

# GoLang学习笔记

关键字： GoLang 基础

记录目前学习和使用GoLang过程中的经验总结以及遇到的坑。

## 类型

### 变量

- 使用关键字 `var` 定义变量，自动初始化为零值。如果 提供初始化值，可 省略变量类型，由编译器自动推断。

```
var x int
var f float32 = 1.6
var s = "abc"
```

- 在 函数内部，可用 `:=` 方式定义变量。只能用于局部变量

### 常量

- 常量值必须是 编译期可确定 的数字、字符串、布尔值。不能是包含变量的表达式计算值和 Struct，可以是 `len`、`cap`、`unsafe.Sizeof` 等编译期可确定结果的函数返回值。

```
const (
    a = "abc"
    b = len(a)
    c = unsafe.Sizeof(b)
)
```

- 关键字 `iota` 定义常量组中从 0 开始按行计数的自增枚举值。

```
const (
    _ = iota // iota = 0
    KB int64 = 1 << (10 * iota) // iota = 1
    MB // 与 KB 表达式相同，但 iota = 2
    GB
    TB
)
```

- 自定义类型来实现枚举类型

```
type Color int
const (
    Black Color = iota
    Red
    Blue
)
```

## 基本类型

Type	Size	Default(零值)	Specification
bool	1	false	
byte	1	0	uint8
rune	4	0	Unicode, int32
int, uint	4 or 8	0	32 or 64 bit
int8, uint8	1	0	-128 ~ 127, 0 ~ 255
int, uint	4 or 8	0	32 or 64 bit
int16, uint16	2	0	-32768 ~ 32767, 0 ~ 65535
int32, uint32	4	0	-21亿 ~ 21 亿, 0 ~ 42 亿
int64, uint64	8	0	
float32	4	0.0	
float64	8	0.0	
complex64	8		
complex128	16		
uintptr	4 or 8		uint32 or uint64 pointer
array			复合类型，值类型
struct			复合类型，值类型
string		"" (Not nil)	UTF-8 字符串
slice		nil	引用类型，make创建
map		nil	引用类型，make创建
channel		nil	引用类型，make创建
interface		nil	接口
function		nil	函数

对于 复合类型 ， go语言会自动 递归 地将 每一个元素 初始化为其类型对应的 零值 。 比如：数组，结构体。

空指针值 `nil`，非 C/C++ `NULL`。

- 八进制、十六进制，以及科学记数法

```
a, b, c, d := 071, 0x1F, 1e9, math.MinInt16
```