SINGH
COMPUTER
CENTER

Learn C

C is a general-purpose programming language, developed in 1972, and still quite popular.

C is very powerful; it has been used to develop operating systems, databases, applications, etc.

C Introduction

What is C?

C is a general-purpose programming language created by Dennis Ritchie at the Bell Laboratories in 1972.

It is a very popular language, despite being old. The main reason for its popularity is because it is a fundamental language in the field of computer science.

C is strongly associated with UNIX, as it was developed to write the UNIX operating system.

Why Learn C?

- It is one of the most popular programming languages in the world
- If you know C, you will have no problem learning other popular programming languages such as Java, Python, C++, C#, etc, as the syntax is similar
- C is very fast, compared to other programming languages, like Java and Python
- C is very versatile; it can be used in both applications and technologies

Difference between C and C++

- C++ was developed as an extension of C, and both languages have almost the same syntax
- The main difference between C and C++ is that C++ support classes and objects, while C does not

Get Started With C

To start using C, you need two things:

- A text editor, like Notepad, to write C code
- A compiler, like GCC, to translate the C code into a language that the computer will understand

There are many text editors and compilers to choose from. In this tutorial, we will use an ***IDE***

C Syntax

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!");
  return 0;
}
```

Example explained

Line 1: `#include <stdio.h>` is a **header file library** that lets us work with input and output functions, such as `printf()` (used in line 4). Header files add functionality to C programs.

Line 2: A blank line. C ignores white space. But we use it to make the code more readable.

Line 3: Another thing that always appear in a C program is `main()`. This is called a **function**. Any code inside its curly brackets `{}` will be executed.

Line 4: `printf()` is a **function** used to output/print text to the screen. In our example, it will output "Hello World!".

Note that: Every C statement ends with a semicolon `;`

Note: The body of `int main()` could also been written as:
`int main(){printf("Hello World!");return 0;}`

Remember: The compiler ignores white spaces. However, multiple lines makes the code more readable.

Line 5: `return 0` ends the `main()` function.

Line 6: Do not forget to add the closing curly bracket} to actually end the main function.

Many Statements

Most C programs contain many statements.

The statements are executed, one by one, in the same order as they are written:

Example

```
printf("Hello World!");
printf("Have a good day!");
return 0;
```

New Lines

To insert a new line, you can use the `\n` character:

Example

```
#include <stdio.h>

int main() {
  printf("Hello World!\n");
  printf("I am learning C.");
  return 0;
}
```

What is \n exactly?

The newline character (`\n`) is called an **escape sequence**, and it forces the cursor to change its position to the beginning of the next line on the screen. This results in a new line.

Examples of other valid escape sequences are:

Escape Sequence	**Description**
\t	Creates a horizontal tab
\\	Inserts a backslash character (\)
\"	Inserts a double quote character