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

[Interchain Developer Academy](https://ida.interchain.io/)/[Interchain Developer Academy](https://ida.interchain.io/tutorials/5-docker-intro/)



Search

[Interchain Developer Academy](https://ida.interchain.io/)[Interchain Developer Academy](https://ida.interchain.io/tutorials/5-docker-intro/)

Search



Filters

Interchain Developer Academy

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 0 - Getting Started](https://ida.interchain.io/tutorials/5-docker-intro/)

[Getting Started](https://ida.interchain.io/tutorials/5-docker-intro/)

[Blockchain 101](https://ida.interchain.io/tutorials/5-docker-intro/)

[Blockchain History](https://ida.interchain.io/tutorials/5-docker-intro/)

[Public and Managed Blockchains](https://ida.interchain.io/tutorials/5-docker-intro/)

[Consensus in Distributed Networks](https://ida.interchain.io/tutorials/5-docker-intro/)

[Cryptography](https://ida.interchain.io/tutorials/5-docker-intro/)

[Self-Assessment Quiz](https://ida.interchain.io/tutorials/5-docker-intro/)

[Go Introduction - First Steps](https://ida.interchain.io/tutorials/5-docker-intro/)

[Go Basics](https://ida.interchain.io/tutorials/5-docker-intro/)

[Go Interfaces](https://ida.interchain.io/tutorials/5-docker-intro/)

[Control Structures in Go](https://ida.interchain.io/tutorials/5-docker-intro/)

[Arrays and Slices in Go](https://ida.interchain.io/tutorials/5-docker-intro/)

[Standard Packages in Go](https://ida.interchain.io/tutorials/5-docker-intro/)

[Concurrency in Go](https://ida.interchain.io/tutorials/5-docker-intro/)

[Good-To-Know Dev Terms](https://ida.interchain.io/tutorials/5-docker-intro/)

[Docker Introduction](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 1 - Introduction to the Interchain](https://ida.interchain.io/tutorials/5-docker-intro/)

[Introduction to the Interchain](https://ida.interchain.io/tutorials/5-docker-intro/)

[Blockchain Technology and the Interchain](https://ida.interchain.io/tutorials/5-docker-intro/)

[The Interchain Ecosystem](https://ida.interchain.io/tutorials/5-docker-intro/)

[Getting ATOM and Staking It](https://ida.interchain.io/tutorials/5-docker-intro/)

[A Blockchain App Architecture](https://ida.interchain.io/tutorials/5-docker-intro/)

[Accounts](https://ida.interchain.io/tutorials/5-docker-intro/)

[Transactions](https://ida.interchain.io/tutorials/5-docker-intro/)

[Messages](https://ida.interchain.io/tutorials/5-docker-intro/)

[Modules](https://ida.interchain.io/tutorials/5-docker-intro/)

[Protobuf](https://ida.interchain.io/tutorials/5-docker-intro/)

[Multistore and Keepers](https://ida.interchain.io/tutorials/5-docker-intro/)

[BaseApp](https://ida.interchain.io/tutorials/5-docker-intro/)

[Queries](https://ida.interchain.io/tutorials/5-docker-intro/)

[Events](https://ida.interchain.io/tutorials/5-docker-intro/)

[Context](https://ida.interchain.io/tutorials/5-docker-intro/)

[Testing](https://ida.interchain.io/tutorials/5-docker-intro/)

[Relaying with IBC](https://ida.interchain.io/tutorials/5-docker-intro/)

[Interchain Security](https://ida.interchain.io/tutorials/5-docker-intro/)

[Bridges](https://ida.interchain.io/tutorials/5-docker-intro/)

[Migrations](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 1 Quiz](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 2 - First Steps](https://ida.interchain.io/tutorials/5-docker-intro/)

[First Steps](https://ida.interchain.io/tutorials/5-docker-intro/)

[Setup Your Work Environment](https://ida.interchain.io/tutorials/5-docker-intro/)

[Run a Node, API, and CLI](https://ida.interchain.io/tutorials/5-docker-intro/)

[Ignite CLI](https://ida.interchain.io/tutorials/5-docker-intro/)

[Exercise - Make a Checkers Blockchain](https://ida.interchain.io/tutorials/5-docker-intro/)

[Store Object](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create Custom Messages](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create and Save a Game Properly](https://ida.interchain.io/tutorials/5-docker-intro/)

[Add a Way to Make a Move](https://ida.interchain.io/tutorials/5-docker-intro/)

[Emit Game Information](https://ida.interchain.io/tutorials/5-docker-intro/)

[Record the Game Winner](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 2 Exercise](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 3 - Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/5-docker-intro/)

[Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/5-docker-intro/)

[What is IBC?](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC/TAO - Connections (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC/TAO - Channels (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC/TAO - Clients (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC Token Transfer](https://ida.interchain.io/tutorials/5-docker-intro/)

[Interchain Accounts (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create a Custom IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[Integrate IBC Middleware Into a Chain (OPTIONAL)](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC Tooling](https://ida.interchain.io/tutorials/5-docker-intro/)

[What is CosmJS?](https://ida.interchain.io/tutorials/5-docker-intro/)

[Your First CosmJS Actions](https://ida.interchain.io/tutorials/5-docker-intro/)

[Compose Complex Transactions](https://ida.interchain.io/tutorials/5-docker-intro/)

[Learn to Integrate Keplr](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create Custom CosmJS Interfaces](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 4 - Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/5-docker-intro/)

[Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/5-docker-intro/)

[Keep an Up-To-Date Game Deadline](https://ida.interchain.io/tutorials/5-docker-intro/)

[Keep Track Of How Many Moves Have Been Played](https://ida.interchain.io/tutorials/5-docker-intro/)

[Put Your Games in Order](https://ida.interchain.io/tutorials/5-docker-intro/)

[Auto-Expiring Games](https://ida.interchain.io/tutorials/5-docker-intro/)

[Let Players Set a Wager](https://ida.interchain.io/tutorials/5-docker-intro/)

[Handle wager payments](https://ida.interchain.io/tutorials/5-docker-intro/)

[Integration tests](https://ida.interchain.io/tutorials/5-docker-intro/)

[Incentivize Players](https://ida.interchain.io/tutorials/5-docker-intro/)

[Help Find a Correct Move](https://ida.interchain.io/tutorials/5-docker-intro/)

[Play With Cross-Chain Tokens](https://ida.interchain.io/tutorials/5-docker-intro/)

[Understand IBC Denoms](https://ida.interchain.io/tutorials/5-docker-intro/)

[Go Relayer](https://ida.interchain.io/tutorials/5-docker-intro/)

[Hermes Relayer](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 5 - CosmJS Advanced](https://ida.interchain.io/tutorials/5-docker-intro/)

[CosmJS Advanced](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create Custom Objects](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create Custom Messages](https://ida.interchain.io/tutorials/5-docker-intro/)

[Get an External GUI](https://ida.interchain.io/tutorials/5-docker-intro/)

[Integrate CosmJS and Keplr](https://ida.interchain.io/tutorials/5-docker-intro/)

[Backend Script for Game Indexing](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 6 - IBC Deep Dive](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC Deep Dive](https://ida.interchain.io/tutorials/5-docker-intro/)

[IBC Application Developer Introduction](https://ida.interchain.io/tutorials/5-docker-intro/)

[Make a Module IBC-Enabled](https://ida.interchain.io/tutorials/5-docker-intro/)

[Adding Packet and Acknowledgment Data](https://ida.interchain.io/tutorials/5-docker-intro/)

[Extend the Checkers Game With a Leaderboard](https://ida.interchain.io/tutorials/5-docker-intro/)

[Create a Leaderboard Chain](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[Week 7 - From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/5-docker-intro/)

[From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/5-docker-intro/)

[Run in Production](https://ida.interchain.io/tutorials/5-docker-intro/)

[Prepare the Software to Run](https://ida.interchain.io/tutorials/5-docker-intro/)

[Prepare a Validator and Keys](https://ida.interchain.io/tutorials/5-docker-intro/)

[Prepare Where the Node Starts](https://ida.interchain.io/tutorials/5-docker-intro/)

[Prepare and Connect to Other Nodes](https://ida.interchain.io/tutorials/5-docker-intro/)

[Configure, Run, and Set Up a Service](https://ida.interchain.io/tutorials/5-docker-intro/)

[Prepare and Do Migrations](https://ida.interchain.io/tutorials/5-docker-intro/)

[Simulate Production in Docker](https://ida.interchain.io/tutorials/5-docker-intro/)

[Tally Player Info After Production](https://ida.interchain.io/tutorials/5-docker-intro/)

[Add a Leaderboard as a Module](https://ida.interchain.io/tutorials/5-docker-intro/)

[Migrate the Leaderboard Module After Production](https://ida.interchain.io/tutorials/5-docker-intro/)

[Simulate a Migration in Docker](https://ida.interchain.io/tutorials/5-docker-intro/)

[Final Exam](https://ida.interchain.io/tutorials/5-docker-intro/)

[](https://ida.interchain.io/tutorials/5-docker-intro/)

[What's Next?](https://ida.interchain.io/tutorials/5-docker-intro/)

[Continue Your Interchain Journey](https://ida.interchain.io/tutorials/5-docker-intro/)

Docs Version Switcher

On this page

[Concepts](https://ida.interchain.io/tutorials/5-docker-intro/#concepts)

[Containers](https://ida.interchain.io/tutorials/5-docker-intro/#containers)

[Images](https://ida.interchain.io/tutorials/5-docker-intro/#images)

[How to use it](https://ida.interchain.io/tutorials/5-docker-intro/#how-to-use-it)

[Open a shell](https://ida.interchain.io/tutorials/5-docker-intro/#open-a-shell)

[Your own image](https://ida.interchain.io/tutorials/5-docker-intro/#your-own-image)

[Hello World](https://ida.interchain.io/tutorials/5-docker-intro/#hello-world)

[Sharing folders](https://ida.interchain.io/tutorials/5-docker-intro/#sharing-folders)

[Clean up](https://ida.interchain.io/tutorials/5-docker-intro/#clean-up)

Know your way around Docker

**Docker Introduction**

[Docker (opens new window)↗](https://www.docker.com/) is a kind of computer virtualization that provides valuable performance benefits.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#concepts) Concepts

Being different from classic forms of virtualization, Docker introduces new concepts with their vocabulary.

Take a look at two of them: containers and images.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#containers) Containers

You know how a **computer** operates: it uses a disk and memory to run an operating system and the programs within it.

You know how a **virtual machine** works: its computer host offers a *virtual* disk and memory, and the virtual machine uses both to run its own operating system and programs. Virtualization mimics a full computer, which requires disk space and memory. It also takes time to start because the virtualized operating system needs to start from scratch.

Docker uses a feature of Linux called [*namespaces* (opens new window)↗](https://docs.docker.com/get-started/overview/#the-underlying-technology) and *control groups*. With this feature, Linux can launch programs (even low-level ones of the operating system) in an environment where the programs believe they are in a complete Linux environment on their own, i.e. sandboxed. This is also called containerization. A **container** does not have full disk or full memory access to its host (the Linux environment that started it), only to a subset of it. A Linux system can launch multiple containers at a time.

As opposed to proper virtualization, containerization is fast:

* You do not need to start a whole operating system, only to isolate a new container and run its own programs. You now count the start-up time in seconds instead of minutes.
* The memory used by the container is only that of the programs it runs while benefiting from the rest of the Linux system in an isolated way.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#images) Images

Instead of a virtual disk with a full operating system, a container starts from an **image**. The image contains the files that are specific to the container and different from its Linux host. Think of tracing paper recording only the differences laid on top of the host's file system. These differences can be very small.

You can either create images locally via a **Dockerfile** containing the commands that create the image, or you can use an existing image.



There exist registries that store and/or reference them, like your regular package manager. [Docker Hub (opens new window)↗](https://hub.docker.com/) is the main one, and your local Docker knows how to download directly from it.

This introduces an interesting concept whereby [a machine is described as a file (opens new window)↗](https://twitter.com/FrancescoCiull4/status/1509458241524224005), which is very useful for reproducibility and DevOps. Further, the images are optimized so that each image is a *diff* of a parent one.

Images can be versioned and even referenced by their content hash so that you can be sure to use the expected one. For instance, [Node.js (opens new window)↗](https://hub.docker.com/_/node) has a long list of images. Do you want [Node 19.1 (opens new window)↗](https://hub.docker.com/layers/library/node/19.1/images/sha256-fcf7d55d2bea9d86f6890a8c44aec9a9ae2cb8f6351aae50e9d684fc81a4415f), or [Node 19.1 built specifically on Debian Buster (opens new window)↗](https://hub.docker.com/layers/library/node/19.1-buster/images/sha256-9d37aa88366e0a26b621c84e6cb9aff5bee5589d1e783f84d053f7ffe93cfb82)?

Because images contain files to be executed, the files also need to have been compiled for the CPU architecture of your Linux machine. This is why images are often uploaded in different "OS/ARCH" versions, as can be seen [in the Node 19.1 page (opens new window)↗](https://hub.docker.com/layers/library/node/19.1-buster/images/sha256-9d37aa88366e0a26b621c84e6cb9aff5bee5589d1e783f84d053f7ffe93cfb82?context=explore).

For the avoidance of doubt:

* An image is read-only, and when your container starts its read-write file system is a separate entity.
* More than one container can be started from the same image at the same time.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#how-to-use-it) How to use it

What if you do not have a Linux operating system? Not to worry, Docker simplifies your life by installing and running a virtual machine running a barebones Linux on your host computer.

After installation, when you want to use Docker, you *start Docker* - you start the virtual machine running Linux, which is the part that takes time. After it has started, you can use commands to run containers. When you no longer need to use Docker, you can stop it and regain the memory it used.

In these tutorials, you will come across a lot of Docker commands, so it makes sense to familiarize yourself with them.

First, [install it (opens new window)↗](https://www.docker.com/get-started). Next start Docker.

When it has started you can run your first container. For instance, with Node.js' lts-slim image:



Copy

$ docker run -it node:lts-slim

-it is short for --interactive --tty and means "with input and output", instead of a fully detached container. Learn more with docker run --help. This should return you a Node.js prompt:



Copy

Welcome to Node.js v18.12.1.

Type ".help" for more information.

>

Version 18 is a long-term support (lts) version. If you type .help, it will tell you what you need. Exit with .exit.

That was fast. What happened?

Docker downloaded the image from the hub and launched it. By default, the image is configured so that the container launches node, which is what you got. You did not do anything interesting, though. Yet.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#open-a-shell) Open a shell

What if you want to connect to your container with a shell? After all, this is running Linux. Because this image defines Node as its [**entrypoint** (opens new window)↗](https://hub.docker.com/layers/library/node/lts-slim/images/sha256-3139aa3e8915e7c135623498d29f20a75ee3bfc41cf321ceaa59470b2fffc1a5?context=explore), you need to override it:



Copy

$ docker run -it --entrypoint bash node:lts-slim

Now you see something different:



Copy

root@bd3982cf3d68:/#

You are root! But you are root **only in the container**, not in the Linux host. Typically, programs running in a container are left running with root, as the container takes care of the isolation.

The image node:lts-slim, as the slim part indicates, does not have much else besides Node. Try:



Copy

$ curl

It should tell you:



Copy

bash: curl: command not found

This means that you need to pick your image carefully, and even then sometimes you will also have to install the tools you need. Exit with a regular exit.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#your-own-image) Your own image

Suppose you need a Node container with the curl command available. For that, you need to **build** your image. It so happens that node:lts-slim is built on Debian, so you can use apt-get to install new programs.

Create a new file named Dockerfile with:



Copy

FROM node:lts-slim

RUN apt-get update

RUN apt-get install -y curl

You have to update the package registry because the image is kept slim by not even having a local copy. Build your new image with a name of your choosing:



Copy

$ docker build . -t node-with-curl:lts-slim

Now your image is ready.

Can you call up curl? Type:



Copy

$ docker run -it --entrypoint curl node-with-curl:lts-slim

It should return:



Copy

Use "--help category" to get an overview of all categories.

For all options use the manual or "--help all".

So yes, you now have curl and Node in the same container.

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#hello-world) Hello World

What if you wanted to use Node to print Hello World? Create the JavaScript file that can do that:



Copy

$ echo "console.log(\"Hello World\");" > test.js

Now pass it to your container:



Copy

$ docker run -it node:lts-slim test.js

This does not work:



Copy

Error: Cannot find module '/test.js'

This is because Docker took the words test.js and passed them to Node as a string within the context of the container, which has no test.js. In effect, the container ran node test.js. The file is currently only on your host computer.

Try again:



Copy

$ docker run -it node:lts-slim < test.js

Here the content of the file is being passed via StdIn. It should still complain:



Copy

the input device is not a TTY

So remove -t:



Copy

$ docker run -i node:lts-slim < test.js

This time it prints:



Copy

Hello World

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#sharing-folders) Sharing folders

Shared folders are called **volumes**. Instead of sending the content of test.js to the container via StdIn, it could be more judicious to let the container have access to your file directly. Try again, this time by sharing your local folder (pwd) with the container, and mounting it at /root/temp inside the container:



Copy

$ docker run -it -v $(pwd):/root/temp node:lts-slim test.js

It should complain again:



Copy

Error: Cannot find module '/test.js'

To progress, you also have to tell it to work (-w) in the right folder: /root/temp, which you created to access your local files:



Copy

$ docker run -it -v $(pwd):/root/temp -w /root/temp node:lts-slim test.js

This time it returns:



Copy

Hello World

[#Copy link](https://ida.interchain.io/tutorials/5-docker-intro/#clean-up) Clean up

You ran quite a few commands. Where did your containers go?



Copy

$ docker ps --all

You should see a long list like this:



Copy

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

b5930aef7e7a node:lts-slim "docker-entrypoint.s…" 2 minutes ago Exited (0) 2 minutes ago angry\_babbage

e3f58f2cb118 node:lts-slim "docker-entrypoint.s…" 5 minutes ago Exited (0) 5 minutes ago affectionate\_elgamal

...

All have stopped and exited. When you create a container, it is not removed by default so it can be reused (see docker exec --help). For now, you should clean up these pointless containers - **however**, if you have containers that are not part of this introduction, do not run the command:



Copy

$ docker container prune



Having many stopped containers is not ideal, which is why when you want to run a container for a single command the practice is to add --rm, like so:



Copy

$ docker run --rm -it -v $(pwd):/root/temp -w /root/temp node:lts-slim test.js

Including --rm automatically removes the container at the moment it is stopped and exited. You can confirm that there are no remaining containers.

What about the images?



Copy

$ docker image ls

There you should see your images too:



Copy

REPOSITORY TAG IMAGE ID CREATED SIZE

node-with-curl lts-slim 0c75a35d23de 12 minutes ago 266MB

node lts-slim 103943353fa6 9 days ago 241MB

You can delete them, in any order, with:



Copy

$ docker image rm node-with-curl:lts-slim

$ docker image rm node:lts-slim

This concludes your introduction to Docker.



**Further reading**

* [Under the hood (opens new window)↗](https://www.codementor.io/blog/docker-technology-5x1kilcbow).

synopsis

To summarize, this section has explored:

* What are Docker images and containers.
* How to use a container.
* How to share folders via volumes.
* How to create an image.
* How to clean up.

With these basics, you are equipped to handle the Docker examples of the tutorials.

previous

[](https://ida.interchain.io/tutorials/1-tech-terms/)

**[Good-To-Know Dev Terms](https://ida.interchain.io/tutorials/1-tech-terms/)**

up next

**[Introduction to the Interchain](https://ida.interchain.io/ida-course/LPs/week-1/)**

[[](https://ida.interchain.io/ida-course/LPs/week-1/)](https://ida.interchain.io/ida-course/LPs/week-1/)

Rate this Page

icon smile

icon meh

icon frown

Would you like to add a message?

Submit

Thank you for your Feedback!

On this page

[Concepts](https://ida.interchain.io/tutorials/5-docker-intro/#concepts)

[Containers](https://ida.interchain.io/tutorials/5-docker-intro/#containers)

[Images](https://ida.interchain.io/tutorials/5-docker-intro/#images)

[How to use it](https://ida.interchain.io/tutorials/5-docker-intro/#how-to-use-it)

[Open a shell](https://ida.interchain.io/tutorials/5-docker-intro/#open-a-shell)

[Your own image](https://ida.interchain.io/tutorials/5-docker-intro/#your-own-image)

[Hello World](https://ida.interchain.io/tutorials/5-docker-intro/#hello-world)

[Sharing folders](https://ida.interchain.io/tutorials/5-docker-intro/#sharing-folders)

[Clean up](https://ida.interchain.io/tutorials/5-docker-intro/#clean-up)

#### **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)