**Literature**

**Integer literals**

It can be decimal, octal, hexadecimal constant.

0x or 0X for hexa decimal

0 for octal

Nothing for decimal

212        // legal decimal

0213      // octal

0x4b      // hexadecimal

30l         // long

30ul       // unsigned long

215u      // legal unsigned integer

0xFeeL  // legal

078        // illlegal octal digit

032UU   // illegal octal digit

**Floating-point literature**

It is the part of floating point, fractional point and exponent part

3.14159        // legal

31459E-5L   // legal

510E            // illegal

210F            // illegal

.e55             // illegal

**String literature in go**

“Hello, Dear”

“ Hello, \

dear”

“hello,”

**Const literature**

const var type = value;

const LENGTH = 10

const WIDTH = 5

**Go scope rules**

**Local variable**

**Globle variable**

**Firmal parameters**

**Local variable**

Inside th function is called as local variable

import "fmt"

/\* global variable declaration \*/

var g int

func main() {

   /\* local variable declaration \*/

   var a, b int

   /\* actual initialization \*/

   a = 10

   b = 20

   g = a + b

   fmt.Printf("value of a = %d, b = %d and g = %d\n", a, b, g)

}

value of a = 10, b = 20 and g = 30

**Globle variable**

But local variable inside the main has higher preference hence output will be 10 instead of 20

package main

import "fmt"

/\* global variable declaration \*/

var g int = 20

func main() {

   /\* local variable declaration \*/

   var g int = 10

   fmt.Printf ("value of g = %d\n",  g)

}

value of g = 10