

Go Data Types - 1

- A **type** determines a set of values together with operations and methods specific to those values.
- There are **predeclared types**, **introduced types** with **type declarations** and **composite types**: array, slice, map, struct, pointer, function, interface, and channel types.

Predeclared, Built-in Types

• Numeric types

- **int8, int16, int32, int64**
- **uint8, uint16, uint32, uint64**: used to represent unsigned (positive) integers.
- **uint** is an alias for **uint32** or **uint64** based on platform.
- **int** is an alias for **int32** or **int64** based on platform.
- **float32, float64**: zero before the decimal point separator can be omitted (**-0.5 -3. -0.1.4**).
- **complex64, complex128**.
- **byte** (alias for **uint8**).
- **rune** (alias for **int32**).

Go Data Types - 2

- **Bool type**

- pre-defined constants `true` and `false`.

- **String type**

- **Unicode** chars written enclosed by double-quotes.
- A string value is a (possibly empty) sequence of bytes.

- **Array and Slice Type**

- An array is a numbered sequence of elements of a single type, called the element type.
- An array has a fixed length (we specify how many items are in the array when we declare it), but a slice has a dynamic length (it can shrink or grow).

- **Map Type**

- A map is an unordered group of elements of one type, indexed by a set of unique keys of another type.
- A map in Go is similar to dictionary in Python

Go Data Types - 2

- **Struct Type (User defined type)**

- A struct is a sequence of **named elements, called fields**, each of which has a name and a type.
- a structure can be compared to class concept in Object Oriented Programming.

```
type Car struct {  
    brand string  
    price int  
}
```

Go Data Types - 2

- **Pointer Type**
 - A pointer is a variable that **stores the memory address of another variable.**
 - The value of an uninitialized pointer is **nil.**
- **Function and Interface Type**
- **Channel Type**
 - A channel provides a mechanism for concurrently executing functions to communicate by sending and receiving values of a specified element type.