

More On Go

"It is not the strongest of the species that survives, nor the most intelligent. It is the one that is the most adaptable to change." Charles Darwin

1st. the news

1. AU - Research - linear scaling (similar to blockchain) 8000 tx per sec, don't specify latency.
2. Binance, the world's largest cryptocurrency exchange by adjusted trading volume, has just made it easier for users to buy cryptocurrencies.
3. ETH will reach 1M Tx per sec.
4. California bill passed that defines things like "digital signature".
5. Land registry in Mexico is moving onto blockchain.

2nd. Purpose of a business

To make a profit for the owners of the business.

What is "Fiduciary Responsibility". It means that you have been placed / are in a position of legal responsibility for managing somebody else's money.

Go

1. Functions

Parts of a function

```
// Name will do a b c d.
func Name ( i1 Type1, i2 Type2 ) ( o1 Type1, o2 Type2 ) {
    // Comment in body
}
```

Observations 1. Name starts with a capital, therefore exported from package. 2. You can have more than 1 return value. 3. "error" is usually the last return value. 4. Comments on functions need to start with the name of the function.

1. Maps

Go has dictionary/maps

```

var m1 map[string]int
m1 = make(map[string]int)
m1["abc"] = 12
k := m1["abc"]
k2 := m1["xyz"]
k3, ok_t := m1["abc"]
k4, ok_f := m1["xyz"]

```

Observations 1. memory is not allocated to a map when it is declared. 2. You can just use `make` and `:=` to declare a map. 3. You can test to see if you have an un-allocated map by comparing to `nil`. 4. You can find out if a value is in a map.

1. Slices (Arrays)

An Array

```
var a1 [4]int
```

A slice

```
var s1 []int
```

What is a slice?

Allocating memory to a slice. Slices start out as "empty" or `nil`.

```

s1 = make ( []int, 5 )
s1 = make ( []int, 3, 6 )

```

Slice of slice:

```
s1 = s[1:2]
```

All of a slice or an array (how to convert an array to a slice)

```
s2 := s1[:]
```

Pitfalls!

1. Strings

Strings are immutable! How to denote a string.

1. Maps

A map is `var Name map[HashKeyType]ElementType`

Declare:

```
var Hw map[string]int
func init() {
    Hw = make( map[string]int )
}
```

or

```
Hw := make( map[string]int )
```

```
Hw["I80"] = 1421
Hw["US287"] = 841
```

Pull out the value and if a value is set.

```
vv := Hw["aaa"]
ww := Hw["I80"]
mm, found := Hw["I80"]
_, found2 := Hw["I90"]
```

1. io & cli

Output:

```
fmt.Printf ( "Format %s\n", "string" )
```

CLI

```
fmt.Printf ( "%s\n", os.Args[] )
```