

MARMOTA PROJECT - CHEATSHEET



IMPORTANDO PACOTES

```
import "fmt"
// Ou vários pacotes
import (
    "fmt"
    "math"
)
```

RODANDO O PROGRAMA

```
// Buildando
$ go build hello.go
$ ./hello
```

VARIÁVEIS

```
// tipo inferido
var num1 = 5
// tipo explicito
var num2 int = 6
// outra maneira
num3 := 7
num4 := num3
// declarar como string
var str string
// declarar como float
var rates float32 = 4.5
// declarar como booleano
var raining bool = false
// declarar multi variaveis
var num5, num6 int = 8, 9

var (
    age = 25
    name = "Samuel"
)
```

INTERPOLAÇÃO DE STRING

```
var num1 = 5
var rates float32 = 4.5

str := fmt.Sprintf(
    "num1 is %d and rates is %.2f", num1, rates)
```

CONVERSÃO DE TIPOS

```
num1 := 2
num2 := float32(num1)
num3 := uint(num1)

fmt.Println(num1) //2
fmt.Println(num2) //2
fmt.Println(num3) //2

num2 = 3.5
num4 := int(num2)
fmt.Println(num2) //3.5
fmt.Println(num4) // 3
sumStr := "123"
```

```
// converter string para int
import "strconv"
sum, err :=
    strconv.Atoi(sumStr)

// converter int para string
str := strconv.Itoa(sum)
```

TIPO ALIAS

```
type PeopleName string
var name PeopleName
name = "Jonathan"
```

ARRAYS

```
// array com 5 elementos
var nums [5] int
// [0 0 0 0 0]
nums[4] = 100
// [0 0 0 0 100]
fmt.Println("set:", nums[4])
// 100
```

SLICES

```
// cria um slice de 5
// elementos e capacidade = 5
x := make([] int, 5)
// [0 0 0 0 0]

// cria um slice de 2
// elementos e capacidade = 3
x = make([] int, 2, 3)
fmt.Println(x) // [0 0]

primes := [] int {2, 3, 5, 7}
```

SLICING

```
var c[3] string
c[0] = "iOS"
c[1] = "Android"
c[2] = "Windows"

fmt.Println(c[0:2])
//[iOS Android]
```

RANGING

```
primes := [] int {2, 3, 5, 7}

for i, v := range primes {
    fmt.Println(i, v)
}
```

LOOPING

```
for i:=0; i<5; i++ {
    fmt.Println(i)
}
// loop infinito
for {
}
// "while" loop
counter := 0
for counter <5 {
    fmt.Println(counter)
    counter++
}
// iterar através de um array
for i,n:= range c {
    fmt.Println(i,n)
}
```