passing by reference such as name.
2. Establish pointer values and address of operator's point to the memory location because the value of the pointer is the memory location or memory address. The data item saved in memory.

Real-time Example

- If the user is trying to locate the name "paramesh" within the data and the string variable named as name and it will look like `char[]="paramesh"`. Then the address operator is used to know the location or the address of the data using the "name" variable.

How does Address Operator work in C?



The address operator is working for returns the memory address of a variable. These addresses are returned by the address of the operator are known as pointers because they point to the



(<https://www.educba.com/software-development/>)

```
scanf("%d",&variable_name); //stores the value of the variable
```

Displaying the address of the variable

Code:

```
int a=10;  
int address=&a;
```

Examples to Implement Address Operator in C

Below are the examples mentioned:

Example #1

Scanning user integer input and display with the ampersand

Code:

```
//include is used to add basic C libraries  
#include <stdio.h>  
//main method is used to run C application  
int main(void)  
{  
  
    //declaring 2 variables  
    int first_number, second_number;  
    //Asking user to enter integer input
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
printf("User enter numbers %d and %d", first_number,
second_number);
return 0;
}
```

Output:

```
Please enter any 2 integer number
10
20
User enter numbers 10 and 20
```

Example #2

Scanning user String input and display with the ampersand

Code:

```
//include is used to add basic C libraries
#include <stdio.h>
//main method is used to run C application
int main(void)
{
//declaring 2 variables
char first_name[30],last_name[20];
//Asking user to enter input
printf("Please enter your first name = ");
//Store the first_name in ampersand symbol
scanf("%s",&first_name);
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
//displaying output to the end user
printf("Your name is %s %s ", first_name,last_name);
return 0;
}
```

Output:

```
Please enter your first name = Paramesh
Please enter your last name = Nathi
Your name is Paramesh Nathi
```

Example #3

Address of the String input names

Code:

```
//include is used to add basic C libraries
#include <stdio.h>

//main method is used to run C application
int main(void)
{
    //declaring 2 variables
    char first_name[30],last_name[20];
    //Asking user to enter input

    printf("Please enter your first name = ");
    //Store the first_name in ampersand symbol
    scanf("%s",&first_name);
```





(<https://www.educba.com/software-development/>)

```
//store the first_name in ampersand symbol
scanf("%s",&last_name);
//assigning first_name address to a variable
int b=&last_name;
//displaying output to the end user
//As it is address output may vary from compiler to compiler
printf("Address of Your name is %x %x ", a,a);
return 0;
}
```

🔗 Popular Course in this category



C Programming Training (3 Courses, 5 Project)

3 Online Courses | 5 Hands-on Projects | 34+ Hours | Verifiable Certificate of Completion | Lifetime Access

★★★★★ 4.5 (8,604 ratings)

Course Price

\$79 ~~\$399~~

[View Course](#)

(<https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1>)

Related Courses

C++ Training (4 Courses, 5 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/c-course/?btnz=edu-blg-inline-banner1>)

Java Training (10 Courses, 20 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/java-course/?btnz=edu-blg-inline-banner1>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
Please enter your first name = Amardeep
Please enter your last name = Patil
Address of Your name is 9beb76f0 9beb76f0
```

Example #4

Address of the address

Code:

```
//include is used to add basic C libraries
#include <stdio.h>

//main method is used to run C application
int main(void)
{
    //declaring 3 variables
    int *x;
    int **y;
    int input;

    //Asking user to enter input
    printf("Please enter a number = ");

    //Store the first_name in ampersand symbol
    scanf("%d",&input);

    //take the address of the input into the x single pointer

    x=&input;

    //take the address of the x into the y double pointer, it will
    give the address of the address
```





(<https://www.educba.com/software-development/>)

```
printf("Address of *x is %x", &x);
printf("Address of **x is %x", &y);
return 0;
}
```

Output:

```
Please enter a number = 12
Value of input is 12
Address of *x is bf32d4e0
Address of **x is bf32d4e8
```

Conclusion

Address operator is used to storing the address of the variable in C. This is denoted by an ampersand (&). This is also used for scanning the user input.

Recommended Articles

This is a guide to Address Operator in C. Here we discuss the introduction to Address Operator in C, why it is used and how does it work with examples. You can also go through our other related articles to learn more –

1. [C Operators \(https://www.educba.com/c-operators/\)](https://www.educba.com/c-operators/)
2. [Conditional Operator in C \(https://www.educba.com/conditional-operator-in-c/\)](https://www.educba.com/conditional-operator-in-c/)
3. [Arithmetic Operators in C \(https://www.educba.com/arithmetic-operators-in-c/\)](https://www.educba.com/arithmetic-operators-in-c/)
4. [Conditional Operator in Java \(https://www.educba.com/conditional-operator-in-java/\)](https://www.educba.com/conditional-operator-in-java/)



C PROGRAMMING TRAINING (3 COURSES, 5



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

☒ Verifiable Certificate of Completion

☒ Lifetime Access

Learn More

<https://www.educba.com/software-development/courses/c-programming-course/?btnz=educba-blg-inline-banner3>

About Us

Blog (<https://www.educba.com/blog/?source=footer>)

Who is EDUCBA? (<https://www.educba.com/about-us/?source=footer>)

Sign Up (<https://www.educba.com/software-development/signup/?source=footer>)

Corporate Training (<https://www.educba.com/corporate/?source=footer>)

Certificate from Top Institutions (<https://www.educba.com/educbalive/?source=footer>)

Contact Us (<https://www.educba.com/contact-us/?source=footer>)

Verifiable Certificate (<https://www.educba.com/software-development/verifiable-certificate/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

Apps

iPhone & iPad (<https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8>)

Android (<https://play.google.com/store/apps/details?id=com.educba.www>)

Resources

Free Courses (<https://www.educba.com/software-development/free-courses/?source=footer>)

Java Tutorials (<https://www.educba.com/software-development/software-development-tutorials/java-tutorial/?source=footer>)

Python Tutorials (<https://www.educba.com/software-development/software-development-tutorials/python-tutorial/?source=footer>)

All Tutorials (<https://www.educba.com/software-development/software-development-tutorials/?source=footer>)

Certification Courses

All Courses (<https://www.educba.com/software-development/courses/?source=footer>)

Software Development Course - All in One Bundle
(<https://www.educba.com/software-development/courses/software-development-course/?source=footer>)

Become a Python Developer (<https://www.educba.com/software-development/courses/python-certification-course/?source=footer>)

Java Course (<https://www.educba.com/software-development/courses/java-course/?source=footer>)

Become a Selenium Automation Tester (<https://www.educba.com/software-development/courses/selenium-training-certification/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

PHP Course (<https://www.educba.com/software-development/courses/php-course/?source=footer>)

© 2022 - EDUCBA. ALL RIGHTS RESERVED. THE CERTIFICATION NAMES ARE THE TRADEMARKS OF THEIR RESPECTIVE OWNERS.

