

# CLI - Managing tokens

## How to query your balance

Once fetchd is configured for the desired [network ↗](#) . The user can query their balance using the following command:

```
fetchd
query
bank
balances
```

fetch1akvyhle79nts4rwn075t85xrwmp5ysuqynxcn4 If the address exists on the network, then the user will expect to see an output in the following form:

```
balances: - amount: "8000000000000000000" denom: atestfet pagination: next_key: null total: "0"
```

## How to send funds

Before sending funds, make sure the sender address has tokens available by querying your balance as shown above. Checkout the [token faucet](#) page for more information on how to add test tokens to your address.

If you wish to send funds from one address to another, then you would use the tx send subcommand as shown below:

```
fetchd
tx
bank
send
< from
address
or
key
nam e
< target
address s
< amount
```

In a more concrete example, if the user wanted to send 100 atestfet from main key/address to fetch106vm9q6ezu9va7v7e0cvq0nedc54egjm692fcp , then the following command would be needed:

```
fetchd
tx
bank
send
main
fetch106vm9q6ezu9va7v7e0cvq0nedc54egjm692fcp
```

100 atestfet When you run the command, you will get a similar output and prompt. The user can check the details of the transfer and then press y to confirm the transfer.

```
{"body":{"messages":
[{"@type":"/cosmos.bank.v1beta1.MsgSend","from_address":"fetch12cjntwl32dry7fxck8qlgxq6na3fk5juwjdy3","to_address":"fetch1hph8kd54gl6qk0hy5rl08qw9gcr4vitmk3w02v","amount":
[{"denom":"atestfet","amount":"100"}]}], "memo":"","timeout_height":"0","extension_options":[],"non_critical_extension_options":[],"auth_info":{"signer_infos":[],"fee":{"amount":
[], "gas_limit":"200000","payer":"","granter":""}}, "signatures":[]}
```

confirm transaction before signing and broadcasting [y/N]: y Once the transfer has been made, a summary is presented to the user. An example is shown below:

```
code: 0 codespace: "" data: "" gas_used: "0" gas_wanted: "0" height: "0" info: "" logs: [] raw_log: '' timestamp: "" tx: null txhash:
77C7382A0B1B9FE39257A6C16C7E3169A875CB3A87F2CE9D947D7C1335B53E76 On failure, the response will have a non-zero code, as well as some logs under the raw_log key:
```

```
code: 4 codespace: sdk data: "" gas_used: "0" gas_wanted: "0" height: "0" info: "" logs: [] raw_log: 'signature verification failed; please verify account number (5815) and chain-id (dorado-
1): unauthorized' timestamp: "" tx: null txhash: 23701B052B423D63EB4AC94773B5B8227B03A576692A57999E92F2554F2372D4
```

## Was this page helpful?

[CLI - Managing keys](#) [CLI - Multisig keys](#)