

Tipos de Datos en Go

1. Números - Enteros con signo

int -> Depende de la arquitectura (32 o 64 bits)

int8 -> 8 bits (-128 a 127)

int16 -> 16 bits (-32,768 a 32,767)

int32 -> 32 bits (-2,147,483,648 a 2,147,483,647)

int64 -> 64 bits (+/-9.2210)

2. Números - Enteros sin signo

uint -> Depende de la arquitectura

uint8 -> 8 bits (0 a 255)

uint16 -> 16 bits (0 a 65,535)

uint32 -> 32 bits (0 a 4.2910)

uint64 -> 64 bits (0 a 1.8410)

uintptr -> Tamao suficiente para contener direcciones de punteros

Alias:

byte -> alias de uint8

rune -> alias de int32 (para caracteres Unicode)

3. Números decimales (punto flotante)

float32 -> 32 bits (~ 6-7 dgitos de precisin)

float64 -> 64 bits (~ 15-16 dgitos de precisin)

4. Números complejos

complex64 -> Parte real/imag. como float32

complex128 -> Parte real/imag. como float64

Tipos de Datos en Go

5. Texto

string -> Cadena de texto (UTF-8)

6. Booleanos

bool -> true o false

7. Tipos derivados y compuestos

[n]T -> Arreglo de tamaño fijo

[]T -> Slice (arreglo dinámico)

map[K]V -> Mapa (diccionario)

chan T -> Canal (goroutines)

*T -> Puntero

struct -> Agrupación de campos

interface -> Define comportamiento

8. Otros

error -> Interfaz incorporada para manejar errores

nil -> Valor cero para referencias, punteros, etc.

Ejemplo de uso

```
var a int = 42
```

Tipos de Datos en Go

```
var b float64 = 3.14
```

```
var c bool = true
```

```
var d string = "Hola"
```

```
var e byte = 'H'
```

```
var f rune = "
```

```
var g complex128 = 1 + 2i
```