



链滴

一种有趣的 golang "构造函数" 的写法

作者: [269812428](#)

原文链接: <https://ld246.com/article/1522120099804>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

```

package main

import (
    "fmt"
)

type options struct {
    a int64
    b string
    c map[int]string
}

func NewOption(opt ...ServerOption) *options {
    r := new(options)
    for _, o := range opt {
        o(r)
    }
    return r
}

type ServerOption func(*options)

func WriteA(s int64) ServerOption {
    return func(o *options) {
        o.a = s
    }
}

func WriteB(s string) ServerOption {
    return func(o *options) {
        o.b = s
    }
}

func WriteC(s map[int]string) ServerOption {
    return func(o *options) {
        o.c = s
    }
}

func main() {
    opt1 := WriteA(int64(1))
    opt2 := WriteB("test")
    opt3 := WriteC(make(map[int]string,0))

    op := NewOption(opt1, opt2, opt3)

    fmt.Println(op.a, op.b, op.c)
}

```

原文来自: <https://gocn.io/question/1420> > GoCN