1st ASSIGNMENT OF C LANGUAGE

**INTRODUCTION TO C LANGUAGE:**

* The C Language is developed by Dennis Ritchie for creating system applications that directly interact with the hardware devices such as drivers, kernels, etc.
* C programming is considered as the base for other programming languages, that is why it is known as mother language.
* C has now become a widely used professional language for various reasons.Some of these are:

1. Easy to learn

2. Structured language

3. It produces efficient programs.

4. It can handle low-level activities.

5. It can be compiled on a variety of computers.

**VARIABLES IN C LANGUAGE:**

* In programming, a variable is a container use to hold data.
* The value of the C variable may get change in the program.
* syntax to declare a variable:

data\_type variable;

example:

int a;

## Rules for defining variables:

1. A variable can have alphabets, digits, and underscore.
2. A variable name can start with the alphabet, and underscore only. It can't start with a digit.
3. No whitespace is allowed within the variable name.
4. A variable name must not be any reserved word or keyword, e.g. int, float, etc.

* **TYPES OF VARIABLES IN C:**

**1. LOCAL VARIABLE:**

A variable that is declared and used inside the function or block is called local variable.

example:

#include <stdio.h>

void function() {

   int x = 10; // local variable

}

int main()

{

  function();

return 0;

}

**2. GLOBAL VARIABLE:**

A variable that is declared outside the function or block is called a global variable.

example:

#include <stdio.h>

int x = 20;//global variable

void function1()

{

  printf("%d\n" , x);

}

int main() {

  function1();

    return 0;

}

**3. STATIC VARIABLE:**

A variable that retains its value between multiple function calls is known as static varia ble.   
It is declared with the static keyword.

Example:

#include <stdio.h>

void function(){

int x = 20;//local variable

static int y = 30;//static variable

x = x + 10;

y = y + 10;

printf("\n%d,%d",x,y);

}

int main() {

  function();

  function();

  function();

  return 0;

}