C-07 → Constants in C

Constants:

Value won't change in the program.

Types of Constants:

* Numeric → Integer → Decimal → 0-9 (base 10)
  - Octal → 0-7 (base 8)
  - Hexadecimal → 0- (base 16)
  - Floating / Real Constants.

* Character → Single character Constants
  - String Constants.
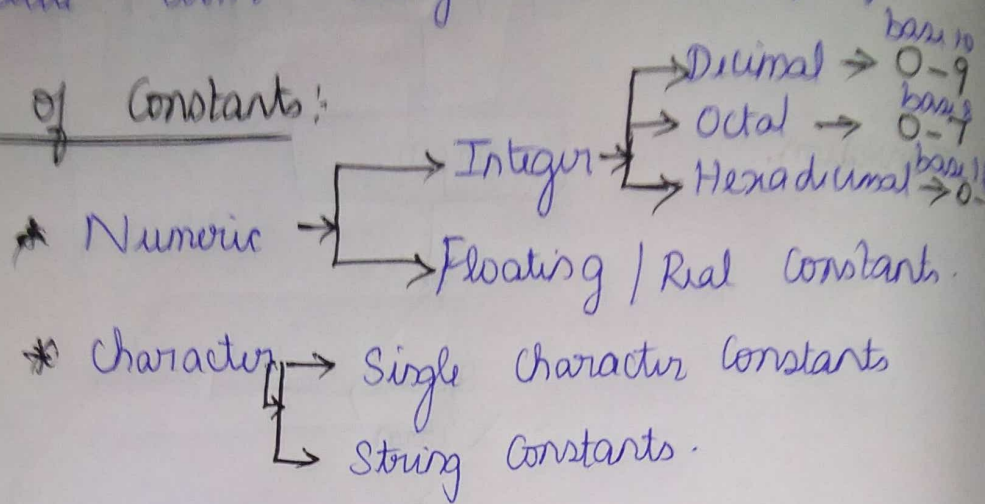
Decimal Constants:

* 0 - 9 ; base 10

  Eg:- 5, 1, 10, ...

Octal Constants:
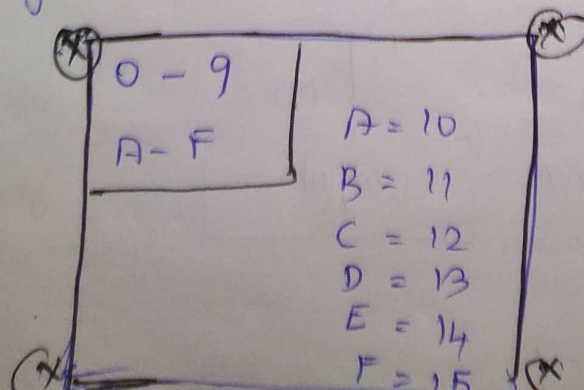
* 0-7 ; base 8

  Eg:- 05 (preceded by 0)

Hexadecimal Constants:

* 0-9 and A-F ; base 16.

  Eg:- 0X74 or 0x74 (preceded by 0X)

0 - 9	
A - F	A = 10
	B = 11
	C = 12
	D = 13
	E = 14
	F = 15

Eg: 0X7F ✓   0X7G ✗

9123 ✗   123 ✓

+1234 ✓   -1234 ✓

56;100 ✗

Floating / Real Constants:-

Eg: +12.56   -12.56 ✓
      ↓   ↓
    real fractional
    part   part   12.56.57 ✗

Single character Constant :-

Eg: 'a', 's', '%', ';', '@'.

* These character constants are stored in the computer in the form of ASCII Codes.

A   'a' = 97   ;   'A' = 65   ASCII Code for A starts from 65

      'b' = 95   'B' = 66.

*   5 ≠ '5'   A-Z = 65-90
                        a-z = 97-122
                        0-9 = 48-57
printf ("%d", 'a'); → 97   Special = 0-47, 58-64
printf ("%c", 97); → a   91-96

String constants:

"Jenny", "abc", "ab$", "12345

$$\boxed{"a" \neq 'a'}$$

↓   ↓

String   Character.

$$\boxed{"!" \neq '!'}$$

"Jenny !o"

↳ Length of string is 6

Const Keyword:

Eg: Void main ()
      {
          Const int a = 10;

          a = 50; → error (read only value)
      }

* Because we declared Const for value of a; so we cannot change value of a.

* If Const keyword not there; then we can change the value of 'a' variable a.

* We can also use `#define NAME VALUE`