Lecture 02 - What is a BlockChain - more on Go.

Question: What is a hash?

1. (math) a mapping form a range to a domain.

2. Text to a number.

```
abc                275f20377d6574b67399702947cb56849d2e02f7112c1d021603346c345b37f8
abd                3212601953780d3a8de118531b87bf6183edb8c81baf6982fdca260033a5f29b
war-and-peace.txt 67c570b0e09d70225d739aec9a7ea91631df1ea06ba44f0c9d9fe99e45f41756
```

3. Different kinds of hash, MD5, SHA1, SHA256, SHA3, SHA512, Keccak256.

Economics of Blockchain

This is an up/down year in the cryptocurrency world. China, India and a few other countries have banned it. On the flip side the US and about $^1/_2$ of the world are looking to replace national currencies with it.

In 2009 - bitcoin invented.

In 2013 - smart contracts - Ethereum.

In 2021 - 6% of transactions in 6 African countries on it. China and Russia are moving to replace national currency with it. Goldman Sacks / Apple are using it. Visa expects $1T in transactions on it.

Ability to create trust between non-trusting parties.

Ability to create economic systems.

Merick - shipping 40% decrease in shipping times. World wide $380 Billion in trade. 90% of all the goods in the world are moved by ship. 38 days average shipping time. A 40% reduction is dropping that to 23 days. Difference is 15 days. 40% of 380 billion is 152 billion in capital that is not tied up - at an average capital cost of 10% = 15.2 billion - over the 23 days. $^{15}/_{365}$ * 15.2 billion. -- About 0.62 billion.

Estonia - All titles and property on the chain. In US 6.20 Million Houses. Average title search $3821. Title search in Estonia, $22.14 each. Title insurance $1408. My calculation $21 billion a year.

Marshal Islands - off of the dollar and onto a blockchain. The estimate is that the government will save around $5M a year or about $90 per resident a year. For entire US 327 million - that is $29 billion dollars.

8 African Countries, representing 1/5th of the world population use some form of crypto-currency as an alternative to the national currency.

Over 22 accredited educational educational institutions now issuing certificates based on blockchain.

Perspective on 28 or 29 billion - free college tuition for all students in the United States is estimated to cost 75 billion.

State of Nevada - Marriage license on blockchain.

The largest single “blockchain” sale is valued at 103 million. In a commercial property in Zurich.

El Salvidor - has made Bitcoin a national currency. Thy just purchased 410 of them.

At the same time, China and India have both banned crypto-currencies! So 2/5ths of the world have made it illegal. Kasascstan has banned crypto-mining because it uses too much electricity.

Go - Intro

Assignment 1 - Due Mon Feb 5 -- Continuing from last time.

Echo - walk through

```
 1: package main
 2:
 3: import (
 4:     "fmt"
 5:     "os"
 6: )
 7:
 8: func main() {
 9:     for i, s := range os.Args {
10:         if i == 0 {
11:         } else if i == len(os.Args)-1 {
12:             fmt.Printf("%s\n", s)
13:         } else {
14:             fmt.Printf("%s ", s)
15:         }
16:     }
17: }
```

1st time - or when you change dependencies

```
$ mkdir echo
$ cd echo
$ vi main.go
$ go mod init
$ go mod tidy
$ go build
```

After that

```
$ go build
```

Marshal and Unmarshal of data - walk through

```
 1: package main
 2:
 3: import (
 4:     "encoding/json"
 5:     "fmt"
 6: )
```

```go

type Demo struct {
    Aa int
    Ab string
}

func main() {
    d := Demo{
        Aa: 33,
        Ab: "Penguines are People Too...",
    }

    buf, err := json.Marshal(d)
    if err != nil {
        fmt.Printf("Error: %s\n", err)
    }

    fmt.Printf("%s\n", buf)

    buf, err = json.MarshalIndent(d, "", "\t")
    if err != nil {
        fmt.Printf("Error: %s\n", err)
    }

    fmt.Printf("%s\n", buf)
}
```

```go
package main

import (
    "encoding/json"
    "fmt"
)

type Demo struct {
    Aa int `json:"A_cx"`
    Ab string
}

func main() {
    s := `{
        "A_cx": 33,
        "Ab": "Penguines are People Too...",
        "Ac": "skips this, no error"
    }`

    var d Demo
    err := json.Unmarshal([]byte(s), &d)
    if err != nil {
        fmt.Printf("Error: %s\n", err)
    }

    fmt.Printf("%+v\n", d)
}
```

For loops

```go
package main

```

```
import "fmt"

var aSlice = []string{"abc", "def", "ghi"}
var aMap = map[string]int{
    "alice": 22,
    "bob":   23,
    "tom":   25,
}

func main() {
    for i := 0; i < 5; i++ {
        fmt.Printf("Loop 1: %d\n", i)
    }
    fmt.Printf("\n")

    for i, v := range aSlice {
        fmt.Printf("Loop 2: at:%d ->%s<-\n", i, v)
    }
    fmt.Printf("\n")

    for key, val := range aMap {
        fmt.Printf("Loop 3: key:%s ->%v<-\n", key, val)
    }
}
```

Functions

```
package main

import "fmt"

func Qs(ss []string) (rv []string) {

    partition := func(arr []string, low, high int) ([]string, int) {
        pivot := arr[high]
        i := low
        for j := low; j < high; j++ {
            if arr[j] < pivot {
                arr[i], arr[j] = arr[j], arr[i]
                i++
            }
        }
        arr[i], arr[high] = arr[high], arr[i]
        return arr, i
    }

    var quickSort func(arr []string, low, high int) []string
    quickSort = func(arr []string, low, high int) []string {
        if low < high {
            var p int
            arr, p = partition(arr, low, high)
            arr = quickSort(arr, low, p-1)
            arr = quickSort(arr, p+1, high)
        }
        return arr
    }

    rv = quickSort(ss, 0, len(ss)-1)
    return
}

func main() {
```

```
    r := Qs([]string{"def", "ghi", "abc", "ddd", "zzz"})
    fmt.Printf("%v\n", r)
}
```

```
package main_test

import (
    "reflect"
    "testing"

    main "github.com/Univ-Wyo-Education/S22-4010/class/lect/02/funcDemo"
)

func Test_Qs(t *testing.T) {
    expect := []string{"abc", "ddd", "def", "ghi", "zzz"}
    data := []string{"def", "ghi", "abc", "ddd", "zzz"}

    rv := main.Qs(data)
    if len(rv) != len(expect) {
        t.Errorf("Expected %v got %v\n", expect, rv)
    }
    if !reflect.DeepEqual(rv, expect) {
        t.Errorf("Expected %v got %v\n", expect, rv)
    }

}
```

Go Generics Faster

https://dominictobias.medium.com/go-is-about-to-get-a-whole-lot-faster-a50c1e7d60b9

```
package g_lib

import (
        "constraints"
)

func Min[T constraints.Ordered](a, b T) T {
        if a < b {
                return a
        }
        return b
}
```