

Week3

Laboratory 03

Stellar Technical Academy



Self-Assessment Exercises

Objectives

- The purpose of this Week 3 exercise is to acquaint you with the token economics and dynamics of Stellar.
- In order to do so, you are going to learn how to issue an asset on the Stellar network.
- In this exercise you will:
 - Use the Albedo wallet that you created in the Week 1 Laboratory as the Distributor and the xBull wallet you created in the Week 2 Laboratory as the Issuer
- If you have issued the asset successfully, you are finished and you can receive your badge NFT certificate for Week 3!

Create a Stellar-network token

Introduction

- In Stellar network, any account can issue an asset and as we saw previously, and anyone can set up a Stellar account.
- To do it, Stellar provides built-in mechanisms that allow you to tune your asset to specific use cases.
- This is the first exercise that helps you see how Stellar can be of valuable help in real-world use cases.



Self-Assessment Exercises

Exercise 3 – Issue a custom asset using the Stellar Laboratory

1. You are going to create a Stellar-network token using the Stellar SDK.
2. The exercise is divided into issuing an asset and then publishing information about the asset.

The first part is split into 3 steps

- Create an Issuing and Distribution account
- Create a Trustline
- Create a Payment transaction

The second part shows how the information is created and stored in an asset.

Self-Assessment Exercises

Exercise 3 – Issue a custom asset using the Stellar Laboratory

- In the first part, familiarize yourself with the Stellar Laboratory tools presented in the previous slides, specifically the Stellar Laboratory and the Stellar Expert tools.
- More specifically, as it was described in the Theory part, you will:
 - Get to know the anatomy of an asset
 - Start to issue an asset
 - Learn why two accounts are necessary
 - Get to know the importance of a Trustline
 - Executing a payment transaction
- Then, you will check on the Stellar Expert to verify that the transaction was successfully made.
- In the second part, you will learn how to publish information about your network token.

Create a Stellar-network token

Exercise 3 – Issue a custom asset using the Stellar Laboratory - Anatomy of an Asset

Stellar assets have two characteristics.

- The asset code
- The asset issuer

And, currently, there are two supported formats:

- Alphanumeric 4-character maximum.
- Alphanumeric 12-character maximum.

Further, although you can choose any asset code you like, to allow Stellar users to easily identify what a token represents, there are recommendations as follows:

- For currencies, use the ISO 4217 code standard.
- For stocks or bonds, use the ISIN number standard.

Create a Stellar-network token

Exercise 3 – Issue a custom asset using the Stellar Laboratory - How is an asset created?

There is no reliable operation to create an asset on Stellar. Instead, assets are created with a payment operation using two accounts.

The first account will be the **issuing account** that makes a payment using the asset it's issuing and creates the asset through that payment. Also, the public key or address of the issuing account will be the one linked on the ledger to the asset. It has control over the asset (issuing control, meta-data about the asset and authorization flags).

The second account is the **base or distribution account**, which is the one that transacts with other Stellar accounts. It holds the asset balance issued by the issuing account since the creation of the asset is done like a payment transaction, from the issuing account to the base account.

Create a Stellar-network token

Exercise 3 – Issue a custom asset using the Stellar Laboratory!

To create the asset with a payment operation four steps are required:

- Create an Issuing account
- Create a Distribution account
- Establish a Trustline
- Make a payment

After asset issuance, it is necessary to provide a clear representation about what the asset represents and publish that information using tools in the Stellar Laboratory.

The process can be done in two ways:

- Using the Stellar laboratory
- Or by code.

We are going to utilize the Albedo wallet and the xBull wallet we have created in the previous weeks as Distributor and Issuer, respectively.

Create a Stellar-network token

Exercise 3 – Issue a custom asset using the Stellar Laboratory - Why have two accounts?

Why have separate accounts for issuing and distribution?

- Security:
 - Some web services have access to sign transactions on the distribution account. If a bad actor gains access to the account you use to distribute your asset and it is also the issuing account, the malicious actor can now issue as much of your asset as they like

But if the two accounts are different, when a compromised account is identified, the issuer account can essentially freeze the asset amount of the compromised account and start over with a new distribution account. This can be done without having to change the issuing account.

- Auditing
 - The issuing account is unable to maintain a balance in its own asset. It's easier to keep track of your personal assets if you keep track of them in a separate account. This is a pattern that can be found in a variety of ledgering solutions.

Distribution accounts also isolate our ecosystem standards from issuance, which is a nice plus. This enables ecosystem players to produce novel ideas such as non-issuing Anchors without having to change protocol.

Create a Stellar-network token

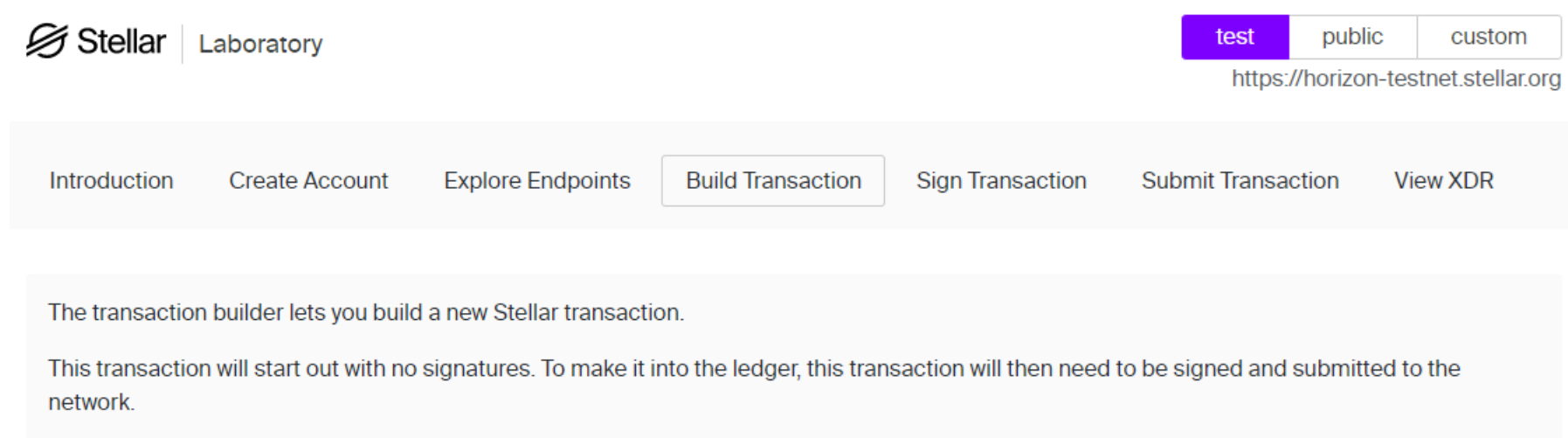
Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- As previously mentioned, you are going to use the Albedo wallet you created on Week 1 as the Distributor and the xBull wallet you created on Week 2 as the Issuer.
- In case that the wallets are not activated, you are going to follow the next steps:
- In order to activate the xBull wallet, which is the issuer account, you are going to send 5 XLM from the Albedo wallet with the memo 'Activate Issuer Account'.
- In order to activate the Albedo wallet, which is the distributor account, you are going to send 10 XLM from the xBull wallet with the memo 'Activate Distributor Account'.

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Then, we are going to utilize the Stellar Laboratory and choose the 'Build Transaction' tab, in order to establish Trustline between the Distributor and the Issuer.
- Always make sure that you are working on the testnet of the Stellar Laboratory!



Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- As the first step, you are going to copy and paste the public key of the distributor account in the 'Source Account' field, which in this case is the public key of the Albedo account (Distributor).

The screenshot shows the 'Transaction Type' section with two buttons: 'Transaction' (highlighted in purple) and 'Fee Bump'. Below this is the 'Source Account' field, which contains the public key: `GDJDY7CBJ6AZ5KCSLFUAUNERNAWSX5KRLXIU0ZZMD4XR00KQJINZ5EKV4`. A note below the field states: 'If you don't have an account yet, you can create and fund a test net account with the [account creator](#).'

- Up next, click on 'Fetch next sequence number for account starting with '.....' and in the memo add 'ASSET1' as a text.

The screenshot shows the 'Transaction Sequence Number' field with the value `4803474243977217`. Below the field is a note: 'The transaction sequence number is usually one higher than current account sequence number.' and a purple button labeled 'Fetch next sequence number for account starting with "GDJDY7CBJ6"'. Below the button is the text 'Fetching from: <https://horizon-testnet.stellar.org>'. Below this is the 'Memo (optional)' section with four buttons: 'None', 'Text' (highlighted in purple), 'ID', 'Hash', and 'Return'. The 'Text' button is selected, and the memo field contains the text 'ASSET1'.

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Then, scroll down to the 'Operation Type' form and choose 'Change Trust'.

The screenshot shows a web interface for creating a Stellar network token. On the left, there is a sidebar with a box containing the number '1' and a purple 'duplicate' button. The main area contains a form with several fields: 'Operation Type' (with a question mark icon), 'Asset', 'Trust Limit (optional)', and 'Source Account (optional)'. A dropdown menu is open over the 'Operation Type' field, displaying a list of operations. The 'Change Trust' option is highlighted in blue. Other options in the list include 'Select operation type', 'Create Account', 'Payment', 'Path Payment Strict Send', 'Path Payment Strict Receive', 'Manage Sell Offer', 'Manage Buy Offer', 'Create Passive Sell Offer', 'Set Options', 'Allow Trust', 'Account Merge', 'Manage Data', 'Bump Sequence', 'Create Claimable Balance', 'Claim Claimable Balance', 'Begin Sponsoring Future Reserves', 'End Sponsoring Future Reserves', 'Revoke Sponsorship', and 'Clawback'. To the right of the form, there is a text input field labeled 'Liquidity pool shares' and another text input field containing the address '4UWDKHTCKADVS2AHF3UI2ZMO3DPUSM6Q4UG'.

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- In the 'Asset' field choose between 'Alphanumeric 4' or 'Alphanumeric 12'. This defines the possible length of your Asset Code.
- Then, enter your Asset Code, which is your token's name, in the corresponding field (e.g. STAC), and write it down.
- In the 'Issuer Account ID' field, copy and paste the public key of the issuer account, which, in this case, is the public key of the xBull wallet (issuer).

Asset	<div>Alphanumeric 4</div> <div>Alphanumeric 12</div> <div>Liquidity pool shares</div>
	<div>ASSET1</div> <div>GCFMDKGWQ3BTO4PBCGWMBYK2CV46CNVG3JK</div>

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- After that, enter the amount of tokens you want to generate into the 'Trust Limit' field (e.g. 150,000).

Trust Limit (optional)	<input type="text" value="150000"/>
	Leave empty to default to the max int64. Set to 0 to remove the trust line.

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Then, scroll down and click on ‘Sign in Transaction Signer’.

Success! Transaction Envelope XDR:

Network Passphrase:

Test SDF Network ; September 2015

Hash:

980d6281f213ef89cd2433deea698838e80a01645a0e99f602bca935c52bdac0

XDR:

AAAAAgAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAAGQAERC8AAAAAQAAAEAAAAAAAAAAAAAAAAAAAAAAAAAA
AAEAAAAAAAAABgAAAAJNWVRPS0VOAAAAAAAAAAAAAisGo1obDN3HhEazA4VoVeeE2ptpUDF4yygIG4HqI3FEAAAFdPveYAAAAAAAAAA
AA

In order for the transaction to make it into the ledger, a transaction must be successfully signed and submitted to the network. The laboratory provides the [Transaction Signer](#) for signing a transaction, and the [Post Transaction endpoint](#) for submitting one to the network.

[Sign in Transaction Signer](#)

[View in XDR Viewer](#)

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Now, copy and paste the secret key of the Distributor (the Albedo wallet) to the 'Add Signer' field.

Signatures ?

Add Signer	<input type="text" value="SCS4LAZEEBOKS3UUUYNB5ODNIRMFJZPL"/>
	<input type="text" value="Secret key (starting with S) or hash preimage (in hex)"/>
BIP Path	<input type="text" value="44'/148'/0'"/>
	<div><button>Sign with Ledger</button><button>Sign with Trezor</button></div>
	<p>NOTE: Trezor devices require upper time bounds to be set (non-zero), otherwise the signature will not be verified.</p>
Albedo	<div><button>Sign with Albedo</button></div>

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Afterwards, scroll down and click on ‘Submit in Transaction Submitter’.

Transaction signed!

1 signature(s) added; 1 signature(s) total

```
AAAAAgAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAAGQAERC8AAAAAQAAAAEAAAAAAAAAAAAAAAAAAAAAA  
AAAEAAAAAAAAABgAAAAJNWVRPS0V0AAAAAAAAAAAAisGo1obDN3HhEazA4VoVeeE2ptpUDF4yygrG4Hqr3FEAAAFdPveYAAAAAAAAAA  
ABMEt10gAAAEED56oH/iY2kDGTdgAo1uBUe6MVrfS8fjCEd18VeXI2i4PTz6LpWz9beHRV/xq2VWEvdWGALLXRSDP+vI0Io7zQ0
```

Now that this transaction is signed, you can submit it to the network. Horizon provides an endpoint called Post Transaction that will relay your transaction to the network and inform you of the result.

[Submit in Transaction Submitter](#)[View in XDR Viewer](#)[Wrap with Fee Bump](#)

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Then, click on ‘Submit Transaction’.

Input a base-64 encoded TransactionEnvelope:

```
AAAAAgAAAADSPHxBT4GegFJZaAoOkWgtK+1KFdFHa0z3Lx96MEt10gA  
AAGQAERC8AAAAABwAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAQAAA  
AZBU1NFVDEAAAAAAAAEAAAAAAAAABgAAAAJBU1NFVDEAAAAAAAAA  
AAAAisGo1obDN3HhEazA4VoVeeE2ptpUDF4yygrG4Hqr3FEAAAFdPveYA  
AAAAAAAAABMEt10gAAAEcvt7QVNZUIVG5XFCCHaIR1KFa2jzxcHW  
OHyzXjIUac+uifnc6bla1u1rGRKP9ZFBzouEDyCeL60FkxveEYUM
```

Submit Transaction

TransactionEnvelope: [envelopeTypeTx]

v1

tx

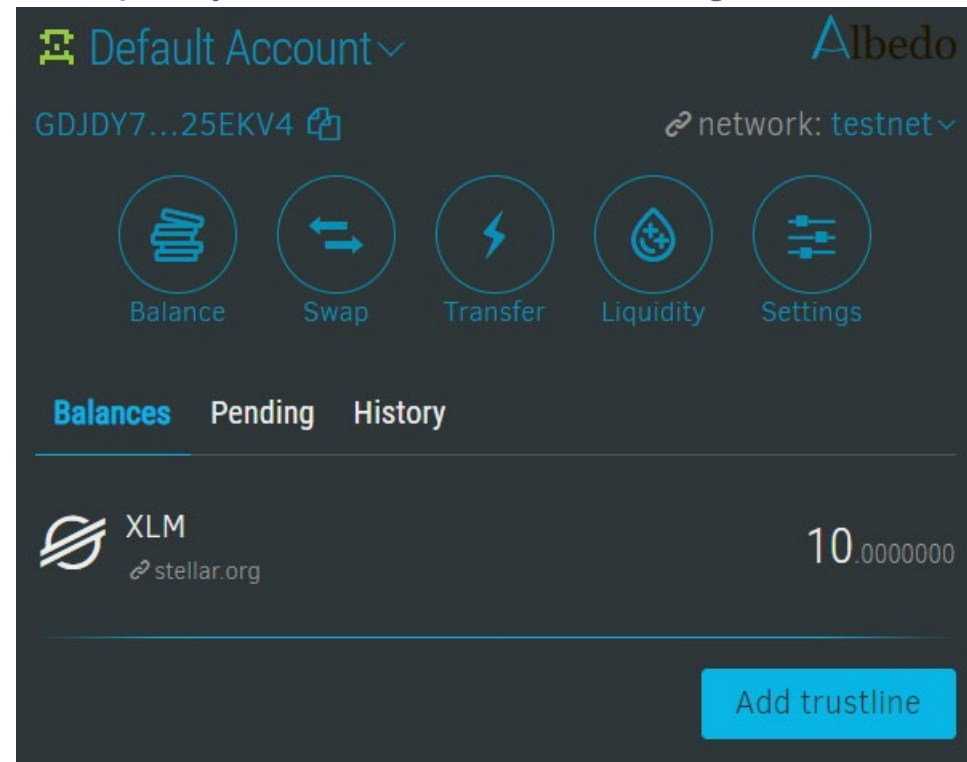
sourceAccount: [keyTypeEd25519]

ed25519: GDJDY7CBJ6AZ5KCSLFUAUNERNAWSX3KKCXIU022M64XR66RQJN25EKV4

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

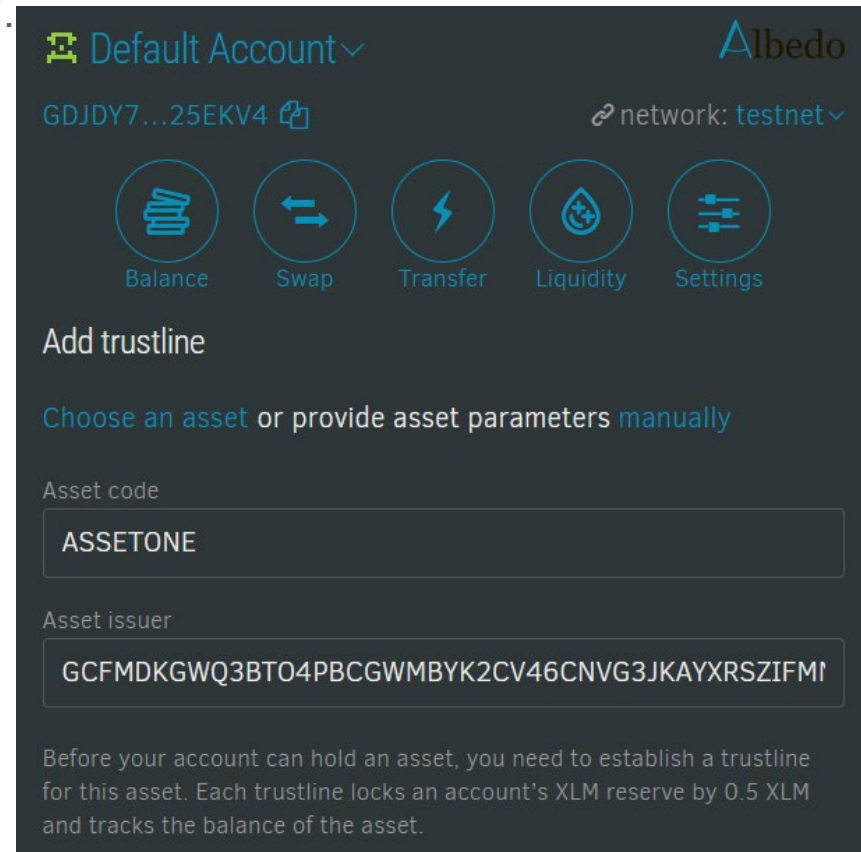
- As an **alternative** choice, instead of establishing a Trustline via the Stellar Laboratory, you can choose to establish a Trustline via the Albedo interface:
- In order to do so, you can open your Distributor wallet, e.g., the Albedo wallet, and click on 'Add Trustline'.



Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- Then, fill in the 'Asset code' and 'Asset issuer' fields with the same values you used in the Laboratory and click on 'Add trustline'.



The screenshot shows the Albedo interface for a 'Default Account' on the 'testnet' network. The account ID is 'GDJDY7...25EKV4'. Below the account information are five icons: Balance, Swap, Transfer, Liquidity, and Settings. The 'Add trustline' section is active, showing two input fields: 'Asset code' with the value 'ASSETONE' and 'Asset issuer' with the value 'GCFMDKGWQ3BTO4PBCGWMBYK2CV46CNVG3JKAYXRSZIFM'. A note at the bottom explains that establishing a trustline locks an account's XLM reserve by 0.5 XLM and tracks the balance of the asset.

Default Account ✓ Albedo

GDJDY7...25EKV4 network: testnet ✓

Balance Swap Transfer Liquidity Settings

Add trustline

Choose an asset or provide asset parameters manually

Asset code

ASSETONE

Asset issuer

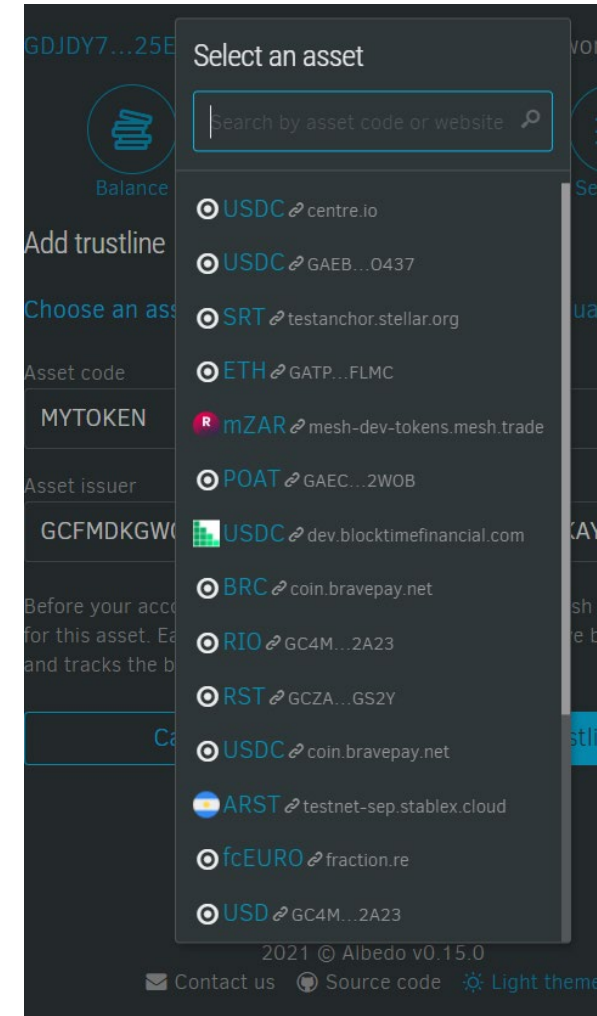
GCFMDKGWQ3BTO4PBCGWMBYK2CV46CNVG3JKAYXRSZIFM

Before your account can hold an asset, you need to establish a trustline for this asset. Each trustline locks an account's XLM reserve by 0.5 XLM and tracks the balance of the asset.

Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

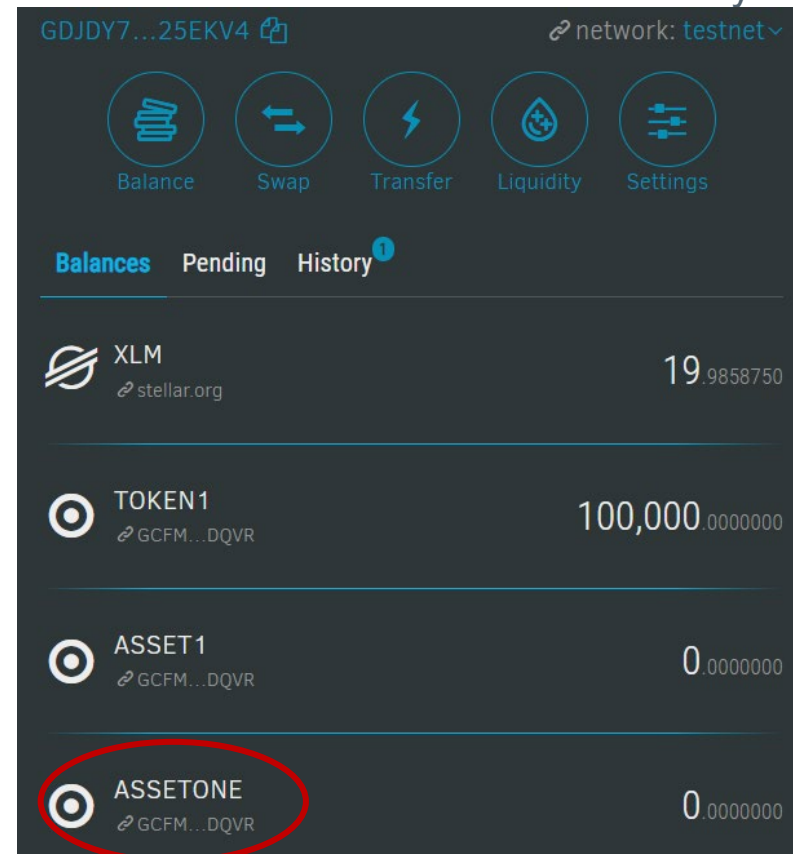
- Alternatively, you could click on 'Choose an asset' and select the asset that you prefer.
- However, in this example you will proceed with XLM.



Create a Stellar-network token

Exercise 3.1 – Establish a Trustline between the Distributor and the Issuer

- After you are requested to type your password in order to confirm the Trustline, you will be directed to your landing page and see that the Trustline has been successfully created.



Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- The first steps are pretty similar to the ones you followed on exercise 3.1.
- After you have cleared all previously entered data, head over to the 'Build a transaction' tab of the Stellar Laboratory again.
- However, now you are going to copy and paste the public key of the ISSUER account in the 'Source Account' field, which in this case is the public key of the xBull account (issuer) and click on 'Fetch next sequence number for account starting with '.....' and fill in the memo.

Source Account ?

GCFMDKGWQ3BTO4PBCGWMBYK2CV46C

If you don't have an account yet, you can create and fund a test net account with the [account creator](#).

Transaction Sequence Number ?

4547116236013577

The transaction sequence number is usually one higher than current account sequence number.

Fetch next sequence number for account starting with "GCFMDKGWQ3"

Fetching from: <https://horizon-testnet.stellar.org>

Base Fee ?

100

The [network base fee](#) is currently set to 100 stroops (0.00001 lumens). Based on current network activity, we suggest setting it to 100 stroops. Final transaction fee is equal to base fee times number of operations in this transaction.

Memo ?
(optional)

None

Text

ID

Hash

Return

ASSET1

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Then, scroll down to the 'Operation Type' form and select 'Payment'.

The screenshot shows a web interface for creating a Stellar network operation. On the left, there is a box with the number '1' and a 'duplicate' button. The main form has several fields: 'Operation Type' (with a dropdown menu open), 'Destination', 'Asset', 'Amount', and 'Source Account (optional)'. The dropdown menu for 'Operation Type' lists various options, with 'Payment' highlighted in blue. Below the form, there is a large blue button labeled '+ Add Operation'.

1

duplicate

Operation Type ?

Payment

Select operation type

Create Account

Payment

Path Payment Strict Send

Path Payment Strict Receive

Manage Sell Offer

Manage Buy Offer

Create Passive Sell Offer

Set Options

Change Trust

Allow Trust

Account Merge

Manage Data

Bump Sequence

Create Claimable Balance

Claim Claimable Balance

Begin Sponsoring Future Reserves

End Sponsoring Future Reserves

Revoke Sponsorship

Clawback

Destination

Asset

Amount

Source Account (optional)

+ Add Operation

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Then, paste the distributor's public key in the 'Destination' field and in the 'Asset' field choose between Alphanumeric 4 or Alphanumeric 12 as the length for your Asset Code which you picked earlier.
- After that, enter your Asset Code (e.g. MYTOKEN2) and the issuer's public key.
- Choose the amount of your token you want to send to the distributor (e.g. 150,000 for all token which you created earlier) in the 'Amount' field.

Destination	<input type="text" value="GDJDY7CBJ6AZ5KCSLFUAUNERNAWSX3KKCXI"/>		
Asset	native	Alphanumeric 4	Alphanumeric 12
	<input type="text" value="ASSET1"/>		
	<input type="text" value="GCFMDKGWQ3BTO4PBCGWMBYK2CV46CNVG3JH"/>		
Amount	<input type="text" value="5"/>		

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Next, scroll down to ‘Sign in Transaction Signer’ and click it.

Success! Transaction Envelope XDR:

Network Passphrase:

Test SDF Network ; September 2015

Hash:

11ca9c45a58a5c2930bcf80dfc7d8d9214c266af588b3dddb006c31302a93a33

XDR:

AAAAAgAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAA+gAERC8AAAAAgAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAEAAAAAAAAAAQAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAAAJNWVRPS0VOMgAAAAAAAAAAisGo1obDN3HhEa
zA4VoVeeE2ptpUDF4yygrG4Hqr3FEAAAFdPveYAAAAAAAAAAAAA

In order for the transaction to make it into the ledger, a transaction must be successfully signed and submitted to the network. The laboratory provides the [Transaction Signer](#) for signing a transaction, and the [Post Transaction endpoint](#) for submitting one to the network.

[Sign in Transaction Signer](#)

[View in XDR Viewer](#)

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Next, copy and paste the secret key of the Distributor (the Albedo wallet) to the “Add Signer” field.

Signatures ?

Add Signer	<input type="text" value="SBNBSJD5Z2TDCTXE5QPXBMMF5PPIPBYUSZL"/> <input type="text" value="Secret key (starting with S) or hash preimage (in hex)"/>
BIP Path	<input type="text" value="44'/148'/0'"/> <div><div>Sign with Ledger</div><div>Sign with Trezor</div></div> <p>NOTE: Trezor devices require upper time bounds to be set (non-zero), otherwise the signature will not be verified.</p>
Albedo	<div>Sign with Albedo</div>

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Once more, scroll down and click on 'Submit in Transaction Submitter' and then 'Submit Transaction'.

Transaction signed!

1 signature(s) added; 1 signature(s) total

```
AAAAAgAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAAGQAERC8AAAAAgAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAA  
AAAEAAAAAAAAAAQAAAADSPHxBT4GeqFJZaAo0kWgtK+1KFdFHa0z3Lx96MEt10gAAAAJNWVRPS0VOMgAAAAAAAAAAisGo1obDN3HhEa  
zA4VoVeeE2ptpUDF4yygIrG4HqIr3FEAAAFdPveYAAAAAAAAAAAAABMEt10gAAAEEntwOShSiyyspUxefNdDbUsQ5Ir1g1YJUwvgUwnqiRK  
hJwPT1sRcRd9SMnJs+DhK6fbp+wxcSPMIQZ9j5azLwH
```

Now that this transaction is signed, you can submit it to the network. Horizon provides an endpoint called Post Transaction that will relay your transaction to the network and inform you of the result.

[Submit in Transaction Submitter](#)[View in XDR Viewer](#)[Wrap with Fee Bump](#)

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Once more, scroll down and click on 'Submit in Transaction Submitter' and then 'Submit Transaction'.

Transaction submitted!

Transaction succeeded with 1 operation(s).

Hash:

e282881853d713c5abf10a8a3d7c10893bab7eff9e744c110519053efc9982b3

Ledger number:

1122902

Paging token:

4822827366625280

Result XDR:

AAAAAAAAAGQAAAAAAAAAAQAAAAAAAAABAAAAAAAAAA=

Result Meta XDR:

AAAAAgAAAAIAAADABEiVgAAAAAAAAAAisGo1obDN3HhEazA4VoVeeE2ptpUDF4yygrG4Hqr3FEAAAXMJ8MHAAQJ5QAAAAIAAAAAAA

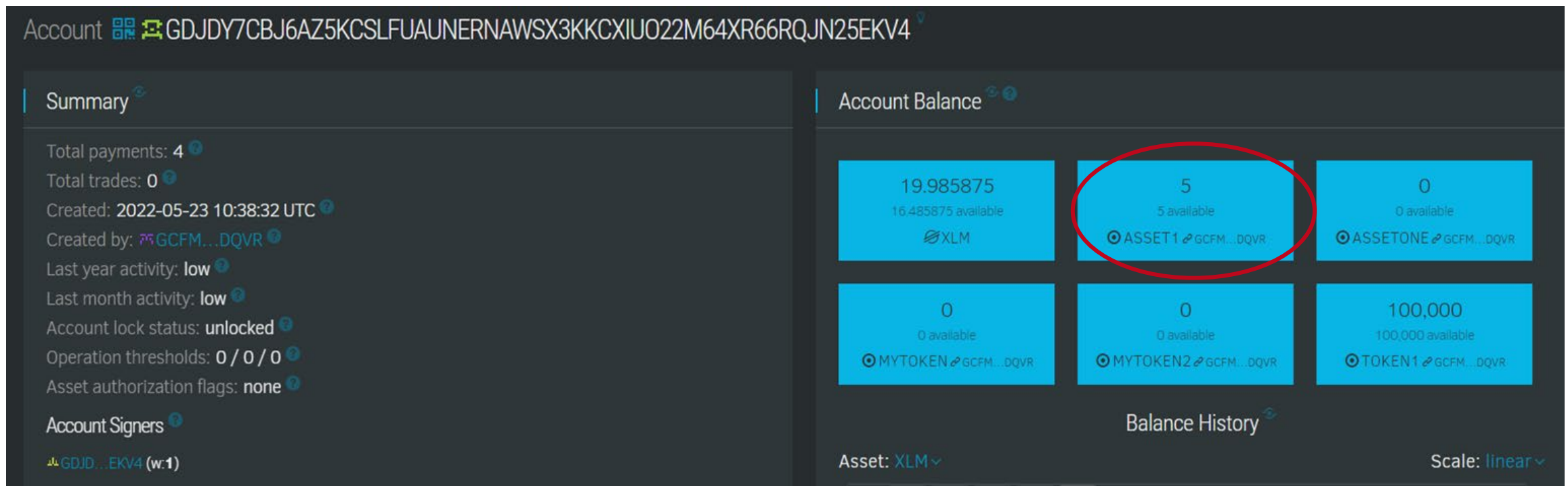
Fee Meta XDR:

AAAAAgAAAAIAESI0AAAAAAAAAAACKWajWhsM3ceERrMDhWhV54Tam21QMXjLKCsbgeqvcUQAAABcwnwyAABAn1AAAAAgAAAAAAAAAAAA

Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

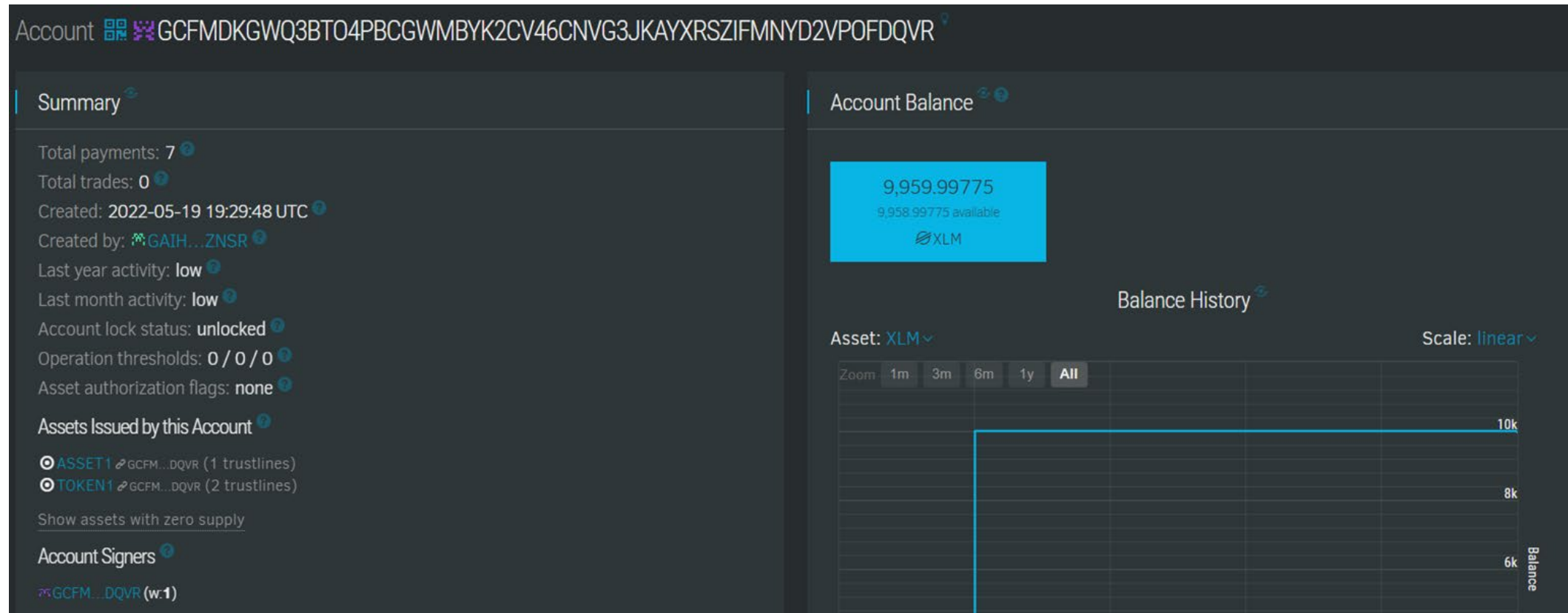
- Give it a minute, then check both accounts on Stellar.expert. Both accounts should present the amount of XLM, plus the amount of your new token (e.g. ASSET1) on the DISTRIBUTOR account.
- Stellar.expert is a tool provided by Stellar and it serves as an explorer for every Stellar account.
- You can search each account using its public key and gain access to many useful analytics.



Create a Stellar-network token

Exercise 3.2 – Send a new token from the Issuer to the Distributor

- Make sure you are on the testnet!



Create a Stellar-network token

Exercise 3.2 – Issue and distribute your own custom asset

- After, successfully completing all the tasks above, take the same steps in order to issue and distribute another custom asset of your choice, e.g., ASSET2
- Once you are done, copy the transaction ID of the transaction of the 1st custom asset to the corresponding moodle field and repeat for the 2nd custom asset as well.
- Remember, the transaction ID is the hash number of the transaction.

Questions?

Contact Us: [Stellar Developers Discord](#)

Twitter: [@StellarOrg](#)

