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

[Interchain Developer Academy](https://ida.interchain.io/)/[Interchain Developer Academy](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)



Search

[Interchain Developer Academy](https://ida.interchain.io/)[Interchain Developer Academy](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

Search



Filters

Interchain Developer Academy

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 0 - Getting Started](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Getting Started](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Blockchain 101](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Blockchain History](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Public and Managed Blockchains](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Consensus in Distributed Networks](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Cryptography](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Self-Assessment Quiz](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Go Introduction - First Steps](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Go Basics](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Go Interfaces](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Control Structures in Go](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Arrays and Slices in Go](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Standard Packages in Go](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Concurrency in Go](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Good-To-Know Dev Terms](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Docker Introduction](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 1 - Introduction to the Interchain](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Introduction to the Interchain](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Blockchain Technology and the Interchain](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[The Interchain Ecosystem](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Getting ATOM and Staking It](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[A Blockchain App Architecture](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Accounts](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Transactions](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Modules](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Protobuf](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Multistore and Keepers](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[BaseApp](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Queries](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Events](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Context](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Testing](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Relaying with IBC](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Interchain Security](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Bridges](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Migrations](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 1 Quiz](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 2 - First Steps](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[First Steps](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Setup Your Work Environment](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Run a Node, API, and CLI](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Ignite CLI](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Exercise - Make a Checkers Blockchain](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Store Object](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create Custom Messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create and Save a Game Properly](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Add a Way to Make a Move](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Emit Game Information](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Record the Game Winner](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 2 Exercise](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 3 - Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Introduction to IBC and CosmJS](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[What is IBC?](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC/TAO - Connections (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC/TAO - Channels (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC/TAO - Clients (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC Token Transfer](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Interchain Accounts (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create a Custom IBC Middleware (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Integrate IBC Middleware Into a Chain (OPTIONAL)](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC Tooling](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[What is CosmJS?](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Your First CosmJS Actions](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Compose Complex Transactions](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Learn to Integrate Keplr](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create Custom CosmJS Interfaces](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 4 - Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Ignite CLI and IBC Advanced](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Keep an Up-To-Date Game Deadline](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Keep Track Of How Many Moves Have Been Played](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Put Your Games in Order](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Auto-Expiring Games](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Let Players Set a Wager](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Handle wager payments](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Integration tests](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Incentivize Players](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Help Find a Correct Move](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Play With Cross-Chain Tokens](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Understand IBC Denoms](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Go Relayer](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Hermes Relayer](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 5 - CosmJS Advanced](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[CosmJS Advanced](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create Custom Objects](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create Custom Messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Get an External GUI](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Integrate CosmJS and Keplr](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Backend Script for Game Indexing](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 6 - IBC Deep Dive](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC Deep Dive](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[IBC Application Developer Introduction](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Make a Module IBC-Enabled](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Adding Packet and Acknowledgment Data](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Extend the Checkers Game With a Leaderboard](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Create a Leaderboard Chain](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Week 7 - From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[From Code to MVP to Production and Migrations](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Run in Production](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Prepare the Software to Run](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Prepare a Validator and Keys](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Prepare Where the Node Starts](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Prepare and Connect to Other Nodes](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Configure, Run, and Set Up a Service](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Prepare and Do Migrations](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Simulate Production in Docker](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Tally Player Info After Production](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Add a Leaderboard as a Module](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Migrate the Leaderboard Module After Production](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Simulate a Migration in Docker](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Final Exam](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[What's Next?](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

[Continue Your Interchain Journey](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html)

Docs Version Switcher

On this page

[Send multiple tokens using sendTokens](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#send-multiple-tokens-using-sendtokens)

[Introducing signAndBroadcast](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#introducing-signandbroadcast)

[Token transfer messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#token-transfer-messages)

[What is this long string?](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#what-is-this-long-string)

[Multiple token transfer messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#multiple-token-transfer-messages)

[Mixing other message types](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#mixing-other-message-types)

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#compose-complex-transactions) **Compose Complex Transactions**



In the Interchain, a transaction is able to encapsulate multiple messages.   
  
In this section, you will:

* Send multiple tokens in a single transaction.
* Sign and broadcast.
* Assemble multiple messages.

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#send-multiple-tokens-using-sendtokens) Send multiple tokens using sendTokens

In the [previous exercise](https://ida.interchain.io/tutorials/7-cosmjs/2-first-steps.html), you had Alice send tokens back to the faucet. To refresh your memory, this is what the sendTokens function takes as input:



Copy

public async sendTokens(

senderAddress: string,

recipientAddress: string,

amount: readonly Coin[],

fee: StdFee | "auto" | number,

memo = "",

): Promise<DeliverTxResponse>;

packages /

stargate /

src /

signingstargateclient.ts

[View source→](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/signingstargateclient.ts" \l "L180-L186" \t "_blank)

[Coin (opens new window)↗](https://github.com/confio/cosmjs-types/blob/a14662d/src/cosmos/base/v1beta1/coin.ts#L13-L16) allows Alice to send not just stake but also any number of other coins as long as she owns them. So she can:

**Send one token type**

**Send two token types**



Copy

const result = await signingClient.sendTokens(

alice,

faucet,

[

{ denom: "uatom", amount: "100000" },

],

{

amount: [{ denom: "uatom", amount: "500" }],

gas: "200000",

},

)

Copy

const result = await signingClient.sendTokens(

alice,

faucet,

[

{ denom: "uatom", amount: "100000" },

{ denom: "token", amount: "12" },

],

{

amount: [{ denom: "uatom", amount: "500" }],

gas: "200000",

},

)

However, there are limitations with this function. First, Alice **can only target a single recipient per transaction**, faucet in the previous examples. If she wants to send tokens to multiple recipients, then she needs to create as many transactions as there are recipients. Multiple transactions cost slightly more than packing transfers into the array because of transaction overhead. Additionally, in some cases it is considered a bad user experience to make users sign multiple transactions.

The second limitation is that **separate transfers are not atomic**. It is possible that Alice wants to send tokens to two recipients and it is important that either they both receive them or neither of them receive anything.

Fortunately, there is a way to atomically send tokens to multiple recipients.

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#introducing-signandbroadcast) Introducing signAndBroadcast

SigningStargateClient has the signAndBroadcast function:



Copy

public async signAndBroadcast(

signerAddress: string,

messages: readonly EncodeObject[],

fee: StdFee | "auto" | number,

memo = "",

): Promise<DeliverTxResponse>;

packages /

stargate /

src /

signingstargateclient.ts

[View source→](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/signingstargateclient.ts" \l "L280-L285" \t "_blank)

The basic components of a transaction are the signerAddress, the messages that it contains, as well as the fee and an optional memo. As such, [Cosmos transactions](https://ida.interchain.io/academy/2-cosmos-concepts/3-transactions.html) can indeed be composed of multiple [messages](https://ida.interchain.io/academy/2-cosmos-concepts/4-messages.html).

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#token-transfer-messages) Token transfer messages

In order to use signAndBroadcast to send tokens, you need to figure out what messages go into the messages: readonly EncodeObject[]. Examine the sendTokens function body:



Copy

const sendMsg: MsgSendEncodeObject = {

typeUrl: "/cosmos.bank.v1beta1.MsgSend",

value: {

fromAddress: senderAddress,

toAddress: recipientAddress,

amount: [...amount],

},

};

return this.signAndBroadcast(senderAddress, [sendMsg], fee, memo);

packages /

stargate /

src /

signingstargateclient.ts

[View source→](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/signingstargateclient.ts" \l "L187-L195" \t "_blank)

Therefore, when sending back to the faucet, instead of calling:



Copy

const result = await signingClient.sendTokens(

alice,

faucet,

[{ denom: "uatom", amount: "100000" }],

{

amount: [{ denom: "uatom", amount: "500" }],

gas: "200000",

},

)

Alice may as well call:



Copy

const result = await signingClient.signAndBroadcast(

// the signerAddress

alice,

// the message(s)

[

{

typeUrl: "/cosmos.bank.v1beta1.MsgSend",

value: {

fromAddress: alice,

toAddress: faucet,

amount: [

{ denom: "uatom", amount: "100000" },

],

},

},

],

// the fee

{

amount: [{ denom: "uatom", amount: "500" }],

gas: "200000",

},

)

Confirm this by making the change in your experiment.ts from the previous section, and running it again.



Building a transaction in this way is recommended. SigningStargateClient offers you convenience methods such as sendTokens for simple use cases, and to demonstrate how to build messages.



If you are wondering whether there could be any legitimate situation where the transaction's signer (here alice) could ever be different from the message's fromAddress (here alice too), then have a look at the [tutorial on authz](https://ida.interchain.io/tutorials/8-understand-sdk-modules/1-authz.html).

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#what-is-this-long-string) What is this long string?

As a reminder from the previous tutorial, the typeUrl: "/cosmos.bank.v1beta1.MsgSend" string comes from the [Protobuf](https://ida.interchain.io/academy/2-cosmos-concepts/6-protobuf.html) definitions and is a mixture of:

1. The package where MsgSend is initially declared:



Copy

package cosmos.bank.v1beta1;

proto /

cosmos /

... /

v1beta1 /

tx.proto

[View source→](https://github.com/cosmos/cosmos-sdk/blob/3a1027c/proto/cosmos/bank/v1beta1/tx.proto" \l "L2" \t "_blank)

1. And the name of the message itself, MsgSend:



Copy

message MsgSend {

...

}

proto /

cosmos /

... /

v1beta1 /

tx.proto

[View source→](https://github.com/cosmos/cosmos-sdk/blob/3a1027c/proto/cosmos/bank/v1beta1/tx.proto" \l "L22" \t "_blank)



To learn how to make your own types for your own blockchain project, head to [Create Custom CosmJS Interfaces](https://ida.interchain.io/tutorials/7-cosmjs/5-create-custom.html).

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#multiple-token-transfer-messages) Multiple token transfer messages

From here, you add an extra message for a token transfer from Alice to someone else:



Copy

const result = await signingClient.signAndBroadcast(

// signerAddress

alice,

[

// message 1

{

typeUrl: "/cosmos.bank.v1beta1.MsgSend",

value: {

fromAddress: alice,

toAddress: faucet,

amount: [

{ denom: "uatom", amount: "100000" },

],

},

},

// message 2

{

typeUrl: "/cosmos.bank.v1beta1.MsgSend",

value: {

fromAddress: alice,

toAddress: some\_other\_address,

amount: [

{ denom: "token", amount: "10" },

],

},

},

],

// the fee

"auto",

)

Note how the custom fee input was replaced with the auto input, which simulates the transaction to estimate the fee for you. In order to make that work well, you need to define the gasPrice you are willing to pay and its prefix when setting up your signingClient. You replace your original line of code with:



Copy

const signingClient = await SigningStargateClient.connectWithSigner(

rpc,

aliceSigner,

+ {

+ prefix: "cosmos",

+ gasPrice: GasPrice.fromString("0.0025uatom")

+ }

)

[#Copy link](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#mixing-other-message-types) Mixing other message types

The above example shows you two token-transfer messages in a single transaction. You can see this with their typeUrl: "/cosmos.bank.v1beta1.MsgSend".

Neither the Cosmos SDK nor CosmJS limits you to combining messages of the same type. You can decide to combine other message types together with a token transfer. For instance, in one transaction Alice could:

1. Send tokens to the faucet.
2. Delegate some of her tokens to a validator.

How would Alice create the second message? The SigningStargateClient contains a predefined list (a *registry*) of typeUrls that are [supported by default (opens new window)↗](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/signingstargateclient.ts#L55-L69), because they're considered to be the most commonly used messages in the Cosmos SDK. Among the [staking types (opens new window)↗](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/signingstargateclient.ts#L62) there is MsgDelegate, and that is exactly what you need. Click the source links above and below to see the rest of the typeUrls that come with SigningStargateClient:



Copy

export const stakingTypes: ReadonlyArray<[string, GeneratedType]> = [

...

["/cosmos.staking.v1beta1.MsgDelegate", MsgDelegate],

...

];

packages /

stargate /

... /

staking /

messages.ts

[View source→](https://github.com/cosmos/cosmjs/blob/v0.28.2/packages/stargate/src/modules/staking/messages.ts" \l "L13" \t "_blank)

Click through to the type definition, and in the cosmjs-types repository:



Copy

export interface MsgDelegate {

delegatorAddress: string;

validatorAddress: string;

amount?: Coin;

}

src /

cosmos /

... /

v1beta1 /

tx.ts

[View source→](https://github.com/confio/cosmjs-types/blob/v0.4.1/src/cosmos/staking/v1beta1/tx.ts" \l "L46-L50" \t "_blank)

Now that you know the typeUrl for delegating some tokens is /cosmos.staking.v1beta1.MsgDelegate, you need to find a validator's address that Alice can delegate to. Find a list of validators in the [testnet explorer (opens new window)↗](https://explorer.theta-testnet.polypore.xyz/validators). Select a validator and set their address as a variable:



Copy

const validator: string = "cosmosvaloper178h4s6at5v9cd8m9n7ew3hg7k9eh0s6wptxpcn" //01node

Use this variable in the following script, which you can copy to replace your original token transfer:



Copy

const result = await signingClient.signAndBroadcast(

alice,

[

{

typeUrl: "/cosmos.bank.v1beta1.MsgSend",

value: {

fromAddress: alice,

toAddress: faucet,

amount: [

{ denom: "uatom", amount: "100000" },

],

},

},

{

typeUrl: "/cosmos.staking.v1beta1.MsgDelegate",

value: {

delegatorAddress: alice,

validatorAddress: validator,

amount: { denom: "uatom", amount: "1000", },

},

},

],

"auto"

)

When you create [your own message types in CosmJS](https://ida.interchain.io/tutorials/7-cosmjs/5-create-custom.html), they have to follow this format and be declared in the same fashion.

synopsis

To summarize, this section has explored:

* How to move past the one-transaction-one-recipient limitations of the previous exercise, which could compel a user to sign potentially many transactions at a time, and denies the possibility of sending *atomic* transactions to multiple recipients (for example, a situation in which either all recipients receive tokens or none of them do).
* How to include two token-transfer messages in a single transaction, and how to combine messages of different types in a single transaction (for example, sending tokens to the faucet *and* delegating tokens to a validator).

previous

[](https://ida.interchain.io/tutorials/7-cosmjs/2-first-steps.html)

**[Your First CosmJS Actions](https://ida.interchain.io/tutorials/7-cosmjs/2-first-steps.html)**

up next

**[Learn to Integrate Keplr](https://ida.interchain.io/tutorials/7-cosmjs/4-with-keplr.html)**

[[](https://ida.interchain.io/tutorials/7-cosmjs/4-with-keplr.html)](https://ida.interchain.io/tutorials/7-cosmjs/4-with-keplr.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

[Send multiple tokens using sendTokens](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#send-multiple-tokens-using-sendtokens)

[Introducing signAndBroadcast](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#introducing-signandbroadcast)

[Token transfer messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#token-transfer-messages)

[What is this long string?](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#what-is-this-long-string)

[Multiple token transfer messages](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#multiple-token-transfer-messages)

[Mixing other message types](https://ida.interchain.io/tutorials/7-cosmjs/3-multi-msg.html#mixing-other-message-types)

#### **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)