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

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



Search

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

Search



Filters

Interchain Developer Academy

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Docs Version Switcher

On this page

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

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

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

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

It is time to discuss control structures in Go. You will start with the if statement.

[#Copy link](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html#if-statement) if statement

The syntax is as follows:



Copy

if boolean\_expression\_1 {

// execute, if boolean\_expression\_1 is true

} else if boolean\_expression\_2 {

// execute, if boolean\_expression\_1 is false but boolean\_expression\_2

// is true

// } else if ...

} else if boolean\_expression\_n {

// execute, if boolean\_expression\_1, ... , boolean\_expression\_(n-1) are

// false, but boolean\_expression\_n is true

} else {

// execute, if boolean\_expression\_1, ... , boolean\_expression\_n are false

}

You need brackets {} for each if statement. The { bracket needs to be on the same line as the if statement.

If you want to use an else if or else statement, note that they have to be on the same line as the } bracket of the previous block.

You can also give if or else if a short statement to execute before the condition:



Copy

if s := 10%2; s==0 {

fmt.Println(s)

}

Boolean expressions evaluate for true or false. The comparison operators are:



Copy

== equal

!= not equal

< less

<= less or equal

> greater

>= greater or equal

The if statement is also used for error handling. You will often see code like:



Copy

if err != nil {

fmt.Printf(err)

}

[#Copy link](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html#switch-statement) switch statement

The syntax of a switch statement is as follows:



Copy

switch expression {

case value\_1:

// do something if expression is equal to value\_1

case value\_2:

// do something if expression is equal to value\_2

// ...

case value\_n:

// do something if expression is equal to value\_n

default:

// do something if value\_1 ... value\_n does not match expression

}

Cases are evaluated from top to bottom. The switch finishes if a case succeeds.



values\_x must have the same type as expression.

If you do not give an expression to switch, then Go interprets it as switch true. This provides another way to write if-else if-else chains:



Copy

switch {

case boolean\_expression\_1:

// if true is equal to boolean\_expression\_1

// which is the same as if boolean\_expression\_1 is true.

case boolean\_expression\_2:

// else if

// ...

case boolean\_expression\_n:

// else if

default:

// else

}



Unlike other languages (for example **C**), you do not need to break to avoid fallthrough. In Go, you need to fallthrough explicitly. fallthrough will enter the next case, even if the expression does not match.

There are so-called type switches in Go:



Copy

var i interface{}

switch i.(type) {

case bool:

// if i has type bool

fmt.Printf("Value of i is %v\n", i.(bool))

case int32, in64:

// ...

// ...

default:

// i has another type

}



You can use multiple value cases, like case 1, 2, 4, 9, 16:.

[#Copy link](https://ida.interchain.io/tutorials/4-golang-intro/4-control.html#for-statement) for statement

for is the only looping statement in Go. The syntax is as follows:



Copy

for init\_statement; condition\_expression; post\_statement {

// do something

}

Before the iterations start, it will first execute the init\_statement. The loop body will be executed as long as condition\_expression is true. The post\_statement will be executed at the end of every iteration.



init\_statement and post\_statement are optional. Without them, the for statement is like the while statement in other languages.

You can use continue to skip the iteration or break to terminate the execution, like in C, C#, Java, etc.



**Further reading:**

* [Errors are values (opens new window)↗](https://blog.golang.org/errors-are-values)

synopsis

To summarize, this section has explored:

* Control structures in Go, and the syntax used by the if statement, the switch statement, and the for statement.
* How unlike other languages Go does not need to break to avoid fallthrough, instead you need to provide the fallthrough instruction explicitly.

previous

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

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

up next

**[Arrays and Slices in Go](https://ida.interchain.io/tutorials/4-golang-intro/5-arrays.html)**

[[](https://ida.interchain.io/tutorials/4-golang-intro/5-arrays.html)](https://ida.interchain.io/tutorials/4-golang-intro/5-arrays.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

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

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

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

#### **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)