[](https://ida.interchain.io/)

[Interchain Developer Academy](https://ida.interchain.io/)/[Interchain Developer Academy](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)



Search

[Interchain Developer Academy](https://ida.interchain.io/)[Interchain Developer Academy](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

Search



Filters

Interchain Developer Academy

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 0 - Getting Started](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Getting Started](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Blockchain 101](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Blockchain History](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Public and Managed Blockchains](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Consensus in Distributed Networks](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Cryptography](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Self-Assessment Quiz](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Go Introduction - First Steps](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Go Basics](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Go Interfaces](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Control Structures in Go](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Arrays and Slices in Go](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Standard Packages in Go](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Concurrency in Go](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Good-To-Know Dev Terms](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Docker Introduction](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 1 - Introduction to the Interchain](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Introduction to the Interchain](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Blockchain Technology and the Interchain](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[The Interchain Ecosystem](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Getting ATOM and Staking It](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[A Blockchain App Architecture](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Accounts](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Transactions](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Messages](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Modules](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Protobuf](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Multistore and Keepers](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[BaseApp](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Queries](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Events](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Context](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Testing](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Relaying with IBC](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Interchain Security](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Bridges](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Migrations](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 1 Quiz](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 2 - First Steps](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[First Steps](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Setup Your Work Environment](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Run a Node, API, and CLI](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Ignite CLI](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Exercise - Make a Checkers Blockchain](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Store Object](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create Custom Messages](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create and Save a Game Properly](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Add a Way to Make a Move](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Emit Game Information](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Record the Game Winner](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 2 Exercise](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 3 - Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[What is IBC?](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC/TAO - Connections (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC/TAO - Channels (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC/TAO - Clients (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC Token Transfer](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Interchain Accounts (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create a Custom IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Integrate IBC Middleware Into a Chain (OPTIONAL)](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC Tooling](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[What is CosmJS?](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Your First CosmJS Actions](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Compose Complex Transactions](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Learn to Integrate Keplr](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create Custom CosmJS Interfaces](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 4 - Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Keep an Up-To-Date Game Deadline](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Keep Track Of How Many Moves Have Been Played](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Put Your Games in Order](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Auto-Expiring Games](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Let Players Set a Wager](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Handle wager payments](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Integration tests](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Incentivize Players](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Help Find a Correct Move](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Play With Cross-Chain Tokens](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Understand IBC Denoms](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Go Relayer](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Hermes Relayer](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 5 - CosmJS Advanced](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[CosmJS Advanced](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create Custom Objects](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create Custom Messages](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Get an External GUI](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Integrate CosmJS and Keplr](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Backend Script for Game Indexing](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 6 - IBC Deep Dive](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC Deep Dive](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[IBC Application Developer Introduction](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Make a Module IBC-Enabled](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Adding Packet and Acknowledgment Data](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Extend the Checkers Game With a Leaderboard](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Create a Leaderboard Chain](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Week 7 - From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Run in Production](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Prepare the Software to Run](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Prepare a Validator and Keys](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Prepare Where the Node Starts](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Prepare and Connect to Other Nodes](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Configure, Run, and Set Up a Service](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Prepare and Do Migrations](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Simulate Production in Docker](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Tally Player Info After Production](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Add a Leaderboard as a Module](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Migrate the Leaderboard Module After Production](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Simulate a Migration in Docker](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Final Exam](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[What's Next?](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

[Continue Your Interchain Journey](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html)

Docs Version Switcher

On this page

[Simple unit test](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html#simple-unit-test)

[#Copy link](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html#go-interfaces) **Go Interfaces**

It is time to look at interfaces in Go. After that, you will briefly examine testing in Go.

Go offers the so-called "interface type". This is a collection of method signatures. An interface value can hold any value that implements those methods. Try it:



Copy

package main

import (

"fmt"

"math"

)

type Vector3D struct {

x, y, z float64

}

type Vector2D struct {

x, y float64

}

type Number float64

type Euclid interface {

Norm() float64

}

func (v Vector3D) Norm() float64 {

return math.Sqrt(v.x\*v.x + v.y\*v.y + v.z\*v.z)

}

func (v Vector2D) Norm() float64 {

return math.Sqrt(v.x\*v.x + v.y + v.y)

}

func (n Number) Norm() float64 {

if n > 0 {

return float64(n)

}

return -float64(n)

}

func main() {

var v Euclid

v = Vector3D{1, 2, 3}

fmt.Println(v.Norm())

v = Vector2D{1, 2}

fmt.Println(v.Norm())

v = Number(-2.5)

fmt.Println(v.Norm())

}



[Test it online (opens new window)↗](https://go.dev/play/p/lOEgFlWL2JI).

You see the declaration of the three types and methods as before. You have also declared an additional interface, Euclid, which includes a method signature Norm() float64. Since all defined types implement the Norm method, You can now use the Euclid interface to hold the instances of those types.

There is a special empty interface: interface{}. Because it has no method signatures, it is implemented by all types and can be used to hold values of any type:



Copy

package main

import "fmt"

func main() {

var i interface{}

i = 2

fmt.Println(i)

i = "Test"

fmt.Println(i)

s, ok := i.(string)

fmt.Println(s, ok)

}



[Test it online (opens new window)↗](https://go.dev/play/p/p9bTIVlCPfr).

The syntax for direct access to the underlying value of the interface value is i.(T). This is useful for *type switches*. In the next module, you will learn the control constructs.

[#Copy link](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html#simple-unit-test) Simple unit test

Go offers the testing package testing and a tool called go test. These are very helpful.

To explore the basics, first, write a function sum. This is the function you will test:



Copy

package sumutil

func Sum(a,b int) (s int) {

for i:=a; i<=b; i++ {

s+=i

}

return

}

You should be able to see what this does and know that it probably works. Even so, you should test it.

Save the previous program as sum.go in a folder sumutil. Then make another file with the following:



Copy

package sumutil

import "testing"

func TestSum(t \*testing.T) {

cases := []struct {

start, end, result int

}{

{ 5, 6, 11 },

{ 1, 9, 45 }, // (1+9)+(2+8)+(3+7)+(4+6)+5=45

{ 0, 9, 45 },

{ 1, 3, 6 },

{ 10, 19, 145 }, // 10+(10+1)+(10+2)+...+(10+9)=

// 10\*10+sum(1,9)=145

}

for \_, c := range cases {

if got := Sum(c.start, c.end); got != c.result {

t.Errorf("Sum(%v,%v) got %v, want %v", c.start, c.end,got, c.result)

}

}

}

func BenchmarkSum(b \*testing.B) {

for i:=0; i<b.N; i++ {

Sum(0,i)

}

}

Save this file as sum\_test.go. Now run go test.

You will see that it passes the test.



A test function has the syntax TestXXX.   
  
A benchmark function has the syntax BenchXXX.

Use go test -help to see what you need to run benchmarks.



This video provides a quick demonstration of working with interfaces in Golang.



**Further readings**

* [Testing package (opens new window)↗](https://golang.org/pkg/testing/)
* [Writing unit tests (opens new window)↗](https://blog.alexellis.io/golang-writing-unit-tests/)

synopsis

To summarize, this section has explored:

* How the "interface type" offered by Go is a collection of method signatures, and an interface value is used to implement those methods.
* How the Euclid interface can hold instances of all defined types implementing the Norm method.
* How the special empty interface interface{} can be implemented by all types and hold values of any type because it has no method signatures.
* How to use Go's testing package testing and the go test tool.

previous

[](https://ida.interchain.io/tutorials/4-golang-intro/2-basics.html)

**[Go Basics](https://ida.interchain.io/tutorials/4-golang-intro/2-basics.html)**

up next

**[Control Structures in Go](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html)**

[[](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html)](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html)

Rate this Page

icon smile

icon meh

icon frown

Would you like to add a message?

Submit

Thank you for your Feedback!

[](https://ida.interchain.io/ida-course/discord-info.html)

On this page

[Simple unit test](https://ida.interchain.io/tutorials/4-golang-intro/3-interfaces.html#simple-unit-test)

#### **Get Cosmos updates**

Unsubscribe at any time. [Privacy Policy↗](https://v1.cosmos.network/privacy)

     Next

Documentation

[Cosmos SDK](https://docs.cosmos.network/)[Cosmos Hub](https://hub.cosmos.network/)[CometBFT](https://docs.cometbft.com/)[IBC Protocol](https://ibc.cosmos.network/)

Community

[Interchain blog](https://blog.cosmos.network/)[Forum](https://forum.cosmos.network/)[Discord](https://discord.gg/cosmosnetwork)

Contributing

[Source code on GitHub](https://github.com/cosmos/sdk-tutorials)

[](https://ida.interchain.io/)

[Interchain Developer Academy](https://ida.interchain.io/)

**[](https://blog.cosmos.network/)[](https://twitter.com/cosmos)[](https://discord.gg/cosmosnetwork)[](https://www.linkedin.com/company/interchain-foundation/about/)[](https://reddit.com/r/cosmosnetwork)[](https://t.me/cosmosproject)[](https://www.youtube.com/c/CosmosProject)**



Dark mode

† This website is maintained by the Interchain Foundation (ICF). The contents and opinions of this website are those of the ICF. The ICF provides links to cryptocurrency exchanges as a service to the public. The ICF does not warrant that the information provided by these websites is correct, complete, and up-to-date. The ICF is not responsible for their content and expressly rejects any liability for damages of any kind resulting from the use, reference to, or reliance on any information contained within these websites.

Cosmos is a registered trademark of the [Interchain Foundation.](https://interchain.io/)[Privacy](https://v1.cosmos.network/privacy)