C++ 3_1

3.1 Variables
A variable in C++ is a name for a piece of memory that can be used to store information. There are many types of variables, which determines the size and layout of the variable's memory;

Variable Names

we can use any combination of letters and numbers for Variable and function names but it must start with a letter.
We can use Underscore (_) as a letter in variable name and can begin with an underscore But Identifiers beginning with an underscore are reserved, And identifiers beginning with an underscore followed by a lower case letter are reserved for file scope identifiers Therefore using underscore as starting letter is not desirable.

Akki and akki are different identifiers because upper and lower case letters are treated as different identifiers

Variable Types

There are many 'built-in' data types in C.

short int -128 to 127 (1 byte)

unsigned short int 0 to 255 (1 byte)

char 0 to 255 or -128 to +127 (1 byte)

unsigned char 0 to 255 (1 byte)

signed char -128 to 127 (1 byte)

int -32,768 to +32,767 (2 bytes)

unsigned int 0 to +65,535 (2 bytes)

long int -2,147,483,648 to +2,147,483,647 (4 bytes)

unsigned long int 0 to 4,294,967,295 (4 bytes)

float single precision floating point (4 bytes)

double double precision floating point (8 bytes)

long double extended precision floating point (10 bytes)

Definition, Declaration & Initialization

Definition is the place where variable is created (allocated storage).

Declaration is a place where nature (type) of variable is stated, but no storage is allocated.

Initialization means assigning a value to the variable.

Variables can be declared many times, but defined only once. Memory space is not allocated for a variable while declaration. It happens only on variable definition.

Variable declaration
syntax

```
data_type variable_name;
```

example

```
int a, b, c;
char flag;
```

Variable initialization
syntax

```
data_type variable_name = value;
```

example

```
int a = 50;
char flag = 't';
```

external and static
initialisation done once only.

auto and register
initialisation done each time block is entered.

external and static variables cannot be initialised with a value that is not known until run-time; the initialiser must be a constant expression.

A variable that has not been assigned a value is called an uninitialized variable. Uninitialized variables are very dangerous because they cause intermittent problems (due to having different values each time you run the program). This can make them very hard to debug.