Variables in C / Identifiers in C

- variable is the container for storing value.
- variable refers to the name given to a memory location
- variable need to be declared before using.

Declaration: Announcing properties ex, name & size

Format:

datatype variable-name

decides the size to be occupied in memory

Definition: allocating memory to a variable

Initialization: Simultaneous declaration & definition

Ex: int var = 3;

→ Rules for naming Variables

- can't be of void datatype
- composed of letters, numbers, etc, digits, etc
- doesn't start with digit
- not recommed to start with underscore
- case sensitive
- special characters not allowed. except underscore (_)
- Spaces in both are not allowed.
- don't use keywords as variables
- prefer using long names for var. (limit for 31 char)
- Exception: there can be same funct & variable name.

Keywords in C

- These are the reserved words which have special meaning in C language.
- There are 32 keywords in C language
- These can't be used as variables / identifiers

Identifiers in C

A user defined -

- Name of a variable, funcn, structure, union, array, etc. is called identifier
- The name is used to identify the given variable / entity on behalf of its name.
- Two or more entities should not have same name as identifier.
- All variables are identifiers but the inverse is not true.
- Once identifier is declared, it can be used anywhere in the program to refer to the associated value.

Naming convention is same as that of Variables.