Basic Types

This lesson list of inbuilt variable types in Go.

WE'LL COVER THE FOLLOWING

List of Built In Types

The following are the built in types in Go:

Common Types

`bool`	true or false
`string`	an array of characters

Numeric types

`uint`	either 32 or 64 bits.
`int`	same size as uint.
`uintptr`	an unsigned integer large enough to store the uninterpreted bits of a

	pointer value
`uint8`	the set of all unsigned 8-bit integers (0 to 255)
`uint16`	the set of all unsigned 16-bit integers (0 to 65535)
`uint32`	the set of all unsigned 32-bit integers (0 to 4294967295)
`uint64`	the set of all unsigned 64-bit integers (0 to 18446744073709551615)
`int8`	the set of all signed 8-bit integers (-128 to 127)
`int16`	the set of all signed 16-bit integers (-32768 to 32767)
`int32`	the set of all signed 32-bit integers (-2147483648 to 2147483647)
`int64`	the set of all signed 64-bit integers (-9223372036854775808 to 9223372036854775807)
`float32`	the set of all IEEE-754 32-bit floating-point numbers
`float64`	the set of all IEEE-754 64-bit floating-point numbers
`complex64`	the set of all complex numbers with `float32` real and imaginary parts

`complex128`	the set of all complex numbers with `float64` real and imaginary parts
`byte`	alias for `uint8`
`rune`	alias for `int32` (represents a Unicode code point)

Example

Given below are example declarations of variables of some of the built-in types:

Environment Variables

Key:	Value:
GOPATH	/go

```
package main

import (
	"fmt"
	"math/cmplx"
)

var (
	goIsFun bool       = true //declaring a variable of type bool
	maxInt  uint64     = 1<<64 - 1 //declaring a variable of type uint64
	complex complex128 = cmplx.Sqrt(-5 + 12i) //declaring a variable of type complex128
)

func main() {
	const f = "%T(%v)\n"
	fmt.Printf(f, goIsFun, goIsFun)
	fmt.Printf(f, maxInt, maxInt)
	fmt.Printf(f, complex, complex)
}
```