Summary

Go's `iota` identifier is used in `const` declarations to simplify definitions of incrementing numbers. Because it can be used in expressions, it provides a generality beyond that of simple enumerations.

The value of iota is reset to 0 whenever the reserved word `const` appears in the source (i.e. each const block) and incremented by one after each ConstSpec e.g. each Line. This can be combined with the constant shorthand (leaving out everything after the constant name) to very concisely define related constants.

Iota: https://go.dev/ref/spec#Iota

Constant declarations: https://go.dev/ref/spec#Constant_declarations

Examples

The official spec has two great examples:

https://go.dev/ref/spec#Iota

Here's one from Effective Go:

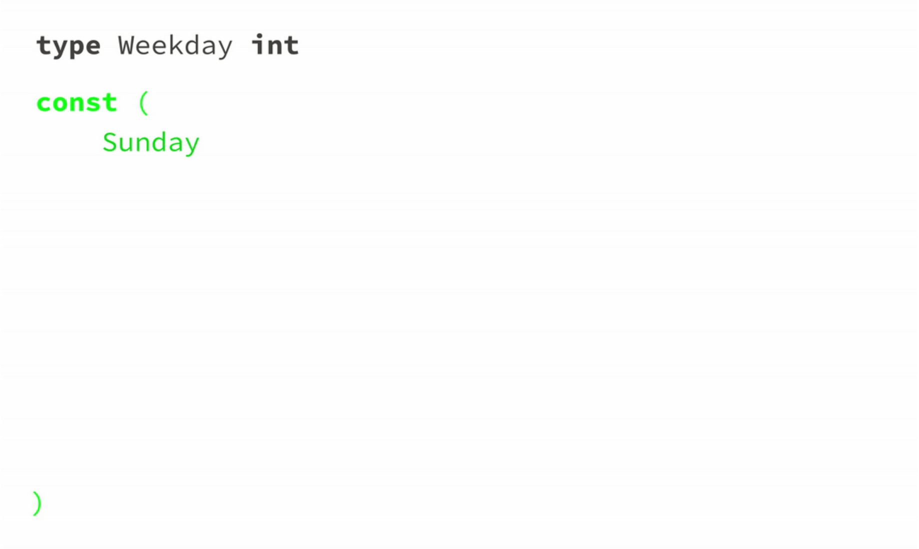
```
type ByteSize float64

const (
    _           = iota // ignore first value by assigning to blank identifier
    KB ByteSize = 1 << (10 * iota)
    MB
    GB
    TB
    PB
    EB
    ZB
    YB
)
```

Weekday enum example - How iota is calculated - From Learn Go Programming Blog:

Articles

- Ultimate Visual Guide to Go Enums and Iota *2017-10-09*



```go
type Weekday int

const (
    Sunday
)
```

Figure 1: How iota works