

Estructuras de control de flujo en Go (Golang)

Estructuras condicionales

If básico

```
package main

import "fmt"

func main() {
    encendido := true

    if encendido {
        fmt.Println("Está encendido")
    }
}
```

Notarán que es muy parecido a la sintaxis de C o a la mayoría de los lenguajes inspirados en este, la principal diferencia es que los paréntesis alrededor de la condición no son obligatorios, pero de igual forma comprueba si dicha condición es verdadera y en caso de que así sea, realiza las instrucciones que se encuentran dentro de sus respectivas llaves, en caso contrario las instrucciones son ignoradas, nada raro hasta el momento. Bastante útil, pero ¿Que sucede si necesito que se ejecute una instrucción en caso de que a condición resulte falsa?

If-else

```
package main

import "fmt"

func main() {
    encendido := false

    if encendido {
        fmt.Println("Está encendido")
    } else {
        fmt.Println("Está apagado")
    }
}
```

Perfecto, la confiable instrucción else al rescate. En este caso la condición sea evaluada como falsa, las instrucciones dentro del if son ignoradas y se ejecutan las del else.

Otro caso frecuente es necesitar comprobar más de una condición, fácilmente podrías anidar condicionales (colocar otro if dentro del bloque del else, por ejemplo), pero dependiendo del caso podría ser más adecuado simplemente usar la estructura else-if, aunque será necesario recurrir a un ejemplo diferente al anterior.

Else-if

```
package main

import "fmt"

func main() {
    var edad uint8 = 180

    if edad > 150 {
        fmt.Println("¿Eres inmortal?")
    } else if edad >= 18 {
        fmt.Println("Eres mayor de edad")
    } else {
        fmt.Println("Eres menor de edad")
    }
}
```

Hasta el momento nada fuera de lo común, me parece, ahora veamos algo más propio de Go.

If con declaración corta

```
package main

import "fmt"

func main() {
    if edad := -5; edad > 150 {
        fmt.Println("¿Eres inmortal?")
    } else if edad >= 18 {
        fmt.Println("Eres mayor de edad")
    } else if edad < 18 && edad > 0 {
        fmt.Println("Eres menor de edad")
    } else {
        fmt.Println("Edad fuera del rango")
    }
}
```

Así es, `if` permite ejecutar una declaración corta antes de comenzar a evaluar las condiciones, es importante resaltar que el ámbito de esa variable está limitado a la propia estructura condicional, intentar acceder a ella desde fuera generará un error al momento de compilar.