

Variables & Data Types In C: C Tutorial In Hindi #6



Course: C Language Tutorials For Beginners ⊕

[Overview](#) [Q&A](#) [Downloads](#) [Announcements](#)

Variables & Data Types In C: C Tutorial In Hindi #6

So guys in this tutorial, we are going to learn about variables and go over various data types. This tutorial is mostly going to be theoretical, and we will only touch the code for the purpose of understanding, except for that we will not be performing any coding related work, as theory is what makes your basis strong and firm foundation can help you grasp the coding part more efficiently.

As we have already discussed in the previous tutorial while going through identifiers that variables are nothing more than simple names given to a specific space in memory for reservation. I will get into more detail about it with the help of an example but first, let us cover some basics.

Declaration:

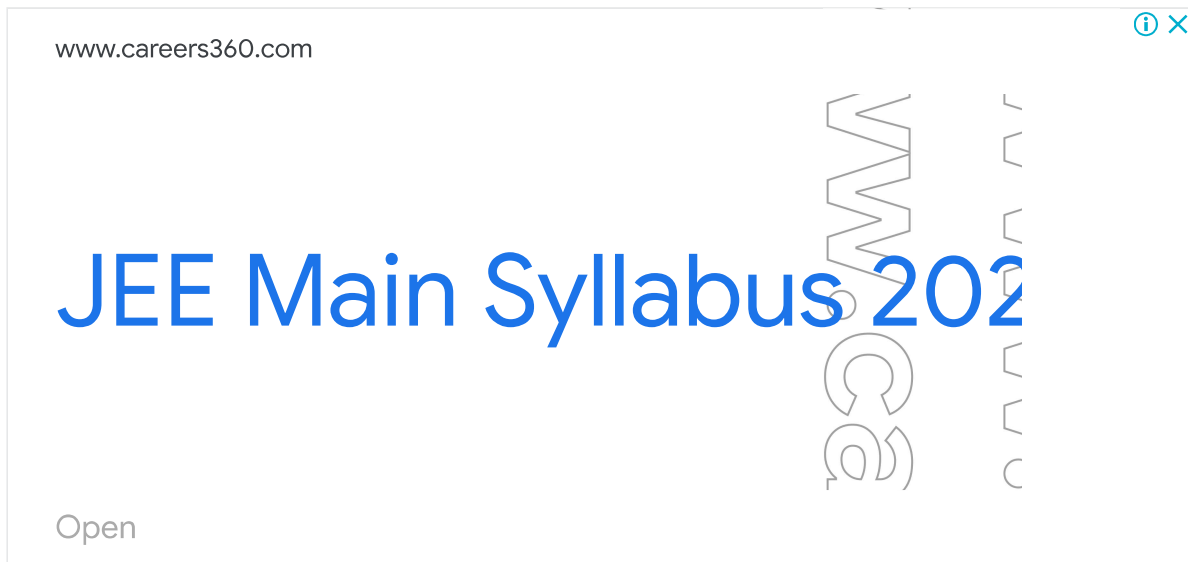
We cannot declare a **variable** without specifying its **data type**. The data type of a variable depends on what we want to store in the variable and how much space we want it to hold. The syntax for declaring a variable is simple:

```
data_type variable_name;
```

or

```
data_type variable_name = value;
```

the data type can be int, float, char, depending on what kind of value we want to store.



Naming a Variable:

A variable name can be of anything we want to call out variable. Yet there are specific rules we must follow while naming a variable:

- A variable name can contain **alphabets**, **digits**, and **underscore** (-) only.
- The starting letter can not be a **digit**.
- **White spaces** cannot be used.
- The name should not be **reserved keyword or special character**.

We can declare and assign value to a variable in two ways.

1st way:

```
int a = 12;
```

2nd way:

```
int a;  
a = 12;
```

Both of these have exactly the same working.

A variable as it names define can be altered, or its value can be changed, but same is not true for its type. If a variable is of integer type, then it will only store an integer value, which means that we cannot assign a character type value to an integer variable. We can not even store a decimal value into an integer variable.

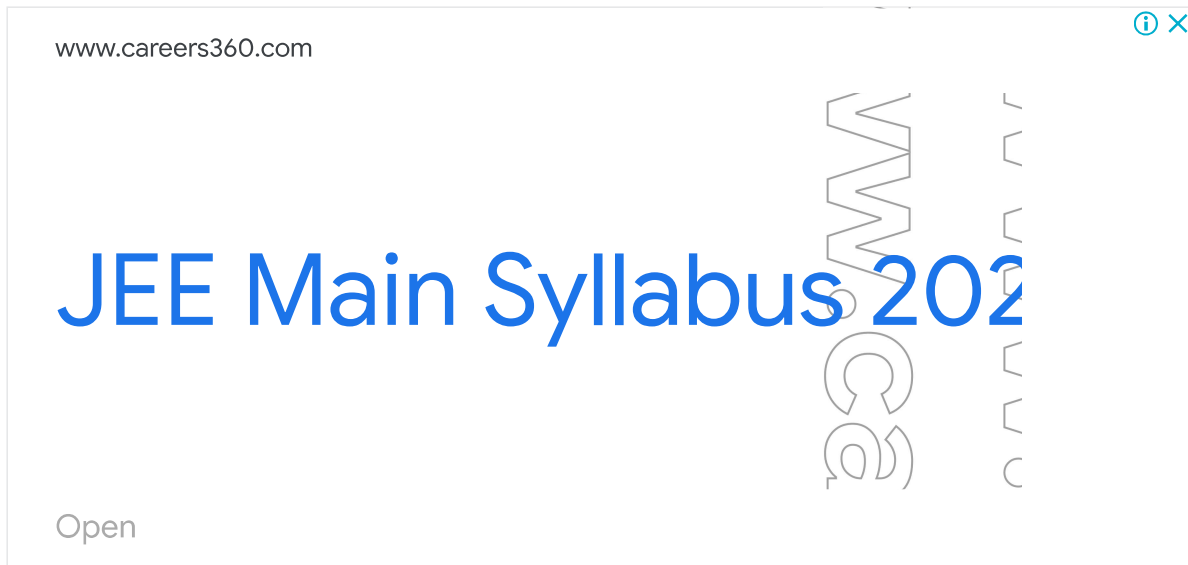
Let's see this with an example:

Example 1:

```
#include <stdio.h>  
  
int main()  
{  
    int a = 12.2221;  
    printf("Output = %d" , a);  
    return 0;  
}
```

We are sending 12.2221 as a value in a, but since it is an integer type variable, the output will be only 12.

Output = 12



Example 2:

```
#include <stdio.h>

int main()
{
    float a = 12.2221;
    printf("Output = %f" , a);
    return 0;
}
```

Here we are using float as a data type. In this case, you can see the output below is 12.222100

Output = 12.222100

Note that we used %f instead of %d in case of float.

The reason is that int can store only **2 bytes** worth data as its storage capacity is 2 bytes while float storage capacity is **4 bytes**.

DATA TYPE	MEMORY (BYTES)	RANGE
Char	1	-128 to 127

signed char	1	-128 to 127
-------------	---	-------------

CodeWithHarry



short int	2	-32,768 to 32,767
-----------	---	-------------------

unsigned short int	2	0 to 65,535
--------------------	---	-------------

unsigned int	4	0 to 65,535
--------------	---	-------------

int	2	--32,768 to 32,767
-----	---	--------------------

long int	4	-2,147,483,648 to 2,147,483,647
----------	---	---------------------------------

unsigned long int	4	0 to 4,294,967,295
-------------------	---	--------------------

float	4	
-------	---	--

double	8	
--------	---	--

long double 10

Code as described/written in the video

```
#include <stdio.h>

int main()
{
    printf("%lu", sizeof(int));
    return 0;
}
```

[Previous](#)[Next](#)

CodeWithHarry

Copyright © 2022 CodeWithHarry.com

