|  |
| --- |
| **CHAPTER** 3 |

**3. C Basic Syntax**

* Tokens in C
* Comments
* Semicolon (;)
* Whitespace in C

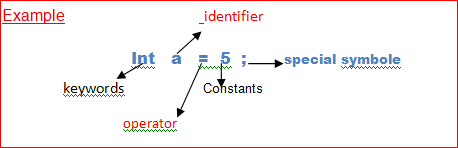
**[TOKENS]**

Tokens are the smallest individual unit of a program. A C program is made up of

different **tokens** combined together.

**C support six type of tokens:**

1. [Identifiers](https://www.w3schools.in/c-tutorial/identifiers/) ( A C identifier is a name used to identify a variable, function, or any other user-defined item.)
2. [Constants](https://www.w3schools.in/c-tutorial/constants/) ()
3. [Strings](https://www.w3schools.in/c-tutorial/strings/) Literals (String literal or string constant is a line of characters enclosed by double quotes)
4. Special Symbols
5. Keywords
6. Operators



Note Count total number of tokens in given program.

#include<stdio.h> //

void Main()

{

int a=3;

Printf(“hello hi %d”,a);

}

**[IDENTIFIRE IN C**] f

A C identifier is a name used to identify a variable, function, or any other user-defined item.etc

Rules of naming Identifier

* Identifier names must be unique
* An identifier can only have alphanumeric characters (a-z , A-Z , 0-9) (i.e. letters & digits) and underscore( \_ ) symbol.
* The first character must be an alphabet or underscore.
* You cannot use a keyword as identifiers.
* Only the first thirty-one (31) characters are significant.
* It must not contain white spaces.
* Identifiers are case-sensitive.

**Why w use identifiers**

Suppose we take

Int x= 10; // here we can’t determine why we take X as a variable name . so we use identifier to identify the variable name.

If we take int age = 10; // here we can easily determine we take age variable for storing the age in this variable

# EXAMPLE{

int amount;

double totalbalance;

}

In the above example amount and totalbalance are identifiers and int and double are keyword

**[CONSTENT IN C**]

Constants in C are the fixed value that are used in a program. We can’t change these value during the execution of program.

Like a=5 (here 5 is a fixed value we can’t changed during the execution of program.)

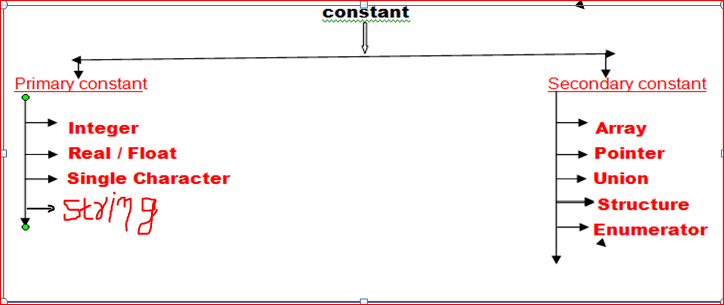
* Constants are also called literals. Integer, character etc these are literals or
* Constants can be any of the [data types](https://www.w3schools.in/c-tutorial/data-types/). may be data types.

Note 🗒

* The value of variable can be changed during the execution of program.
* The value of constant can never be changed during the execution of program.

C & C++ supports several kind of literal constants like Int , char , floating point number & strings. Literal constant do not have memory location.

# Type of constant.



**[KEYWORDS IN C**]

Keywords are the words whose meaning has already been explained to the c compiler.

You can't use a keyword as an identifier in your C programs, its reserved words in C library and used to perform an internal operation. The meaning and working of these keywords are already known to the compiler. It is also known as predefine words or reserved words.

**List of keywords**

A list of 32 reserved keywords in c language is given below:

Double , int , char, float, void these keywords are also a primitive data type.

Notes

* We can’t use these keywords as a variable name.

Ex -: int int= 10;

* These keywords are written in small latter.

|  |  |  |  |
| --- | --- | --- | --- |
| auto | Double | int | struct |
| break | Else | long | switch |
| case | Enum | register | typedef |
| char | Extern | return | union |
| const | Float | short | unsigned |
| continue | For | signed | Void |
| default | Goto | sizeof | volatile |
| do | If | static | while |

**BACKSLASH CHARACTER CONSTANT**

|  |  |
| --- | --- |
| Constants | Meaning |
| \a | beep sound |
| \b | Backspace |
| \f | form feed |
| \n | new line |
| \r | carriage return |
| \t | horizontal tab |
| \v | vertical tab |
| \' | single quote |
| \" | double quote |
| \\ | Backslash |
| \0 | Null |

C supports some character constants having a backslash in front of it. The lists of backslash characters have a specific meaning which is known to the compiler. They are also termed as "Escape Sequence".

For Example:

\t is used to give a tab

\n is used to give a new line

**[COMMENT IN C**]

Comments is used to increase the READABILITY of the program.

* 1. Single line comment.

Ex- //hi I am anurag.

* 1. Multi line comment.

Ex- /\* hi am anurag

I am from bihar \*/

Comment cannot be nested.

Ex- /\* cal of Si /\* Sam date\*/\*/

## [Semicolons]

In a C program, the semicolon is a statement terminator. That is, each individual statement must be ended with a semicolon. It indicates the end of one logical entity.

#include <stdio.h>

int main()

{

int a=3;

printf("hello hi: %d", a);

return 0;

}

*//semicolon(;) is use to terminate the statement.*