C5-Live

October 5, 2023

[1]:
```
import os
import sys
import math
import time
import IPython
import logging
import warnings
import datetime
import pandas as pd
import numpy  as np
import webbrowser
# import scipy  as sp
import pprint as pp
from random   import randrange
from notebook import notebookapp
from IPython.display import display
from IPython.core.display import HTML
display(HTML("<style>.container { width:98% !important; }</style>"))
sys.path.insert(1, os.path.join(sys.path[0], '{}/lib/'.format(os.
  ↪environ['BaseDir'])))
from corda5Interface import Corda5
%matplotlib inline
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
pd.set_option('display.max_colwidth', None)

# def set_in_master_list(k, v): replace_maters_list[k] = v
# https://localhost:8888/api/v1/flow/{{Dave Hash}}/30020

apiServer='https://localhost:8888/api/v1'
apiServerAuth=('admin', 'admin')
package = 'com.r3.developers.configurableInstrument'
replace_maters_list = { 'API_SERVER': apiServer, 'PACKAGE': package}

nodes = {
        'Authority' : 'CN=Authority, OU=Test Dept, O=R3, L=London, C=GB',
        'Bob'       : 'CN=Bob, OU=Test Dept, O=R3, L=London, C=GB',
        'Charlie'   : 'CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB',
        'Alice'     : 'CN=Alice, OU=Test Dept, O=R3, L=London, C=GB',
#         'NotaryRep1': 'CN=NotaryRep1, OU=Test Dept, O=R3, L=London, C=GB',
        'Dave'      : 'CN=Dave, OU=Test Dept, O=R3, L=London, C=GB',
        }


h = Corda5(level=logging.DEBUG, nodes=nodes, apiServer=apiServer,␣
 ↪apiServerAuth=apiServerAuth, version=2)
# h = Corda5(level=logging.INFO, nodes=nodes, apiServer=apiServer,␣
 ↪apiServerAuth=apiServerAuth, version=2)

log = h.log
log.info("Base directory {}".format(os.environ['BaseDir']))
pp.pprint(h.x500)
```

```
<IPython.core.display.HTML object>
```

```
 05-10 15:12:35 | D | 200
 05-10 15:12:35 | D | {'virtualNodes': [{'holdingIdentity':
{'x500Name': 'CN=Dave, OU=Test Dept, O=R3, L=London, C=GB', 'groupId':
'62578ae6-e689-4b0f-9edf-1c7e232d503b', 'shortHash': 'DEA38FB8988C', 'fullHash':
'DEA38FB8988C6C41B42E64F691B73536A57C2F442C178FB4F0FE6F7EAE9F634E'},
'cpiIdentifier': {'cpiName': 'MyCorDapp', 'cpiVersion': '1.0-SNAPSHOT',
'signerSummaryHash':
'SHA-256:A3C6D650D91B1A9CC4DA9B93E83B269FBD2CF771C17A33BCE30C440180D6E3C5'},
'vaultDdlConnectionId': 'b540c2a8-ad36-4ed4-b351-641f4d377714',
'vaultDmlConnectionId': '0806476d-728f-442d-9e7a-11e8bdd1ca2e',
'cryptoDdlConnectionId': '23957035-7174-4c6c-be35-fe11f5b60be4',
'cryptoDmlConnectionId': '7a332430-6dbc-42c8-a7b3-9fdbe9dbc354',
'uniquenessDdlConnectionId': '4850d46a-1f7f-4a26-8a90-af72f5550907',
'uniquenessDmlConnectionId': 'e70255c0-061c-45d0-85b0-0ac435608f61',
'hsmConnectionId': 'null', 'flowP2pOperationalStatus': 'ACTIVE',
'flowStartOperationalStatus': 'ACTIVE', 'flowOperationalStatus': 'ACTIVE',
'vaultDbOperationalStatus': 'ACTIVE', 'operationInProgress': None,
'externalMessagingRouteConfiguration': None}, {'holdingIdentity': {'x500Name':
'CN=Bob, OU=Test Dept, O=R3, L=London, C=GB', 'groupId':
'62578ae6-e689-4b0f-9edf-1c7e232d503b', 'shortHash': '5945EE77CCB7', 'fullHash':
'5945EE77CCB70556466C4DDA115DCFFB1E2235BF21BAC8C72529019532E9DFCE'},
'cpiIdentifier': {'cpiName': 'MyCorDapp', 'cpiVersion': '1.0-SNAPSHOT',
'signerSummaryHash':
'SHA-256:A3C6D650D91B1A9CC4DA9B93E83B269FBD2CF771C17A33BCE30C440180D6E3C5'},
'vaultDdlConnectionId': '3e3bc84e-f830-40ce-88be-35280c164627',
'vaultDmlConnectionId': '6c125850-7f68-476e-9589-4d1c80c65981',
'cryptoDdlConnectionId': 'edab8f83-1d78-4e81-b470-e3caa3409409',
'cryptoDmlConnectionId': '6e5b37cd-40ac-4d28-a0e4-c7b3304f959d',
'uniquenessDdlConnectionId': '88f0d37d-4e6f-4a09-a4bf-4bb370aec4bc',
'uniquenessDmlConnectionId': '0792c07a-f1fd-48e1-be73-19c3e03dd1a1',
'hsmConnectionId': 'null', 'flowP2pOperationalStatus': 'ACTIVE',
'flowStartOperationalStatus': 'ACTIVE', 'flowOperationalStatus': 'ACTIVE',
'vaultDbOperationalStatus': 'ACTIVE', 'operationInProgress': None,
'externalMessagingRouteConfiguration': None}, {'holdingIdentity': {'x500Name':
'CN=Authority, OU=Test Dept, O=R3, L=London, C=GB', 'groupId':
'62578ae6-e689-4b0f-9edf-1c7e232d503b', 'shortHash': '9FA89008DEEE', 'fullHash':
'9FA89008DEEED650B010B98876EE58E47A0EB05732B61574419CE84E1B3725C4'},
 05-10 15:12:35 | D | CN=Dave, OU=Test Dept, O=R3, L=London, C=GB
DEA38FB8988C
 05-10 15:12:35 | D | CN=Bob, OU=Test Dept, O=R3, L=London, C=GB
5945EE77CCB7
 05-10 15:12:35 | D | CN=Authority, OU=Test Dept, O=R3, L=London,
C=GB 9FA89008DEEE
 05-10 15:12:35 | D | CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
BAA7F076B971
 05-10 15:12:35 | D | CN=NotaryRep1, OU=Test Dept, O=R3, L=London,
C=GB 57ECB8DA3FFC
 05-10 15:12:35 | D | CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB 0AA37CC89C79
 05-10 15:12:35 | I | Base directory
/Users/Manoj.Gupta/git/c5_configurableLedgerAsset/python3
```

```
{'CN=Alice, OU=Test Dept, O=R3, L=London, C=GB': 'BAA7F076B971',
 'CN=Authority, OU=Test Dept, O=R3, L=London, C=GB': '9FA89008DEEE',
 'CN=Bob, OU=Test Dept, O=R3, L=London, C=GB': '5945EE77CCB7',
 'CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB': '0AA37CC89C79',
 'CN=Dave, OU=Test Dept, O=R3, L=London, C=GB': 'DEA38FB8988C',
 'CN=NotaryRep1, OU=Test Dept, O=R3, L=London, C=GB': '57ECB8DA3FFC'}
```

[3]:
```
#␣
 ↪===========================================================================================
# Government Bond with defined coupon payments
#␣
 ↪===========================================================================================
h.message("Government Bond with defined coupon payments")

(req_id, response, uuid_bond) = h.issue('Authority', 'Alice', 'Government Bond␣
 ↪2024',
        quantity=10,
        transferable=True,
        expiry=None,
        verifiable=True,
        attributes={'payments': "['10Jun2023', '10Sep2023', '10Dec2023',␣
 ↪'10Mar2024']"})

h.message([req_id, response, uuid_bond])


# #␣
 ↪===========================================================================================
# # Digital currency (eRupee)
# #␣
 ↪==========================================================================================
# h.message("Digital currency (eRupee)", h=4)
# (req_id, response, uuid_erupee) = h.issue('Authority', 'Alice', 'eRupee',
#         quantity=100001,
#         transferable=True,
#         expiry=None,
#         verifiable=True
#        )

# h.message([req_id, response, uuid_erupee])

#␣
 ↪==========================================================================================
# Fiat currency with unique number in digital form.
#␣
 ↪==========================================================================================
h.message("Fiat currency with unique number in digital form.")
(req_id, response, uuid_fiat) = h.issue('Authority', 'Alice', 'Digital Currency␣
 ↪INR 500 Note',
        quantity=None,
        transferable=True,
        expiry=None,
        verifiable=True,
        attributes={'number': 'OMV 336048', 'year': '1981'})

h.message([req_id, response, uuid_fiat])

#␣
 ↪==========================================================================================
# Amazon Vouchers
#␣
 ↪==========================================================================================
h.message("Amazon Vouchers")
(req_id, response, uuid_amazon) = h.issue('Authority', 'Alice', 'Amazon␣
 ↪Voucher',
        quantity=200,
        transferable=False,
        expiry=None,
        verifiable=True,
        attributes={'currency': 'INR'})
h.message([req_id, response, uuid_amazon])

#␣
 ↪==========================================================================================
# Graduation certificate
#␣
 ↪=============================================================================
h.message("Graduation certificate")
(req_id, response, uuid_cert) = h.issue('Authority', 'Charlie', 'BE Computer␣
 ↪Engineering', quantity=None, transferable=False, expiry=None,␣
 ↪verifiable=True,
        attributes={
            'university': 'University of Mumbai',
            'year': '2023',
            'score': '450/500',
            'major subject': "['Robotics', 'AI', 'DSP']"})
h.message([req_id, response, uuid_cert])
```

```
<IPython.core.display.HTML object>

 05-10 14:55:03 | I | Issuing instrument='Government Bond 2024' to

'Alice'
 05-10 14:55:03 | I | Running action 'Issue' on

'Authority/9FA89008DEEE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 14:55:03 | D | Request type=post to

url='https://localhost:8888/api/v1/flow/9FA89008DEEE'
 05-10 14:55:03 | D | Original Request = {'clientRequestId':

'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Issue',

'requestBody': {'name': 'Government Bond 2024', 'to': 'CN=Alice, OU=Test Dept,

O=R3, L=London, C=GB', 'transferable': 'true', 'verifiable': 'true', 'quantity':

10, 'attributes': {'payments': "['10Jun2023', '10Sep2023', '10Dec2023',

'10Mar2024']"}}}
 05-10 14:55:03 | D | {'clientRequestId':

'36552.1696497903.553813', 'flowClassName':

'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':

{'name': 'Government Bond 2024', 'to': 'CN=Alice, OU=Test Dept, O=R3, L=London,

C=GB', 'transferable': 'true', 'verifiable': 'true', 'quantity': 10,

'attributes': {'payments': "['10Jun2023', '10Sep2023', '10Dec2023',

'10Mar2024']"}}}
```

```
 05-10 14:55:03 | D | After filling values = {'clientRequestId':
'36552.1696497903.553813', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Government Bond 2024', 'to': 'CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB', 'transferable': 'true', 'verifiable': 'true', 'quantity': 10,
'attributes': {'payments': "['10Jun2023', '10Sep2023', '10Dec2023',
'10Mar2024']"}}}
 05-10 14:55:03 | D | 202
 05-10 14:55:03 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:25:03.603Z'}
 05-10 14:55:07 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:07 | D | 200
 05-10 14:55:07 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:11 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:11 | D | 200
 05-10 14:55:11 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:15 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:15 | D | 200
 05-10 14:55:15 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:19 | D | 200
 05-10 14:55:19 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:23 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:23 | D | 200
 05-10 14:55:23 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:27 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:27 | D | 200
 05-10 14:55:27 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:31 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:31 | D | 200
 05-10 14:55:31 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:35 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:36 | D | 200
 05-10 14:55:36 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:40 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:40 | D | 200
 05-10 14:55:40 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:04.634Z'}
 05-10 14:55:44 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497903.553813'
 05-10 14:55:44 | D | 200
 05-10 14:55:44 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497903.553813', 'flowId':
'24d9a8a4-eeb1-4269-a40a-7d891dd44d0d', 'flowStatus': 'COMPLETED', 'flowResult':
'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d', 'flowError': None, 'timestamp':
'2023-10-05T09:25:40.881Z'}
 05-10 14:55:44 | I | d0f0dc0e-b67c-4d27-b335-a091a5ed7c7d
 05-10 14:55:44 | I | Time taken = 0:00:40.543460
36552.1696497903.553813<Response [200]>d0f0dc0e-b67c-4d27-b335-a091a5ed7c7d
<IPython.core.display.HTML object>
 05-10 14:55:44 | I | Issuing instrument='Digital Currency INR 500
Note' to 'Alice'
 05-10 14:55:44 | I | Running action 'Issue' on
'Authority/9FA89008DEEE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 14:55:44 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/9FA89008DEEE'
 05-10 14:55:44 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Issue',
'requestBody': {'name': 'Digital Currency INR 500 Note', 'to': 'CN=Alice,
OU=Test Dept, O=R3, L=London, C=GB', 'transferable': 'true', 'verifiable':
'true', 'attributes': {'number': 'OMV 336048', 'year': '1981'}}}
 05-10 14:55:44 | D | {'clientRequestId':
'36552.1696497944.1001952', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Digital Currency INR 500 Note', 'to': 'CN=Alice, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'true', 'verifiable': 'true', 'attributes':
{'number': 'OMV 336048', 'year': '1981'}}}
```

```
 05-10 14:55:44 | D | After filling values = {'clientRequestId':
'36552.1696497944.1001952', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Digital Currency INR 500 Note', 'to': 'CN=Alice, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'true', 'verifiable': 'true', 'attributes':
{'number': 'OMV 336048', 'year': '1981'}}}
 05-10 14:55:44 | D | 202
 05-10 14:55:44 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:25:44.118Z'}
 05-10 14:55:48 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:55:48 | D | 200
 05-10 14:55:48 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:55:52 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:55:52 | D | 200
 05-10 14:55:52 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:55:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:55:56 | D | 200
 05-10 14:55:56 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:56:00 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:56:00 | D | 200
 05-10 14:56:00 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:56:04 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:56:04 | D | 200
 05-10 14:56:04 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:56:08 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:56:08 | D | 200
 05-10 14:56:08 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:56:12 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:56:12 | D | 200
 05-10 14:56:12 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:25:44.556Z'}
 05-10 14:56:16 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497944.1001952'
 05-10 14:56:16 | D | 200
 05-10 14:56:16 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497944.1001952', 'flowId':
'd835f426-21e4-42b5-b219-8405b3785318', 'flowStatus': 'COMPLETED', 'flowResult':
'28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7', 'flowError': None, 'timestamp':
'2023-10-05T09:26:13.657Z'}
 05-10 14:56:16 | I | 28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7
 05-10 14:56:16 | I | Time taken = 0:00:32.421132
36552.1696497944.1001952<Response [200]>28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7
```

```
<IPython.core.display.HTML object>
 05-10 14:56:16 | I | Issuing instrument='Amazon Voucher' to
'Alice'
 05-10 14:56:16 | I | Running action 'Issue' on
'Authority/9FA89008DEEE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 14:56:16 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/9FA89008DEEE'
 05-10 14:56:16 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Issue',
'requestBody': {'name': 'Amazon Voucher', 'to': 'CN=Alice, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'false', 'verifiable': 'true', 'quantity': 200,
'attributes': {'currency': 'INR'}}}
 05-10 14:56:16 | D | {'clientRequestId':
'36552.1696497976.524884', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Amazon Voucher', 'to': 'CN=Alice, OU=Test Dept, O=R3, L=London, C=GB',
'transferable': 'false', 'verifiable': 'true', 'quantity': 200, 'attributes':
{'currency': 'INR'}}}
 05-10 14:56:16 | D | After filling values = {'clientRequestId':
'36552.1696497976.524884', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Amazon Voucher', 'to': 'CN=Alice, OU=Test Dept, O=R3, L=London, C=GB',
'transferable': 'false', 'verifiable': 'true', 'quantity': 200, 'attributes':
{'currency': 'INR'}}}
 05-10 14:56:16 | D | 202
 05-10 14:56:16 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:26:16.545Z'}
 05-10 14:56:20 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:20 | D | 200
 05-10 14:56:20 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:24 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:24 | D | 200
 05-10 14:56:24 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:28 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:28 | D | 200
 05-10 14:56:28 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:32 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:32 | D | 200
 05-10 14:56:32 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:36 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:36 | D | 200
 05-10 14:56:36 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:40 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:40 | D | 200
 05-10 14:56:40 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:44 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:44 | D | 200
 05-10 14:56:44 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:16.903Z'}
 05-10 14:56:48 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696497976.524884'
 05-10 14:56:48 | D | 200
 05-10 14:56:48 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696497976.524884', 'flowId':
'776f4778-29bf-4e66-8706-ab2789ccabd3', 'flowStatus': 'COMPLETED', 'flowResult':
'0fb152f0-805c-4d63-a889-3485c92df378', 'flowError': None, 'timestamp':
'2023-10-05T09:26:45.069Z'}
 05-10 14:56:48 | I | 0fb152f0-805c-4d63-a889-3485c92df378
 05-10 14:56:48 | I | Time taken = 0:00:32.372210
36552.1696497976.524884<Response [200]>0fb152f0-805c-4d63-a889-3485c92df378
<IPython.core.display.HTML object>
 05-10 14:56:48 | I | Issuing instrument='BE Computer Engineering'
to 'Charlie'
 05-10 14:56:48 | I | Running action 'Issue' on
'Authority/9FA89008DEEE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 14:56:48 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/9FA89008DEEE'
 05-10 14:56:48 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Issue',
'requestBody': {'name': 'BE Computer Engineering', 'to': 'CN=Charlie, OU=Test
Dept, O=R3, L=London, C=GB', 'transferable': 'false', 'verifiable': 'true',
'attributes': {'university': 'University of Mumbai', 'year': '2023', 'score':
'450/500', 'major subject': "['Robotics', 'AI', 'DSP']"}}}
```

```
 05-10 14:56:48 | D | {'clientRequestId':
'36552.1696498008.9060092', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'BE Computer Engineering', 'to': 'CN=Charlie, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'false', 'verifiable': 'true', 'attributes':
{'university': 'University of Mumbai', 'year': '2023', 'score': '450/500',
'major subject': "['Robotics', 'AI', 'DSP']"}}}
 05-10 14:56:48 | D | After filling values = {'clientRequestId':
'36552.1696498008.9060092', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'BE Computer Engineering', 'to': 'CN=Charlie, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'false', 'verifiable': 'true', 'attributes':
{'university': 'University of Mumbai', 'year': '2023', 'score': '450/500',
'major subject': "['Robotics', 'AI', 'DSP']"}}}
 05-10 14:56:48 | D | 202
 05-10 14:56:48 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:26:48.929Z'}
 05-10 14:56:52 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:56:52 | D | 200
 05-10 14:56:52 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:56:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:56:57 | D | 200
 05-10 14:56:57 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:01 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
```

```
 05-10 14:57:01 | D | 200
 05-10 14:57:01 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:05 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:57:05 | D | 200
 05-10 14:57:05 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:09 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:57:09 | D | 200
 05-10 14:57:09 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:13 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:57:13 | D | 200
 05-10 14:57:13 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:17 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:57:17 | D | 200
 05-10 14:57:17 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:26:49.258Z'}
 05-10 14:57:21 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/36552.1696498008.9060092'
 05-10 14:57:21 | D | 200
 05-10 14:57:21 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '36552.1696498008.9060092', 'flowId':
'1033f8ad-81bd-42db-a2b9-af1ecc02b9ae', 'flowStatus': 'COMPLETED', 'flowResult':
'9eaa5900-59f8-40d7-a25c-c89d146b15f9', 'flowError': None, 'timestamp':
'2023-10-05T09:27:20.609Z'}
 05-10 14:57:21 | I | 9eaa5900-59f8-40d7-a25c-c89d146b15f9
 05-10 14:57:21 | I | Time taken = 0:00:32.425330
```

```
36552.1696498008.9060092<Response [200]>9eaa5900-59f8-40d7-a25c-c89d146b15f9
```

[4]:

```
h.message("Transfer of Bond (expected success)")
(req_id, response, uuid_new_bond) = h.transfer(uuid_bond, 'Alice', 'Charlie',␣
 ↪quantity=4)


h.message("Transfer of Fiat Currency (expected success)")
(req_id, response, uuid_new) = h.transfer(uuid_fiat, 'Alice', 'Charlie')


h.message("Transfer of Amazon Voucher (expected failure)")
(req_id, response, uuid_new) = h.transfer(uuid_amazon, 'Alice', 'Charlie',␣
 ↪quantity=4)


h.message("Transfer of Graduation certificate (expected failure)")
(req_id, response, uuid_new) = h.transfer(uuid_cert, 'Charlie', 'Alice')
```

```
<IPython.core.display.HTML object>
```

```
 05-10 15:00:58 | I | Tranfering instrument id
'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d' to 'Charlie'
 05-10 15:00:58 | I | Running action 'Transfer' on
'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:00:58 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 05-10 15:00:58 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Transfer',
'requestBody': {'id': '__TRANSFER_ASSET_ID__', 'to': '__TRANSFER_TO__',
'quantity': 4}}
```

```
 05-10 15:00:58 | D | {'clientRequestId':
'36552.1696498258.153019', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': 'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB', 'quantity': 4}}
 05-10 15:00:58 | D | After filling values = {'clientRequestId':
'36552.1696498258.153019', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': 'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB', 'quantity': 4}}
 05-10 15:00:58 | D | 202
 05-10 15:00:58 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:30:58.187Z'}
 05-10 15:01:02 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:02 | D | 200
 05-10 15:01:02 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:06 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:06 | D | 200
 05-10 15:01:06 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:10 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:10 | D | 200
 05-10 15:01:10 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
```

```
 05-10 15:01:14 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:14 | D | 200
 05-10 15:01:14 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:18 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:18 | D | 200
 05-10 15:01:18 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:22 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:22 | D | 200
 05-10 15:01:22 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:26 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:26 | D | 200
 05-10 15:01:26 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
 05-10 15:01:30 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:30 | D | 200
 05-10 15:01:30 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:30:58.422Z'}
```

```
 05-10 15:01:34 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498258.153019'
 05-10 15:01:34 | D | 200
 05-10 15:01:34 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498258.153019', 'flowId':
'48805f8a-f674-4232-bed9-efe2c33e6661', 'flowStatus': 'COMPLETED', 'flowResult':
'1c3d8f12-6e82-49a8-b0a5-4b6904c7a634', 'flowError': None, 'timestamp':
'2023-10-05T09:31:33.753Z'}
 05-10 15:01:34 | I | 1c3d8f12-6e82-49a8-b0a5-4b6904c7a634
 05-10 15:01:34 | I | Time taken = 0:00:36.439116
<IPython.core.display.HTML object>
 05-10 15:01:34 | I | Tranfering instrument id
'28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7' to 'Charlie'
 05-10 15:01:34 | I | Running action 'Transfer' on
'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:01:34 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 05-10 15:01:34 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Transfer',
'requestBody': {'id': '__TRANSFER_ASSET_ID__', 'to': '__TRANSFER_TO__'}}
 05-10 15:01:34 | D | {'clientRequestId':
'36552.1696498294.601697', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB'}}
 05-10 15:01:34 | D | After filling values = {'clientRequestId':
'36552.1696498294.601697', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB'}}
 05-10 15:01:34 | D | 202
 05-10 15:01:34 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:31:34.622Z'}
```

```
 05-10 15:01:38 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:38 | D | 200
 05-10 15:01:38 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:01:42 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:42 | D | 200
 05-10 15:01:42 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:01:46 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:46 | D | 200
 05-10 15:01:46 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:01:50 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:50 | D | 200
 05-10 15:01:50 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:01:54 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:54 | D | 200
 05-10 15:01:54 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
```

```
 05-10 15:01:58 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:01:58 | D | 200
 05-10 15:01:58 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:02:02 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:02:02 | D | 200
 05-10 15:02:02 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:02:06 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:02:07 | D | 200
 05-10 15:02:07 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:02:11 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:02:11 | D | 200
 05-10 15:02:11 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:31:35.056Z'}
 05-10 15:02:15 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498294.601697'
 05-10 15:02:15 | D | 200
 05-10 15:02:15 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498294.601697', 'flowId':
'aa4672f9-2c73-47e0-963f-5168c59ba2ef', 'flowStatus': 'COMPLETED', 'flowResult':
'28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7', 'flowError': None, 'timestamp':
'2023-10-05T09:32:12.884Z'}
```

```
 05-10 15:02:15 | I | 28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7
 05-10 15:02:15 | I | Time taken = 0:00:40.501330
```

```
<IPython.core.display.HTML object>
```

```
 05-10 15:02:15 | I | Tranfering instrument id
'0fb152f0-805c-4d63-a889-3485c92df378' to 'Charlie'
 05-10 15:02:15 | I | Running action 'Transfer' on
'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:02:15 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 05-10 15:02:15 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Transfer',
'requestBody': {'id': '__TRANSFER_ASSET_ID__', 'to': '__TRANSFER_TO__',
'quantity': 4}}
 05-10 15:02:15 | D | {'clientRequestId':
'36552.1696498335.108681', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '0fb152f0-805c-4d63-a889-3485c92df378', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB', 'quantity': 4}}
 05-10 15:02:15 | D | After filling values = {'clientRequestId':
'36552.1696498335.108681', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '0fb152f0-805c-4d63-a889-3485c92df378', 'to': 'CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB', 'quantity': 4}}
 05-10 15:02:15 | D | 202
 05-10 15:02:15 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498335.108681', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:32:15.134Z'}
 05-10 15:02:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498335.108681'
 05-10 15:02:19 | D | 200
 05-10 15:02:19 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498335.108681', 'flowId':
'0eee4ebc-1b7d-4281-bf5d-593c42463725', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:15.374Z'}
```

```
 05-10 15:02:23 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498335.108681'
 05-10 15:02:23 | D | 200
 05-10 15:02:23 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498335.108681', 'flowId':
'0eee4ebc-1b7d-4281-bf5d-593c42463725', 'flowStatus': 'FAILED', 'flowResult':
None, 'flowError': {'type': 'FLOW_FAILED', 'message':
'net.corda.v5.ledger.utxo.ContractVerificationException: Verification of ledger
transaction with ID
SHA-256D:6A6D36117B6E2993067AB875F0E93B8DEF6B3B7BFC401507C30FA872ED5CAFE0
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:6A6D36117B6E2993067AB875F0E93B8DEF6B3B7BFC401507C30FA872ED5CAFE0.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'}, 'timestamp':
'2023-10-05T09:32:18.618Z'}
 05-10 15:02:23 | I | {'type': 'FLOW_FAILED', 'message':
'net.corda.v5.ledger.utxo.ContractVerificationException: Verification of ledger
transaction with ID
SHA-256D:6A6D36117B6E2993067AB875F0E93B8DEF6B3B7BFC401507C30FA872ED5CAFE0
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:6A6D36117B6E2993067AB875F0E93B8DEF6B3B7BFC401507C30FA872ED5CAFE0.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'}
 05-10 15:02:23 | I | Time taken = 0:00:08.152318
<IPython.core.display.HTML object>
 05-10 15:02:23 | I | Tranfering instrument id
'9eaa5900-59f8-40d7-a25c-c89d146b15f9' to 'Alice'
 05-10 15:02:23 | I | Running action 'Transfer' on
'Charlie/0AA37CC89C79/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 05-10 15:02:23 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/0AA37CC89C79'
 05-10 15:02:23 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Transfer',
'requestBody': {'id': '__TRANSFER_ASSET_ID__', 'to': '__TRANSFER_TO__'}}
 05-10 15:02:23 | D | {'clientRequestId':
'36552.1696498343.2683609', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '9eaa5900-59f8-40d7-a25c-c89d146b15f9', 'to': 'CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB'}}
 05-10 15:02:23 | D | After filling values = {'clientRequestId':
'36552.1696498343.2683609', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Transfer', 'requestBody':
{'id': '9eaa5900-59f8-40d7-a25c-c89d146b15f9', 'to': 'CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB'}}
 05-10 15:02:23 | D | 202
 05-10 15:02:23 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498343.2683609', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:32:23.296Z'}
 05-10 15:02:27 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498343.2683609'
 05-10 15:02:27 | D | 200
 05-10 15:02:27 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498343.2683609', 'flowId':
'ef796277-431a-42b9-8a7b-14c5ef39774b', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:23.581Z'}
 05-10 15:02:31 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498343.2683609'
 05-10 15:02:31 | D | 200
 05-10 15:02:31 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498343.2683609', 'flowId':
'ef796277-431a-42b9-8a7b-14c5ef39774b', 'flowStatus': 'FAILED', 'flowResult':
None, 'flowError': {'type': 'FLOW_FAILED', 'message':
'net.corda.v5.ledger.utxo.ContractVerificationException: Verification of ledger
transaction with ID
SHA-256D:18F7EA6D657B5A3BC9EE0A61CD7D15DFDED46CBBC1A8E0515E7EBFB974A666DD
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:18F7EA6D657B5A3BC9EE0A61CD7D15DFDED46CBBC1A8E0515E7EBFB974A666DD.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'}, 'timestamp':
'2023-10-05T09:32:27.611Z'}
 05-10 15:02:31 | I | {'type': 'FLOW_FAILED', 'message':
'net.corda.v5.ledger.utxo.ContractVerificationException: Verification of ledger
transaction with ID
SHA-256D:18F7EA6D657B5A3BC9EE0A61CD7D15DFDED46CBBC1A8E0515E7EBFB974A666DD
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:18F7EA6D657B5A3BC9EE0A61CD7D15DFDED46CBBC1A8E0515E7EBFB974A666DD.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'}
 05-10 15:02:31 | I | Time taken = 0:00:08.136343
```

[10]:
```
# (uuid_bond, uuid_new_bond, uuid_fiat, uuid_amazon, uuid_cert)
```

[4]:
```
h.message("Query from Alice")
df  = h.query('Alice')
tdf = df[(df.id == uuid_bond) | (df.id == uuid_new_bond) | (df.id == uuid_fiat)␣
 ↪| ( df.id == uuid_amazon ) | (df.id == uuid_cert)]
display(tdf)


h.message("Query from Charlie")
```

```
df  = h.query('Charlie')
tdf = df[(df.id == uuid_bond) | (df.id == uuid_new_bond) | (df.id == uuid_fiat)␣
 ↪| ( df.id == uuid_amazon ) | (df.id == uuid_cert)]
display(tdf)


h.message("Query from Authority")
df  = h.query('Authority')
tdf = df[(df.id == uuid_bond) | (df.id == uuid_new_bond) | (df.id == uuid_fiat)␣
 ↪| ( df.id == uuid_amazon ) | (df.id == uuid_cert)]
display(tdf)
```

```
<IPython.core.display.HTML object>
 24-08 15:36:11 | D | Running query=ListInstrument on
'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 24-08 15:36:11 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 24-08 15:36:11 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 24-08 15:36:11 | D | {'clientRequestId':
'37034.1692871571.1036139', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:11 | D | After filling values = {'clientRequestId':
'37034.1692871571.1036139', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:11 | D | 202
 24-08 15:36:11 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '37034.1692871571.1036139', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-08-24T10:06:11.138Z'}
 24-08 15:36:15 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/37034.1692871571.1036139'
 24-08 15:36:15 | D | 200
 24-08 15:36:15 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '37034.1692871571.1036139', 'flowId':
'286d0f7a-03ff-4b19-9662-955d9acff0ef', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"f9a7d1d8-63d9-4a66-a400-30cd6296fc2e","name":"Amazon
Voucher","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":200
,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"currency":"
INR"}},{"id":"cf8cd64a-bb59-4efe-85c2-b7f4ec125c20","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":6,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-08-24T10:06:12.487Z'}
 24-08 15:36:15 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/37034.1692871571.1036139'
 24-08 15:36:15 | D | 200
 24-08 15:36:15 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '37034.1692871571.1036139', 'flowId':
'286d0f7a-03ff-4b19-9662-955d9acff0ef', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"f9a7d1d8-63d9-4a66-a400-30cd6296fc2e","name":"Amazon
Voucher","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":200
,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"currency":"
INR"}},{"id":"cf8cd64a-bb59-4efe-85c2-b7f4ec125c20","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":6,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-08-24T10:06:12.487Z'}

                                     id                 name  \
0  f9a7d1d8-63d9-4a66-a400-30cd6296fc2e       Amazon Voucher
1  cf8cd64a-bb59-4efe-85c2-b7f4ec125c20  Government Bond 2024

                                          owner  \
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                        issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       200         False
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True

  expiry  verifiable  \
0   None        True
1   None        True

                                                          attributes
0                                                {'currency': 'INR'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
<IPython.core.display.HTML object>
 24-08 15:36:15 | D | Running query=ListInstrument on
'Charlie/0AA37CC89C79/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
 24-08 15:36:15 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/0AA37CC89C79'
 24-08 15:36:15 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 24-08 15:36:15 | D | {'clientRequestId':
'37034.1692871575.466315', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:15 | D | After filling values = {'clientRequestId':
'37034.1692871575.466315', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:15 | D | 202
 24-08 15:36:15 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '37034.1692871575.466315', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-08-24T10:06:15.520Z'}
 24-08 15:36:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/37034.1692871575.466315'
 24-08 15:36:19 | D | 200
 24-08 15:36:19 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '37034.1692871575.466315', 'flowId':
'f20a1c6d-3d96-442c-835f-6791775eacac', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"51627e0e-50e7-45e2-9609-568eb1aebe89","name":"BE Computer
Engineering","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"university
":"University of Mumbai","year":"2023","score":"450/500","major
subject":"[\'Robotics\', \'AI\',
\'DSP\']"}},{"id":"efc30331-18f2-4960-b2f7-9f36327b90a3","name":"Digital
Currency INR 500 Note","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":true,"expiry":null,"verifiable":true,"attributes":{"number":"OM
V 336048","year":"1981"}},{"id":"464c6085-0dca-4c82-897c-
02425d55aa9e","name":"Government Bond 2024","owner":"CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":4,"transferable":true,"expiry":null,"verifiable":true,"attributes":
{"payments":"[\'10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]',
'flowError': None, 'timestamp': '2023-08-24T10:06:16.479Z'}
 24-08 15:36:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/37034.1692871575.466315'
 24-08 15:36:19 | D | 200
 24-08 15:36:19 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '37034.1692871575.466315', 'flowId':
'f20a1c6d-3d96-442c-835f-6791775eacac', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"51627e0e-50e7-45e2-9609-568eb1aebe89","name":"BE Computer
Engineering","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"university
":"University of Mumbai","year":"2023","score":"450/500","major
subject":"[\'Robotics\', \'AI\',
\'DSP\']"}},{"id":"efc30331-18f2-4960-b2f7-9f36327b90a3","name":"Digital
Currency INR 500 Note","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":true,"expiry":null,"verifiable":true,"attributes":{"number":"OM
V 336048","year":"1981"}},{"id":"464c6085-0dca-4c82-897c-
02425d55aa9e","name":"Government Bond 2024","owner":"CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":4,"transferable":true,"expiry":null,"verifiable":true,"attributes":
{"payments":"[\'10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]',
'flowError': None, 'timestamp': '2023-08-24T10:06:16.479Z'}
                                     id                           name  \
0  51627e0e-50e7-45e2-9609-568eb1aebe89        BE Computer Engineering
1  efc30331-18f2-4960-b2f7-9f36327b90a3  Digital Currency INR 500 Note
2  464c6085-0dca-4c82-897c-02425d55aa9e           Government Bond 2024

                                            owner  \
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB
2  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       NaN         False
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       NaN          True
2  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       4.0          True

  expiry  verifiable  \
0   None        True
1   None        True
2   None        True
```

```
                                                                             ␣
 ↪                                   attributes
0  {'university': 'University of Mumbai', 'year': '2023', 'score': '450/500',␣
 ↪'major subject': '['Robotics', 'AI', 'DSP']'}
1                                                                            ␣
 ↪  {'number': 'OMV 336048', 'year': '1981'}
2                                                        {'payments':␣
 ↪'['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

```
<IPython.core.display.HTML object>
```

```
 24-08 15:36:19 | D | Running query=ListInstrument on
'Authority/9FA89008DEEE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 24-08 15:36:19 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/9FA89008DEEE'
 24-08 15:36:19 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 24-08 15:36:19 | D | {'clientRequestId':
'37034.1692871579.649333', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:19 | D | After filling values = {'clientRequestId':
'37034.1692871579.649333', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 24-08 15:36:19 | D | 202
 24-08 15:36:19 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '37034.1692871579.649333', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-08-24T10:06:19.673Z'}
 24-08 15:36:23 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/37034.1692871579.649333'
 24-08 15:36:23 | D | 200
 24-08 15:36:23 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '37034.1692871579.649333', 'flowId':
'2afddad7-1a10-4252-85a2-75330aa7a562', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"51627e0e-50e7-45e2-9609-568eb1aebe89","name":"BE Computer
Engineering","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"university
":"University of Mumbai","year":"2023","score":"450/500","major
subject":"[\'Robotics\', \'AI\',
\'DSP\']"}},{"id":"cf8cd64a-bb59-4efe-85c2-b7f4ec125c20","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":6,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\',
\'10Mar2024\']"}},{"id":"f9a7d1d8-63d9-4a66-a400-30cd6296fc2e","name":"Amazon
Voucher","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":200
,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"currency":"
INR"}},{"id":"efc30331-18f2-4960-b2f7-9f36327b90a3","name":"Digital Currency INR
500 Note","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":true,"expiry":null,"verifiable":true,"attributes":{"number":"OM
V 336048","year":"1981"}},{"id":"464c6085-0dca-4c82-897c-
02425d55aa9e","name":"Government Bond 2024","owner":"CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":4,"transferable":true,"expiry":null,"verifiable":true,"attributes":
{"payments":"[\'10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]',
'flowError': None, 'timestamp': '2023-08-24T10:06:20.685Z'}
 24-08 15:36:23 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/9FA89008DEEE/37034.1692871579.649333'
 24-08 15:36:23 | D | 200
 24-08 15:36:23 | D | {'holdingIdentityShortHash': '9FA89008DEEE',
'clientRequestId': '37034.1692871579.649333', 'flowId':
'2afddad7-1a10-4252-85a2-75330aa7a562', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"51627e0e-50e7-45e2-9609-568eb1aebe89","name":"BE Computer
Engineering","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"university
":"University of Mumbai","year":"2023","score":"450/500","major
subject":"[\'Robotics\', \'AI\',
\'DSP\']"}},{"id":"cf8cd64a-bb59-4efe-85c2-b7f4ec125c20","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":6,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\',
\'10Mar2024\']"}},{"id":"f9a7d1d8-63d9-4a66-a400-30cd6296fc2e","name":"Amazon
Voucher","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":200
,"transferable":false,"expiry":null,"verifiable":true,"attributes":{"currency":"
INR"}},{"id":"efc30331-18f2-4960-b2f7-9f36327b90a3","name":"Digital Currency INR
500 Note","owner":"CN=Charlie, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":nul
l,"transferable":true,"expiry":null,"verifiable":true,"attributes":{"number":"OM
V 336048","year":"1981"}},{"id":"464c6085-0dca-4c82-897c-
02425d55aa9e","name":"Government Bond 2024","owner":"CN=Charlie, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":4,"transferable":true,"expiry":null,"verifiable":true,"attributes":
{"payments":"[\'10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]',
'flowError': None, 'timestamp': '2023-08-24T10:06:20.685Z'}
```

```
                                     id                         name  \
0  51627e0e-50e7-45e2-9609-568eb1aebe89       BE Computer Engineering
1  cf8cd64a-bb59-4efe-85c2-b7f4ec125c20          Government Bond 2024
2  f9a7d1d8-63d9-4a66-a400-30cd6296fc2e                Amazon Voucher
3  efc30331-18f2-4960-b2f7-9f36327b90a3  Digital Currency INR 500 Note
```

```
4  464c6085-0dca-4c82-897c-02425d55aa9e           Government Bond 2024

                                            owner  \
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB
1    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
2    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
3  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB
4  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       NaN         False
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       6.0          True
2  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB     200.0         False
3  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       NaN          True
4  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB       4.0          True

  expiry  verifiable  \
0   None        True
1   None        True
2   None        True
3   None        True
4   None        True

                                                                               ␣
 ↪                                    attributes
0  {'university': 'University of Mumbai', 'year': '2023', 'score': '450/500',␣
 ↪'major subject': '['Robotics', 'AI', 'DSP']'}
1                                                    {'payments':␣
 ↪'['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
2                                                                              ␣
 ↪                       {'currency': 'INR'}
3                                                                              ␣
 ↪  {'number': 'OMV 336048', 'year': '1981'}
4                                                    {'payments':␣
 ↪'['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[6]:

```python
h.verifier_url='http://20.204.221.225:8880/'
pdf_report = h.print_to_pdf('Alice', uuid_bond, show=False)
webbrowser.open("file:///{}".format(pdf_report))
```

```
 22-05 16:42:20 | I | http://20.204.221.225:8880/?data={"name":
"Government Bond 2024", "owner": "CN=Alice, OU=Test Dept, O=R3, L=London, C=GB",
"issuer": "CN=Authority, OU=Test Dept, O=R3, L=London, C=GB", "quantity": 6.0,
"transferable": true, "expiry": null, "verifiable": true, "attributes":
{"payments": "['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']"}}
 22-05 16:42:23 | I | Time taken = 0:00:07.937600
```

```
[6]: True
```

```
[5]: h.message("Redeem of Bond (expected success)")
     h.redeem(uuid_bond, 'Alice', quantity=1)

     h.message("Redeem of Fiat Currency (expected success)")
     h.redeem(uuid_fiat, 'Charlie')

     h.message("Redeem of Amazon Voucher (expected success)")
     h.redeem(uuid_amazon, 'Alice', quantity=1)
```

```
<IPython.core.display.HTML object>

 05-10 15:02:31 | I | Redeeming instrument id

'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d' quantity '1'
 05-10 15:02:31 | I | Running action 'Redeem' on

'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:02:31 | D | Request type=post to

url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 05-10 15:02:31 | D | Original Request = {'clientRequestId':

'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Redeem',

'requestBody': {'id': '__REDEEM_ASSET_ID__', 'quantity': 1}}
 05-10 15:02:31 | D | {'clientRequestId':

'36552.1696498351.415508', 'flowClassName':

'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':

{'id': 'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d', 'quantity': 1}}
 05-10 15:02:31 | D | After filling values = {'clientRequestId':

'36552.1696498351.415508', 'flowClassName':

'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':

{'id': 'd0f0dc0e-b67c-4d27-b335-a091a5ed7c7d', 'quantity': 1}}
 05-10 15:02:31 | D | 202
 05-10 15:02:31 | D | {'holdingIdentityShortHash': 'BAA7F076B971',

'clientRequestId': '36552.1696498351.415508', 'flowId': None, 'flowStatus':

'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':

'2023-10-05T09:32:31.435Z'}
 05-10 15:02:35 | D | Request type=get to url='https://localhost:8

888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:35 | D | 200
 05-10 15:02:35 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:39 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:39 | D | 200
 05-10 15:02:39 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:43 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:43 | D | 200
 05-10 15:02:43 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:47 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:47 | D | 200
 05-10 15:02:47 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:51 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:51 | D | 200
 05-10 15:02:51 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:55 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:55 | D | 200
 05-10 15:02:55 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:02:59 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:02:59 | D | 200
 05-10 15:02:59 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:32:31.602Z'}
 05-10 15:03:03 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498351.415508'
 05-10 15:03:03 | D | 200
 05-10 15:03:03 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498351.415508', 'flowId':
'2a8fc80f-9e66-4012-9bed-7c49d584283f', 'flowStatus': 'COMPLETED', 'flowResult':
'net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@106943aa', 'flowError':
None, 'timestamp': '2023-10-05T09:33:01.643Z'}
 05-10 15:03:03 | I |
net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@106943aa
 05-10 15:03:03 | I | Time taken = 0:00:32.304391
<IPython.core.display.HTML object>
 05-10 15:03:03 | I | Redeeming instrument id
'28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7' quantity 'None'
 05-10 15:03:03 | I | Running action 'Redeem' on
'Charlie/0AA37CC89C79/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:03:03 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/0AA37CC89C79'
 05-10 15:03:03 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Redeem',
'requestBody': {'id': '__REDEEM_ASSET_ID__'}}
 05-10 15:03:03 | D | {'clientRequestId':
'36552.1696498383.721802', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7'}}
```

```
 05-10 15:03:03 | D | After filling values = {'clientRequestId':
'36552.1696498383.721802', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '28f10fb0-9e2b-4ab0-9fdb-1573565cb3c7'}}
 05-10 15:03:03 | D | 202
 05-10 15:03:03 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:33:03.744Z'}
 05-10 15:03:07 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:07 | D | 200
 05-10 15:03:07 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:11 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:11 | D | 200
 05-10 15:03:11 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:15 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:15 | D | 200
 05-10 15:03:15 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:19 | D | 200
 05-10 15:03:19 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:23 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:23 | D | 200
 05-10 15:03:23 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:27 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:27 | D | 200
 05-10 15:03:27 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:31 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:32 | D | 200
 05-10 15:03:32 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:36 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:36 | D | 200
 05-10 15:03:36 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:03.996Z'}
 05-10 15:03:40 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/0AA37CC89C79/36552.1696498383.721802'
 05-10 15:03:40 | D | 200
 05-10 15:03:40 | D | {'holdingIdentityShortHash': '0AA37CC89C79',
'clientRequestId': '36552.1696498383.721802', 'flowId':
'0e2e9da2-ed4b-482e-8e44-5aafb8429cc3', 'flowStatus': 'COMPLETED', 'flowResult':
'net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@22270714', 'flowError':
None, 'timestamp': '2023-10-05T09:33:38.454Z'}
 05-10 15:03:40 | I |
net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@22270714
 05-10 15:03:40 | I | Time taken = 0:00:36.365537
<IPython.core.display.HTML object>
 05-10 15:03:40 | I | Redeeming instrument id
'0fb152f0-805c-4d63-a889-3485c92df378' quantity '1'
 05-10 15:03:40 | I | Running action 'Redeem' on
'Alice/BAA7F076B971/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 05-10 15:03:40 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/BAA7F076B971'
 05-10 15:03:40 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Redeem',
'requestBody': {'id': '__REDEEM_ASSET_ID__', 'quantity': 1}}
 05-10 15:03:40 | D | {'clientRequestId':
'36552.1696498420.089261', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '0fb152f0-805c-4d63-a889-3485c92df378', 'quantity': 1}}
 05-10 15:03:40 | D | After filling values = {'clientRequestId':
'36552.1696498420.089261', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '0fb152f0-805c-4d63-a889-3485c92df378', 'quantity': 1}}
 05-10 15:03:40 | D | 202
 05-10 15:03:40 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-10-05T09:33:40.108Z'}
 05-10 15:03:44 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:03:44 | D | 200
 05-10 15:03:44 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:03:48 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:03:48 | D | 200
 05-10 15:03:48 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:03:52 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:03:52 | D | 200
 05-10 15:03:52 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:03:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:03:56 | D | 200
 05-10 15:03:56 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:04:00 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:04:00 | D | 200
 05-10 15:04:00 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:04:04 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:04:04 | D | 200
 05-10 15:04:04 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:04:08 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:04:08 | D | 200
 05-10 15:04:08 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-10-05T09:33:40.432Z'}
 05-10 15:04:12 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/BAA7F076B971/36552.1696498420.089261'
 05-10 15:04:12 | D | 200
 05-10 15:04:12 | D | {'holdingIdentityShortHash': 'BAA7F076B971',
'clientRequestId': '36552.1696498420.089261', 'flowId':
'7ee180c7-d010-47ee-96c6-d69cf2c358da', 'flowStatus': 'COMPLETED', 'flowResult':
'net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@7ebb6a72', 'flowError':
None, 'timestamp': '2023-10-05T09:34:09.520Z'}
 05-10 15:04:12 | I |
net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@7ebb6a72
 05-10 15:04:12 | I | Time taken = 0:00:32.327748
```

```
[5]: ('36552.1696498420.089261',
      <Response [200]>,
      'net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@7ebb6a72')
```

```
[ ]:
```

```
[2]: df = h.query('Alice')
```

```
[2]: h.query_all_nodes()
```

```
 21-05 13:53:02 | I | Result for query=ListInstrument from
'Authority/8175C6EA55DE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.312654)

                                     id                 name
0  3d4c026b-7721-4f1d-97e4-5b823c61c568  Government Bond 2024  \
1  c93b9892-7416-4ae2-9ced-681e326e241d  Government Bond 2024
2  a39b03d2-f27b-4e22-a1a6-6ffba27c40c9  Government Bond 2024
```

```
3  b1a7a507-2c93-4153-a3de-22fbebac153e  Government Bond 2024
4  0184710b-57d9-4e7e-afa6-7ad7373f9e58  Government Bond 2024

                                            owner
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB  \
1    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
2    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
3      CN=Bob, OU=Test Dept, O=R3, L=London, C=GB
4      CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB        10          True
2  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True
3  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True
4  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True

  expiry  verifiable
0   None        True  \
1   None        True
2   None        True
3   None        True
4   None        True

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
2  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
3  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
4  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
 21-05 13:53:06 | I | Result for query=ListInstrument from
'Bob/B31CAD5945F1/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.246007)
                                     id                  name
0  0184710b-57d9-4e7e-afa6-7ad7373f9e58  Government Bond 2024  \
1  b1a7a507-2c93-4153-a3de-22fbebac153e  Government Bond 2024

                                        owner
0  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB  \
1  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True
```

```
  expiry  verifiable
0   None        True  \
1   None        True

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
 21-05 13:53:10 | I | Result for query=ListInstrument from
'Charlie/AB7643993711/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.248719)
                                     id                  name
0  3d4c026b-7721-4f1d-97e4-5b823c61c568  Government Bond 2024  \

                                            owner
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB  \

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True  \

  expiry  verifiable
0   None        True  \

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
 21-05 13:53:14 | I | Result for query=ListInstrument from
'Alice/415E407297AA/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.315236)
                                     id                  name
0  c93b9892-7416-4ae2-9ced-681e326e241d  Government Bond 2024  \
1  a39b03d2-f27b-4e22-a1a6-6ffba27c40c9  Government Bond 2024

                                          owner
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB  \
1  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB        10          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True

  expiry  verifiable
0   None        True  \
1   None        True
```

```
                                                         attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

```
 21-05 13:53:19 | I | Result for query=ListInstrument from
'Dave/FB9F8A560956/CN=Dave, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.284337)
```

```
Empty DataFrame
Columns: []
Index: []
```

[5]:

```
# Issue -- Type 1, Transferable = Y, Quantifiable = Y
# Example 1 -- Government Bond with defined coupon payments -- Type 1 / Test 1␣
 ↪-- Transferable, Quantifiable Instrument
h.message("Type 1, Transferable = Y, Quantifiable = Y -- Example 1 --␣
 ↪Government Bond with defined coupon payments", h=3)
h.message("Issue -- expected=success", h=4)

(req_id, response, return_val_uuid) = h.issue('Authority', 'Alice', 'Government␣
 ↪Bond 2024',
        quantity=10,
        transferable=True,
        expiry=None,
        verifiable=True,
        attributes={'payments': "['10Jun2023', '10Sep2023', '10Dec2023',␣
 ↪'10Mar2024']"})

h.message([req_id, response, return_val_uuid])
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
 21-05 21:17:16 | I | Issuing instrument='Government Bond 2024' to
'Alice'
 21-05 21:17:16 | I | Running action 'Issue' on
'Authority/8175C6EA55DE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 21-05 21:17:24 | I | a2e4f105-4f79-4425-98d6-bfeed72d68c5
 21-05 21:17:24 | I | Time taken = 0:00:08.376892
```

```
5837.1684684036.282727<Response [200]>a2e4f105-4f79-4425-98d6-bfeed72d68c5
```

[3]:

```
help(h.transfer)
```

```
Help on method transfer in module corda5Interface:

transfer(inst_id, from_node, to_node, quantity=None) method of
```

```
corda5Interface.Corda5 instance
```

```
[6]: # Transfer from issuer -- expected=failure
     h.message("Transfer from issuer -- for instrument Type 1, Transferable = Y,␣
      ↪Quantifiable = Y -- expected=failure", h=4)
     (req_id, response, return_val) = h.transfer(return_val_uuid, 'Authority',␣
      ↪'Charlie', quantity=4)
     h.message([req_id, response, return_val])
```

```
<IPython.core.display.HTML object>

 21-05 21:17:36 | I | Tranfering instrument id

'a2e4f105-4f79-4425-98d6-bfeed72d68c5' to 'Charlie'
 21-05 21:17:36 | I | Running action 'Transfer' on

'Authority/8175C6EA55DE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 21-05 21:17:40 | I | {'type': 'FLOW_FAILED', 'message': 'Not able

to find a owning state with id a2e4f105-4f79-4425-98d6-bfeed72d68c5'}
 21-05 21:17:40 | I | Time taken = 0:00:04.191121

5837.1684684056.2325<Response [200]>{'type': 'FLOW_FAILED', 'message': 'Not able
to find a owning state with id a2e4f105-4f79-4425-98d6-bfeed72d68c5'}
```

```
[7]: # Partial transfer from owner
     h.message("Partial transfer from owner -- for instrument Type 1, Transferable =␣
      ↪Y, Quantifiable = Y -- expected=success", 4)
     (req_id, response, return_val) = h.transfer(return_val_uuid, 'Alice', 'Bob',␣
      ↪quantity=6)
     h.message([req_id, response, return_val], 4)
```

```
<IPython.core.display.HTML object>

 21-05 21:17:49 | I | Tranfering instrument id

'a2e4f105-4f79-4425-98d6-bfeed72d68c5' to 'Bob'
 21-05 21:17:49 | I | Running action 'Transfer' on

'Alice/415E407297AA/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 21-05 21:18:06 | I |
net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@23e7b70
 21-05 21:18:06 | I | Time taken = 0:00:16.496626

5837.1684684069.8207748<Response
[200]>net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@23e7b70
```

```
[ ]: # print(return_val_uuid)
     h.query_all_nodes()
```

```
 21-05 13:52:33 | I | Result for query=ListInstrument from
'Authority/8175C6EA55DE/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.303915)
                                     id                  name  \
0  3d4c026b-7721-4f1d-97e4-5b823c61c568  Government Bond 2024   
1  c93b9892-7416-4ae2-9ced-681e326e241d  Government Bond 2024   
2  a39b03d2-f27b-4e22-a1a6-6ffba27c40c9  Government Bond 2024   
3  b1a7a507-2c93-4153-a3de-22fbebac153e  Government Bond 2024   
4  0184710b-57d9-4e7e-afa6-7ad7373f9e58  Government Bond 2024   

                                            owner  \
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB   
1    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB   
2    CN=Alice, OU=Test Dept, O=R3, L=London, C=GB   
3      CN=Bob, OU=Test Dept, O=R3, L=London, C=GB   
4      CN=Bob, OU=Test Dept, O=R3, L=London, C=GB   

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True   
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB        10          True   
2  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True   
3  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True   
4  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True   

  expiry  verifiable  \
0   None        True   
1   None        True   
2   None        True   
3   None        True   
4   None        True   

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
2  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
3  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
4  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
 21-05 13:52:37 | I | Result for query=ListInstrument from
'Bob/B31CAD5945F1/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'. (Time
taken=0:00:04.251358)
                                     id                  name  \
0  0184710b-57d9-4e7e-afa6-7ad7373f9e58  Government Bond 2024   
1  b1a7a507-2c93-4153-a3de-22fbebac153e  Government Bond 2024   
```

```
                                        owner
0  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB  \
1  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True

  expiry  verifiable
0   None        True  \
1   None        True

                                                              attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[13]:

```
h.query('Alice')
```

[13]:

```
                                     id                  name
0  c93b9892-7416-4ae2-9ced-681e326e241d  Government Bond 2024  \

                                          owner
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB  \

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB        10          True  \

  expiry  verifiable
0   None        True  \

                                                              attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[13]:

```
# pdf_report = h.print_to_pdf('Alice', 'return_val_uuid', show=False)
# webbrowser.open("file:///{}".format(pdf_report))
```

[13]: True

[12]:

```
# Full Transfer from owner/holder -- expected=success
h.message("Full Transfer from owner/holder -- for instrument Type 1,␣
 ↪Transferable = Y, Quantifiable = Y -- expected=success", 4)

# (req_id, response, return_val) = h.transfer(return_val_uuid, 'Alice',␣
 ↪'Charlie', quantity=4)
(req_id, response, return_val) = h.transfer(return_val_uuid, 'Alice', 'Charlie')
h.message([req_id, response, return_val])
```

```
<IPython.core.display.HTML object>
```

```
 17-05 16:16:48 | I | Tranfering instrument id
'3d4c026b-7721-4f1d-97e4-5b823c61c568' to 'Charlie'
 17-05 16:16:48 | I | Running action 'Transfer' on
'Alice/415E407297AA/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 17-05 16:17:05 | I |
net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@7feee57e
 17-05 16:17:05 | I | Time taken = 0:00:16.414052

55970.1684320408.736785<Response
[200]>net.corda.ledger.utxo.flow.impl.FinalizationResultImpl@7feee57e
```

[14]:

```
# Transfer attempt from non-holder -- expected=failure
h.message("Transfer from non-holder -- for instrument Type 1, Transferable = Y,␣
 ↪Quantifiable = Y -- expected=failure", 4)
(req_id, response, return_val) = h.transfer(return_val_uuid, 'Alice',␣
 ↪'Charlie', quantity=4)
h.message([req_id, response, return_val])
```

```
<IPython.core.display.HTML object>

 17-05 16:18:55 | I | Tranfering instrument id
'3d4c026b-7721-4f1d-97e4-5b823c61c568' to 'Charlie'
 17-05 16:18:55 | I | Running action 'Transfer' on
'Alice/415E407297AA/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 17-05 16:18:59 | I | {'type': 'FLOW_FAILED', 'message': 'Not able
to find a owning state with id 3d4c026b-7721-4f1d-97e4-5b823c61c568'}
 17-05 16:18:59 | I | Time taken = 0:00:04.180356

55970.1684320535.016062<Response [200]>{'type': 'FLOW_FAILED', 'message': 'Not
able to find a owning state with id 3d4c026b-7721-4f1d-97e4-5b823c61c568'}
```

[10]:

```
h.query('Charlie')
```

[10]:

```
                                     id                  name  
0  18bf1dee-6f7f-4585-8058-6dea40af6c28  Government Bond 2024   \
1  09fbf9fc-d1d4-42bd-9019-06b892d97757  Government Bond 2024   


                                         owner  
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB   \
1  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB   


                                          issuer  quantity  transferable  
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True   \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True   


  expiry  verifiable
```

```
0   None        True  \
1   None        True

                                                              attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[11]:

```
h.query('Bob')
```

[11]:

```
                                     id                  name
0  e8794d36-64b8-41f4-a758-41a907fa8e8d  Government Bond 2024  \
1  8a1a72ab-959c-4dcc-ab81-94b822eaafe7  Government Bond 2024

                                        owner
0  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB  \
1  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         6          True

  expiry  verifiable
0   None        True  \
1   None        True

                                                              attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[21]:

```
# Redeem
help(h.redeem)
```

```
Help on method redeem in module json_on_jupyter:

redeem(inst_id, from_node, quantity=None) method of
json_on_jupyter.JSON_ON_Jupyter instance
```

[2]:

```
h.query('Alice')
```

[2]:

```
Empty DataFrame
Columns: []
Index: []
```

[24]:

```
# Redeem - Partial by holder -- exepected=success
h.redeem('8f90d61f-5380-417a-a647-1194f4033b01', 'Bob', quantity=1)
```

```
 11-05 18:02:48 | I | Redeeming instrument id
'8f90d61f-5380-417a-a647-1194f4033b01' quantity '1'
 11-05 18:02:48 | I | Running action 'Redeem' on
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 11-05 18:03:41 | I | UtxoSignedTransactionImpl(id=SHA-
256D:39AEE0AB718BFA57EB45ED5EAFC6B7310BE8E52E4B1B8CA22D18FE7D31E5E7E8, signature
s=[DigitalSignatureAndMetadata[signature=[3044022068E89A43D1E511F269CF9DAF9F4A20
304CADB533842605D5D6A87BBA8F50A2F40220374D78E5B8BACFF4F423A6498F9B9C328630E35188
D9A8E85A4AA7D2746171B5],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:32:54.621327Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3044022066F92D6E13224E1D9F92
3C1C6570CA26E8A706843315777BB9DA96E356FC5B8802203D0AF3C7EF13D47F55F90599A1B28BF8
55124FB286921BC4EF0C6B1ECB1BBD7E],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:33:10.806098Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[304402206CA18ADE6192D43C94B9
600F8D1A61DD69220D40B1989737FC123E25E7521B6502205143FC562C20288B18DED43B2700ED43
BCCDB05ECC81ECAE0533EE277043FF67],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:33:23.561467Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@eb457b24]], wireTransaction=W
ireTransaction(id=SHA-
256D:39AEE0AB718BFA57EB45ED5EAFC6B7310BE8E52E4B1B8CA22D18FE7D31E5E7E8,
privacySalt=[365CDC2A1B4784AEAB85F5108F2F564E3235C350209755D2E75E377BC10D7969],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62549),
(size=868, sum=55282)], [(size=295, sum=10343), (size=295, sum=10987)],
[(size=2082, sum=122958)], [(size=556, sum=34280)], [], [(size=774, sum=39636)],
[], [(size=3356, sum=195316)], [(size=790, sum=52841)]]))
```

```
[24]: ('13652.1683808368.2703528',
       <Response [200]>,
       'UtxoSignedTransactionImpl(id=SHA-
      256D:39AEE0AB718BFA57EB45ED5EAFC6B7310BE8E52E4B1B8CA22D18FE7D31E5E7E8, signature
      s=[DigitalSignatureAndMetadata[signature=[3044022068E89A43D1E511F269CF9DAF9F4A20
      304CADB533842605D5D6A87BBA8F50A2F40220374D78E5B8BACFF4F423A6498F9B9C328630E35188
      D9A8E85A4AA7D2746171B5],
      metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:32:54.621327Z,
      signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
      name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
      proof=null], DigitalSignatureAndMetadata[signature=[3044022066F92D6E13224E1D9F92
      3C1C6570CA26E8A706843315777BB9DA96E356FC5B8802203D0AF3C7EF13D47F55F90599A1B28BF8
      55124FB286921BC4EF0C6B1ECB1BBD7E],
      metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:33:10.806098Z,
      signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
      name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
      proof=null], DigitalSignatureAndMetadata[signature=[304402206CA18ADE6192D43C94B9
      600F8D1A61DD69220D40B1989737FC123E25E7521B6502205143FC562C20288B18DED43B2700ED43
      BCCDB05ECC81ECAE0533EE277043FF67],
      metadata=DigitalSignatureMetadata[timestamp=2023-05-11T12:33:23.561467Z,
      signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
      name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
      batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
      batchMerkleTreeDigestAlgorithmName=SHA-256D,
      batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
      batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
      proof=net.corda.crypto.merkle.impl.MerkleProofImpl@eb457b24]], wireTransaction=W
      ireTransaction(id=SHA-
      256D:39AEE0AB718BFA57EB45ED5EAFC6B7310BE8E52E4B1B8CA22D18FE7D31E5E7E8,
      privacySalt=[365CDC2A1B4784AEAB85F5108F2F564E3235C350209755D2E75E377BC10D7969],
      metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
      69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62549),
      (size=868, sum=55282)], [(size=295, sum=10343), (size=295, sum=10987)],
      [(size=2082, sum=122958)], [(size=556, sum=34280)], [], [(size=774, sum=39636)],
      [], [(size=3356, sum=195316)], [(size=790, sum=52841)]]))')
```

```
[36]: # Redeem - Partial by Issuer -- exepected=success
      h.redeem('be8f210d-b376-418c-9c0f-b051ce125a33', 'Authority', quantity=1)
```

```
 11-05 18:54:15 | I | Redeeming instrument id
'be8f210d-b376-418c-9c0f-b051ce125a33' quantity '1'
 11-05 18:54:15 | I | Running action 'Redeem' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 11-05 18:55:16 | I | UtxoSignedTransactionImpl(id=SHA-
256D:FEE9545D50015115BEA3B7C13001FCC9CEB7BA45A73F9F2CB9C2823D709BF403, signature
s=[DigitalSignatureAndMetadata[signature=[30460221009182B3F47C0D9F58547590BE4251
84CEB7D5978A6DC5C981B036EBE32D4D99B0022100D6B40D5C3AD082F3D3562DA9E77898078F82DD
D4837E52818E7E5B146BB86616],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:25.326126Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[30450220689F998A3E8E3E2EEC51
98FB3AEB94E6C0E49E7F2E2E63B1811B105C80D7F221022100B3A0CE17172717BB290CE8543ED24D
21012FC2F741AEC3A809D3A016B9835FBE],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:46.185261Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3044022033CB21729A7AF757B183
315F81615A19851A751BBA7EF2C51C8E3A7E96F56C5902202B8DA2C3F8CC4225E68FEC67111D977A
BDC5B3253EC3A750F6A293326FB08EB3],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:58.148029Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@151961d5]], wireTransaction=W
ireTransaction(id=SHA-
256D:FEE9545D50015115BEA3B7C13001FCC9CEB7BA45A73F9F2CB9C2823D709BF403,
privacySalt=[E64237E76EE682C71CF27CC8692D560B0089E070D93AA3A1CC96972EAE639D60],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62549),
(size=868, sum=55358)], [(size=295, sum=10987), (size=295, sum=10343)],
[(size=2082, sum=122958)], [(size=556, sum=34280)], [], [(size=774, sum=40436)],
[], [(size=3356, sum=195349)], [(size=790, sum=52841)]]))
```

[36]:
```
('13652.1683811455.264299',
 <Response [200]>,
 'UtxoSignedTransactionImpl(id=SHA-
256D:FEE9545D50015115BEA3B7C13001FCC9CEB7BA45A73F9F2CB9C2823D709BF403, signature
s=[DigitalSignatureAndMetadata[signature=[3046022100918 2B3F47C0D9F58547590BE4251
84CEB7D5978A6DC5C981B036EBE32D4D99B0022100D6B40D5C3AD082F3D3562DA9E77898078F82DD
D4837E52818E7E5B146BB86616],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:25.326126Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[30450220689F998A3E8E3E2EEC51
98FB3AEB94E6C0E49E7F2E2E63B1811B105C80D7F221022100B3A0CE17172717BB290CE8543ED24D
21012FC2F741AEC3A809D3A016B9835FBE],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:46.185261Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3044022033CB21729A7AF757B183
315F81615A19851A751BBA7EF2C51C8E3A7E96F56C5902202B8DA2C3F8CC4225E68FEC67111D977A
BDC5B3253EC3A750F6A293326FB08EB3],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:24:58.148029Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@151961d5]], wireTransaction=W
ireTransaction(id=SHA-
256D:FEE9545D50015115BEA3B7C13001FCC9CEB7BA45A73F9F2CB9C2823D709BF403,
privacySalt=[E64237E76EE682C71CF27CC8692D560B0089E070D93AA3A1CC96972EAE639D60],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62549),
(size=868, sum=55358)], [(size=295, sum=10987), (size=295, sum=10343)],
[(size=2082, sum=122958)], [(size=556, sum=34280)], [], [(size=774, sum=40436)],
[], [(size=3356, sum=195349)], [(size=790, sum=52841)]]))')
```

[37]:
```python
h.query('Authority')
```

[37]:
```
                                     id                  name  \
0  8f90d61f-5380-417a-a647-1194f4033b01  Government Bond 2024
1  be8f210d-b376-418c-9c0f-b051ce125a33  Government Bond 2024

                                        owner  \
0  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer   quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         1           True
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         3           True

  expiry  verifiable  \
0   None        True
1   None        True

                                                           attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[40]:

```
# Redeem - Full by holder -- exepected=success
h.redeem('be8f210d-b376-418c-9c0f-b051ce125a33', 'Bob', quantity=3)
```

```
 11-05 19:22:05 | I | Redeeming instrument id
'be8f210d-b376-418c-9c0f-b051ce125a33' quantity '3'
 11-05 19:22:05 | I | Running action 'Redeem' on
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 11-05 19:23:02 | I | UtxoSignedTransactionImpl(id=SHA-
256D:D4DC0A91F99E0973C770A4E9B1B422AEE6BA4B1C9F1AB2BEBE29C2B909D99FF9, signature
s=[DigitalSignatureAndMetadata[signature=[3046022100A76E7F8134BF4707BE2460E00C9B
05E023EBDADF01F5D8304060F22158E9FE57022100FE0527B885912D31D6888DBF90C5BC8A0A228D
034E1B19080483C2DB6A799B2D],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:12.853107Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[30450220632BF51560B3AA562EFF
980B6423D07F517C67B8D065D608D5150A20552FAA89022100E3BAF15335A8B8C2159695048865B5
88A62D111D051C696EA8BD9078DCD51336],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:31.371126Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[304502201F92DEE5F3A3204D4C6D
C0D0CD85E5FF0CF630EA32534EDABF7B0513F42CEE75022100D5A34EE0E2B6C22F1689147F40C9B7
516CF352BA0ED99ED2E98AA34D69555AEA],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:42.734914Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@a09aeaef]], wireTransaction=W
ireTransaction(id=SHA-
256D:D4DC0A91F99E0973C770A4E9B1B422AEE6BA4B1C9F1AB2BEBE29C2B909D99FF9,
privacySalt=[FD5C8D2121157CD4DB751DC006E7023DF867A525FE7CC281A71858CD3D8AB8E6],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@3159a9
56, componentGroupLists=[[(size=1278, sum=109143)], [(size=1210, sum=62549),
(size=868, sum=55111)], [(size=295, sum=10343), (size=295, sum=10987)], [],
[(size=556, sum=34280)], [], [(size=774, sum=39822)], [], [], [(size=790,
sum=52841)]]))
```

[40]:
```
('13652.1683813125.7133381',
 <Response [200]>,
 'UtxoSignedTransactionImpl(id=SHA-
256D:D4DC0A91F99E0973C770A4E9B1B422AEE6BA4B1C9F1AB2BEBE29C2B909D99FF9, signature
s=[DigitalSignatureAndMetadata[signature=[3046022100A76E7F8134BF4707BE2460E00C9B
05E023EBDADF01F5D8304060F22158E9FE57022100FE0527B885912D31D6888DBF90C5BC8A0A228D
034E1B19080483C2DB6A799B2D],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:12.853107Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[30450220632BF51560B3AA562EFF
980B6423D07F517C67B8D065D608D5150A20552FAA89022100E3BAF15335A8B8C2159695048865B5
88A62D111D051C696EA8BD9078DCD51336],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:31.371126Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[304502201F92DEE5F3A3204D4C6D
C0D0CD85E5FF0CF630EA32534EDABF7B0513F42CEE75022100D5A34EE0E2B6C22F1689147F40C9B7
516CF352BA0ED99ED2E98AA34D69555AEA],
metadata=DigitalSignatureMetadata[timestamp=2023-05-11T13:52:42.734914Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@a09aeaef]], wireTransaction=W
ireTransaction(id=SHA-
256D:D4DC0A91F99E0973C770A4E9B1B422AEE6BA4B1C9F1AB2BEBE29C2B909D99FF9,
privacySalt=[FD5C8D2121157CD4DB751DC006E7023DF867A525FE7CC281A71858CD3D8AB8E6],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@3159a9
56, componentGroupLists=[[(size=1278, sum=109143)], [(size=1210, sum=62549),
(size=868, sum=55111)], [(size=295, sum=10343), (size=295, sum=10987)], [],
[(size=556, sum=34280)], [], [(size=774, sum=39822)], [], [], [(size=790,
sum=52841)]]))')
```

[38]:
```python
h.query('Bob')
```

[38]:
```
                                     id                  name  \
0  8f90d61f-5380-417a-a647-1194f4033b01  Government Bond 2024
1  be8f210d-b376-418c-9c0f-b051ce125a33  Government Bond 2024

                                        owner  \
0  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Bob, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         5          True
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         3          True

  expiry  verifiable  \
0   None        True
1   None        True

                                                              attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[30]:

```
h.query_all_nodes() # calling flow = ListInstrument

# for n in nodes:
#     df = h.query(n, query, False)
#     h.log.info("Result for '{}' from '{}/{}/{}'".format(query, n, nodes[n]['x500'], nodes[n]['hash']))
#     display(df)
```

```
 10-05 11:18:38 | D | Running query=ListInstrument on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 10-05 11:18:38 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/069C8B8CD60D'
 10-05 11:18:38 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 10-05 11:18:38 | D | {'clientRequestId':
'76808.1683697718.418418', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:38 | D | After filling values = {'clientRequestId':
'76808.1683697718.418418', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:38 | D | 202
 10-05 11:18:38 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683697718.418418', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-10T05:48:38.474Z'}
```

```
 10-05 11:18:42 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683697718.418418'
 10-05 11:18:42 | D | 200
 10-05 11:18:42 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683697718.418418', 'flowId':
'c167a9a0-a1be-4c47-ab3c-e3cf3b23b1be', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":0,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}},{"id":"104ae922-
d085-4b29-8e24-6014057f1a50","name":"eRupee","owner":"CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":100001,"transferable":true,"expiry":null,"verifiable":true,"attribu
tes":{}}]', 'flowError': None, 'timestamp': '2023-05-10T05:48:40.816Z'}
 10-05 11:18:42 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683697718.418418'
 10-05 11:18:42 | D | 200
 10-05 11:18:42 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683697718.418418', 'flowId':
'c167a9a0-a1be-4c47-ab3c-e3cf3b23b1be', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":0,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}},{"id":"104ae922-
d085-4b29-8e24-6014057f1a50","name":"eRupee","owner":"CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":100001,"transferable":true,"expiry":null,"verifiable":true,"attribu
tes":{}}]', 'flowError': None, 'timestamp': '2023-05-10T05:48:40.816Z'}
 10-05 11:18:42 | I | Result for query=ListInstrument from
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'

                                     id                  name  \
0  9339350b-d16e-4a94-91ac-e363902172a8  Government Bond 2024
```

```
1  104ae922-d085-4b29-8e24-6014057f1a50                eRupee

                                          owner  \
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         0          True
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB    100001          True

  expiry  verifiable  \
0   None        True
1   None        True

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1                                                                    {}
```

```
 10-05 11:18:42 | D | Running query=ListInstrument on
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'
 10-05 11:18:42 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/83214F5048CD'
 10-05 11:18:42 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 10-05 11:18:42 | D | {'clientRequestId':
'76808.1683697722.625798', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:42 | D | After filling values = {'clientRequestId':
'76808.1683697722.625798', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:42 | D | 202
 10-05 11:18:42 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683697722.625798', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-10T05:48:42.647Z'}
 10-05 11:18:46 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/83214F5048CD/76808.1683697722.625798'
 10-05 11:18:46 | D | 200
 10-05 11:18:46 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683697722.625798', 'flowId':
'72c2e28e-0205-4974-9ba7-415b15d54f49', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-10T05:48:45.242Z'}
 10-05 11:18:46 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/83214F5048CD/76808.1683697722.625798'
 10-05 11:18:46 | D | 200
 10-05 11:18:46 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683697722.625798', 'flowId':
'72c2e28e-0205-4974-9ba7-415b15d54f49', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-10T05:48:45.242Z'}
 10-05 11:18:46 | I | Result for query=ListInstrument from
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'

Empty DataFrame
Columns: []
Index: []

 10-05 11:18:46 | D | Running query=ListInstrument on
'Charlie/D728EEE9116F/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
 10-05 11:18:46 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/D728EEE9116F'
 10-05 11:18:46 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 10-05 11:18:46 | D | {'clientRequestId':
'76808.1683697726.758862', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:46 | D | After filling values = {'clientRequestId':
'76808.1683697726.758862', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:46 | D | 202
 10-05 11:18:46 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683697726.758862', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-10T05:48:46.775Z'}
 10-05 11:18:50 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/D728EEE9116F/76808.1683697726.758862'
 10-05 11:18:50 | D | 200
 10-05 11:18:50 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683697726.758862', 'flowId':
'298112ce-9693-40e7-876f-408b034132d6', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-10T05:48:49.176Z'}
 10-05 11:18:50 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/D728EEE9116F/76808.1683697726.758862'
 10-05 11:18:50 | D | 200
 10-05 11:18:50 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683697726.758862', 'flowId':
'298112ce-9693-40e7-876f-408b034132d6', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-10T05:48:49.176Z'}
 10-05 11:18:50 | I | Result for query=ListInstrument from
'Charlie/D728EEE9116F/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'

Empty DataFrame
Columns: []
Index: []

 10-05 11:18:50 | D | Running query=ListInstrument on
'Alice/98A345A3E88B/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 10-05 11:18:50 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/98A345A3E88B'
 10-05 11:18:50 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 10-05 11:18:50 | D | {'clientRequestId': '76808.1683697730.9354',
'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
```

```
 10-05 11:18:50 | D | After filling values = {'clientRequestId':
'76808.1683697730.9354', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 10-05 11:18:50 | D | 202
 10-05 11:18:50 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683697730.9354', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-10T05:48:50.958Z'}
 10-05 11:18:54 | D | Request type=get to
url='https://localhost:8888/api/v1/flow/98A345A3E88B/76808.1683697730.9354'
 10-05 11:18:55 | D | 200
 10-05 11:18:55 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683697730.9354', 'flowId':
'28efb3c8-1f72-4d68-9f43-f3417bd15624', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":5,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}},{"id":"104ae922-
d085-4b29-8e24-6014057f1a50","name":"eRupee","owner":"CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":100001,"transferable":true,"expiry":null,"verifiable":true,"attribu
tes":{}}]', 'flowError': None, 'timestamp': '2023-05-10T05:48:53.615Z'}
 10-05 11:18:55 | D | Request type=get to
url='https://localhost:8888/api/v1/flow/98A345A3E88B/76808.1683697730.9354'
 10-05 11:18:55 | D | 200
 10-05 11:18:55 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683697730.9354', 'flowId':
'28efb3c8-1f72-4d68-9f43-f3417bd15624', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":5,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}},{"id":"104ae922-
d085-4b29-8e24-6014057f1a50","name":"eRupee","owner":"CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB
","quantity":100001,"transferable":true,"expiry":null,"verifiable":true,"attribu
tes":{}}]', 'flowError': None, 'timestamp': '2023-05-10T05:48:53.615Z'}
 10-05 11:18:55 | I | Result for query=ListInstrument from
'Alice/98A345A3E88B/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
                                     id                name  \
0  9339350b-d16e-4a94-91ac-e363902172a8  Government Bond 2024
1  104ae922-d085-4b29-8e24-6014057f1a50                eRupee

                                          owner  \
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB
1  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         5          True
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB    100001          True

  expiry  verifiable  \
0   None        True
1   None        True

                                                               attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1                                                                      {}
```

[3]:

```
 09-05 13:20:19 | I | Issuing instrument='Government Bond 2024' to
'Alice'
 09-05 13:20:19 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
```

```
09-05 13:21:24 | I | 9339350b-d16e-4a94-91ac-e363902172a8
```

[3]:
```
('67712.1683618619.295648',
 <Response [200]>,
 '9339350b-d16e-4a94-91ac-e363902172a8')
```

[41]:
```
# Example 2 -- Digital currency (eRupee) -- Type 1 / Test 1 -- Transferable,␣
 ↪Quantifiable Instrument

(req_id, response, return_val_uuid) = h.issue('Authority', 'Alice', 'eRupee',
        quantity=100001,
        transferable=True,
        expiry=None,
        verifiable=True
       )
print(req_id, response, return_val_uuid)
```

```
 11-05 19:41:42 | I | Issuing instrument='eRupee' to 'Alice'
 11-05 19:41:42 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 11-05 19:42:38 | I | 3d2ecb60-cdb8-4004-9c79-3cde976969fc

13652.1683814302.192371 <Response [200]> 3d2ecb60-cdb8-4004-9c79-3cde976969fc
```

[44]:
```
# Example 3 -- Unique artifacts on Ledger -- Type 2 / Test 2 -- Transferable,␣
 ↪non Quantifiable Instrument
(req_id, response, return_val_uuid) = h.issue('Authority', 'Alice', 'Unique␣
 ↪artifacts on Ledger. The Mona Lisa',
        quantity=None,
        transferable=True,
        expiry=None,
        verifiable=True,
        attributes={'holding autority': 'The Louvre Museum, Paris'}
       )
print(req_id, response, return_val_uuid)
```

```
 11-05 20:01:26 | I | Issuing instrument='Unique artifacts on
Ledger. The Mona Lisa' to 'Alice'
 11-05 20:01:26 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 11-05 20:02:27 | I | aae1090f-beea-4341-a342-1056d02f98ab
```

[ ]:
```
# another example  -- Limited Edition Soverign -- Type 1 / Test 1 --␣
 ↪Transferable, Quantifiable Instrument
```

[45]:
```
# Example 4 -- Fiat currency with unique number in digital form -- Type 2 /␣
 ↪Test 2 -- Transferable, non Quantifiable Instrument
```

```
# - Issue Instrument transferable=True, quantity=null(non aggregable, even if␣
 ↪quantity is 1, still since id is different, it won't aggregage)
#     example "Digital Currency INR 500 Note", attributes{'id': "OMV 336048",␣
 ↪'year': '', '', ''}
(req_id, response, return_val_uuid) = h.issue('Authority', 'Alice', 'Digital␣
 ↪Currency INR 500 Note',
        quantity=None, # quantity is one or None its the same
        transferable=True,
        expiry=None,
        verifiable=True,
        attributes={'number': 'OMV 336048', 'year': '1981'}
       )
print(req_id, response, return_val_uuid)
```

```
 11-05 20:02:27 | I | Issuing instrument='Digital Currency INR 500
Note' to 'Alice'
 11-05 20:02:27 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 11-05 20:03:19 | I | 5819d61a-5d73-44ca-b677-7125ebc8bf97
```

[46]:

```
print(req_id, response, return_val_uuid)
```

```
13652.1683815547.127303 <Response [200]> 5819d61a-5d73-44ca-b677-7125ebc8bf97
```

[ ]:

[ ]:

[28]:

```
# Example 5 -- Travel Points, Redeemable Tokens (Type 3 / Test 3 -- Non␣
 ↪transferable, Quantifiable Instrument)
h.issue('Authority', 'Alice', 'Star Alliance Travel Points', quantity=10,␣
 ↪transferable=False, expiry=None, verifiable=True)
```

```
 09-05 22:57:41 | I | Issuing instrument='Star Alliance Travel
Points' to 'Alice'
 09-05 22:57:41 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 22:57:41 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/069C8B8CD60D'
 09-05 22:57:41 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Issue',
'requestBody': {'name': 'Star Alliance Travel Points', 'to': 'CN=Alice, OU=Test
Dept, O=R3, L=London, C=GB', 'transferable': 'false', 'verifiable': 'true',
'quantity': 10}}
```

```
 09-05 22:57:41 | D | {'clientRequestId':
'76808.1683653261.445578', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Star Alliance Travel Points', 'to': 'CN=Alice, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'false', 'verifiable': 'true', 'quantity':
10}}
 09-05 22:57:41 | D | After filling values = {'clientRequestId':
'76808.1683653261.445578', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Issue', 'requestBody':
{'name': 'Star Alliance Travel Points', 'to': 'CN=Alice, OU=Test Dept, O=R3,
L=London, C=GB', 'transferable': 'false', 'verifiable': 'true', 'quantity':
10}}
 09-05 22:57:41 | D | 202
 09-05 22:57:41 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T17:27:41.489Z'}
 09-05 22:57:45 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:57:45 | D | 200
 09-05 22:57:45 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:57:49 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:57:49 | D | 200
 09-05 22:57:49 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:57:53 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:57:53 | D | 200
 09-05 22:57:53 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:57:57 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:57:57 | D | 200
 09-05 22:57:57 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:01 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:01 | D | 200
 09-05 22:58:01 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:05 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:05 | D | 200
 09-05 22:58:05 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:09 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:09 | D | 200
 09-05 22:58:09 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:13 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:13 | D | 200
 09-05 22:58:13 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:17 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:17 | D | 200
 09-05 22:58:17 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:22 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:22 | D | 200
 09-05 22:58:22 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 22:58:26 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 22:58:26 | D | 200
 09-05 22:58:26 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 23:13:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 23:13:56 | D | 200
 09-05 23:13:56 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 23:28:00 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 23:28:00 | D | 200
 09-05 23:28:00 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 23:28:04 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 23:28:04 | D | 200
 09-05 23:28:04 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 09-05 23:45:54 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 09-05 23:45:54 | D | 200
 09-05 23:45:54 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:03:28 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:03:28 | D | 200
 10-05 00:03:28 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:03:32 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:03:32 | D | 200
 10-05 00:03:33 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:20:02 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:20:02 | D | 200
 10-05 00:20:02 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:36:14 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:36:14 | D | 200
 10-05 00:36:14 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:53:08 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:53:08 | D | 200
 10-05 00:53:08 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 00:53:12 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 00:53:12 | D | 200
 10-05 00:53:12 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:08:27 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:08:27 | D | 200
 10-05 01:08:27 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:08:31 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:08:31 | D | 200
 10-05 01:08:31 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:24:33 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:24:33 | D | 200
 10-05 01:24:33 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:40:19 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:40:19 | D | 200
 10-05 01:40:19 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:58:02 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:58:02 | D | 200
 10-05 01:58:02 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 01:58:06 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 01:58:06 | D | 200
 10-05 01:58:06 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 02:15:31 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 02:15:31 | D | 200
 10-05 02:15:31 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 02:31:53 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 02:31:53 | D | 200
 10-05 02:31:53 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 02:49:53 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 02:49:53 | D | 200
 10-05 02:49:53 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 02:49:57 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 02:49:57 | D | 200
 10-05 02:49:57 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 03:06:54 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 03:06:54 | D | 200
 10-05 03:06:54 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 03:24:17 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 03:24:17 | D | 200
 10-05 03:24:17 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 03:42:14 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 03:42:14 | D | 200
 10-05 03:42:14 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 03:42:18 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 03:42:18 | D | 200
 10-05 03:42:18 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 03:59:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 03:59:56 | D | 200
 10-05 03:59:56 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 04:16:46 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 04:16:46 | D | 200
 10-05 04:16:46 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 04:32:59 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 04:32:59 | D | 200
 10-05 04:32:59 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 04:33:03 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 04:33:03 | D | 200
 10-05 04:33:03 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:27:41.891Z'}
 10-05 04:50:03 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653261.445578'
 10-05 04:50:03 | D | 200
 10-05 04:50:03 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653261.445578', 'flowId':
'251611e4-83dd-426d-87fa-493c0358abda', 'flowStatus': 'COMPLETED', 'flowResult':
"Flow failed, message: Type='PLATFORM_ERROR' Message='Failed to send: 1 of 1
sessions are invalid ['1107bd8e-c58e-4ab7-9e88-dbae20503292'=ERROR]. Please
check CorDapp protocol expectations match on both sides regarding 'send' and
'receive' statements and that sessions are not closed unexpectedly on either
side. Stack traces around this error originating in CorDapp code may provide a
hint.'", 'flowError': None, 'timestamp': '2023-05-09T23:20:00.128Z'}
 10-05 04:50:03 | I | Flow failed, message: Type='PLATFORM_ERROR'
Message='Failed to send: 1 of 1 sessions are invalid
['1107bd8e-c58e-4ab7-9e88-dbae20503292'=ERROR]. Please check CorDapp protocol
expectations match on both sides regarding 'send' and 'receive' statements and
that sessions are not closed unexpectedly on either side. Stack traces around
this error originating in CorDapp code may provide a hint.'
```

```
[28]: ('76808.1683653261.445578',
        <Response [200]>,
        "Flow failed, message: Type='PLATFORM_ERROR' Message='Failed to send: 1 of 1
       sessions are invalid ['1107bd8e-c58e-4ab7-9e88-dbae20503292'=ERROR]. Please
       check CorDapp protocol expectations match on both sides regarding 'send' and
       'receive' statements and that sessions are not closed unexpectedly on either
       side. Stack traces around this error originating in CorDapp code may provide a
hint.'")
```

[9]:
```
# Example 6 -- Amazon Vouchers (# Type 3 / Test 3 -- Non transferable,␣
 ↪Quantifiable Instrument)
h.issue('Authority', 'Alice', 'Amazon Voucher', quantity=2000,␣
 ↪transferable=False, expiry=None, verifiable=True, attributes={'currency':␣
 ↪'INR'})
```

```
09-05 11:18:48 | I | Issuing instrument='Amazon Voucher' to
'Alice'
09-05 11:18:48 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
09-05 11:19:49 | I | 239ca025-092e-4b0b-a1e5-1fca6003b657
```

[9]:
```
('54796.1683611328.611851',
 <Response [200]>,
 '239ca025-092e-4b0b-a1e5-1fca6003b657')
```

[14]:
```
# Example 7 - Graduation certificate (Type 4 / Test 4 -- Non transferable, Non␣
 ↪quantifiable Instrument)
h.issue('Authority', 'Charlie', 'BE Computer Engineering', quantity=None,␣
 ↪transferable=False, expiry=None, verifiable=True,
        attributes={
            'university': 'University of Mumbai',
            'year': '2023',
            'score': '450/500',
            'major subject': "['Robotics', 'AI', 'DSP']"})
```

```
09-05 11:21:37 | I | Issuing instrument='BE Computer Engineering'
to 'Charlie'
09-05 11:21:37 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
09-05 11:22:37 | I | 6c764908-fdb0-40f3-af53-18f0159c6897
```

[14]:
```
('54796.1683611497.236011',
 <Response [200]>,
 '6c764908-fdb0-40f3-af53-18f0159c6897')
```

[16]:
```
# Example 8 - Experience/Participation certificate (Type 4 / Test 4 -- Non␣
 ↪transferable, Non quantifiable Instrument)
h.issue('Authority', 'Bob', 'R3 Hackathon 2023', quantity=None,␣
 ↪transferable=False, expiry=None, verifiable=True, attributes={'Project':␣
 ↪'Corda 5', 'Year': '2023'})
```

```
09-05 11:23:16 | I | Issuing instrument='R3 Hackathon 2023' to
'Bob'
```

```
 09-05 11:23:16 | I | Running action 'Issue' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 11:24:17 | I | 2af47ace-c52c-4237-b336-b2694e75494a
```

```
[16]: ('54796.1683611596.371066',
       <Response [200]>,
       '2af47ace-c52c-4237-b336-b2694e75494a')
```

```
[17]: h.query_all_nodes() # calling flow = ListInstrument
```

```
 09-05 16:08:34 | D | Running query=ListInstrument on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 16:08:34 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/069C8B8CD60D'
 09-05 16:08:34 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 09-05 16:08:34 | D | {'clientRequestId':
'76808.1683628714.729872', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:34 | D | After filling values = {'clientRequestId':
'76808.1683628714.729872', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:34 | D | 202
 09-05 16:08:34 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683628714.729872', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T10:38:34.763Z'}
 09-05 16:08:38 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683628714.729872'
 09-05 16:08:38 | D | 200
 09-05 16:08:38 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683628714.729872', 'flowId':
'9c59e197-1e39-42b6-bef7-04debe84c61b', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":2,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T10:38:36.340Z'}
 09-05 16:08:38 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683628714.729872'
 09-05 16:08:38 | D | 200
 09-05 16:08:38 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683628714.729872', 'flowId':
'9c59e197-1e39-42b6-bef7-04debe84c61b', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":2,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T10:38:36.340Z'}
 09-05 16:08:38 | I | Result for query=ListInstrument from
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'

                                     id                  name  \
0  9339350b-d16e-4a94-91ac-e363902172a8  Government Bond 2024

                                          owner  \
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         2          True

  expiry  verifiable  \
0   None        True

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

```
 09-05 16:08:38 | D | Running query=ListInstrument on
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 16:08:38 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/83214F5048CD'
 09-05 16:08:38 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 09-05 16:08:38 | D | {'clientRequestId':
'76808.1683628718.856797', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:38 | D | After filling values = {'clientRequestId':
'76808.1683628718.856797', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:38 | D | 202
 09-05 16:08:38 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683628718.856797', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T10:38:38.877Z'}
 09-05 16:08:42 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/83214F5048CD/76808.1683628718.856797'
 09-05 16:08:42 | D | 200
 09-05 16:08:42 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683628718.856797', 'flowId':
'22c4a65e-fc55-4c4f-b079-691733819694', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-09T10:38:41.239Z'}
 09-05 16:08:42 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/83214F5048CD/76808.1683628718.856797'
 09-05 16:08:42 | D | 200
 09-05 16:08:43 | D | {'holdingIdentityShortHash': '83214F5048CD',
'clientRequestId': '76808.1683628718.856797', 'flowId':
'22c4a65e-fc55-4c4f-b079-691733819694', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-09T10:38:41.239Z'}
 09-05 16:08:43 | I | Result for query=ListInstrument from
'Bob/83214F5048CD/CN=Bob, OU=Test Dept, O=R3, L=London, C=GB'
```

```
Empty DataFrame
Columns: []
Index: []
 09-05 16:08:43 | D | Running query=ListInstrument on
'Charlie/D728EEE9116F/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 16:08:43 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/D728EEE9116F'
 09-05 16:08:43 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 09-05 16:08:43 | D | {'clientRequestId':
'76808.1683628723.005763', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:43 | D | After filling values = {'clientRequestId':
'76808.1683628723.005763', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:43 | D | 202
 09-05 16:08:43 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683628723.005763', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T10:38:43.027Z'}
 09-05 16:08:47 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/D728EEE9116F/76808.1683628723.005763'
 09-05 16:08:47 | D | 200
 09-05 16:08:47 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683628723.005763', 'flowId':
'9e7d3082-f02c-497a-a9a6-df198c37c2c3', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-09T10:38:45.408Z'}
 09-05 16:08:47 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/D728EEE9116F/76808.1683628723.005763'
 09-05 16:08:47 | D | 200
 09-05 16:08:47 | D | {'holdingIdentityShortHash': 'D728EEE9116F',
'clientRequestId': '76808.1683628723.005763', 'flowId':
'9e7d3082-f02c-497a-a9a6-df198c37c2c3', 'flowStatus': 'COMPLETED', 'flowResult':
'[]', 'flowError': None, 'timestamp': '2023-05-09T10:38:45.408Z'}
 09-05 16:08:47 | I | Result for query=ListInstrument from
'Charlie/D728EEE9116F/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
Empty DataFrame
Columns: []
Index: []
 09-05 16:08:47 | D | Running query=ListInstrument on
'Alice/98A345A3E88B/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 16:08:47 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/98A345A3E88B'
 09-05 16:08:47 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 09-05 16:08:47 | D | {'clientRequestId':
'76808.1683628727.147019', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:47 | D | After filling values = {'clientRequestId':
'76808.1683628727.147019', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 16:08:47 | D | 202
 09-05 16:08:47 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683628727.147019', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T10:38:47.171Z'}
 09-05 16:08:51 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/98A345A3E88B/76808.1683628727.147019'
 09-05 16:08:51 | D | 200
 09-05 16:08:51 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683628727.147019', 'flowId':
'567cc149-9e54-4085-864d-f26881786749', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":5,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T10:38:49.514Z'}
 09-05 16:08:51 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/98A345A3E88B/76808.1683628727.147019'
 09-05 16:08:51 | D | 200
 09-05 16:08:51 | D | {'holdingIdentityShortHash': '98A345A3E88B',
'clientRequestId': '76808.1683628727.147019', 'flowId':
'567cc149-9e54-4085-864d-f26881786749', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":5,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T10:38:49.514Z'}
 09-05 16:08:51 | I | Result for query=ListInstrument from
'Alice/98A345A3E88B/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
                                     id                  name  \
0  9339350b-d16e-4a94-91ac-e363902172a8  Government Bond 2024

                                          owner  \
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB

                                             issuer  quantity  transferable  \
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         5          True

  expiry  verifiable  \
0   None        True

                                                             attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[8]:

```python
df = h.query('Alice', 'ListInstrument')
```

[22]:

```python
df = h.query('Authority')
# inst_id='7e7bd79d-d5d6-4c9d-954b-9a47e2dc517d'
# tdf = df[df['id'] == inst_id]
# tdf.transpose()
# tdf.to_dict('records')[0]
# del(tdf['id'])
```

```
 09-05 22:53:46 | D | Running query=ListInstrument on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 22:53:46 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/069C8B8CD60D'
 09-05 22:53:46 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.query.__QUERY__',
'requestBody': {}}
 09-05 22:53:46 | D | {'clientRequestId':
'76808.1683653026.647611', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 22:53:46 | D | After filling values = {'clientRequestId':
'76808.1683653026.647611', 'flowClassName':
'com.r3.developers.configurableInstrument.query.ListInstrument', 'requestBody':
{}}
 09-05 22:53:46 | D | 202
 09-05 22:53:46 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653026.647611', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T17:23:46.679Z'}
 09-05 22:53:50 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653026.647611'
 09-05 22:53:50 | D | 200
 09-05 22:53:50 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653026.647611', 'flowId':
'936213a2-c57e-49ab-93f9-429ee1c4feb0', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":0,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T17:23:48.361Z'}
 09-05 22:53:50 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683653026.647611'
 09-05 22:53:50 | D | 200
 09-05 22:53:50 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683653026.647611', 'flowId':
'936213a2-c57e-49ab-93f9-429ee1c4feb0', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"9339350b-d16e-4a94-91ac-e363902172a8","name":"Government Bond
2024","owner":"CN=Alice, OU=Test Dept, O=R3, L=London,
C=GB","issuer":"CN=Authority, OU=Test Dept, O=R3, L=London, C=GB","quantity":0,"
transferable":true,"expiry":null,"verifiable":true,"attributes":{"payments":"[\'
10Jun2023\', \'10Sep2023\', \'10Dec2023\', \'10Mar2024\']"}}]', 'flowError':
None, 'timestamp': '2023-05-09T17:23:48.361Z'}
```

[9]:

```
df
```

[9]:

```
                                     id                   name  
0  c93b9892-7416-4ae2-9ced-681e326e241d  Government Bond 2024  \
1  3d4c026b-7721-4f1d-97e4-5b823c61c568  Government Bond 2024   

                                      owner  
0  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB  \
1  CN=Alice, OU=Test Dept, O=R3, L=London, C=GB   

                                         issuer  quantity  transferable  
0  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB        10          True  \
1  CN=Authority, OU=Test Dept, O=R3, L=London, C=GB         4          True   

  expiry  verifiable  
0   None        True  \
1   None        True   
```

```
                                                                attributes
0  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
1  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023', '10Mar2024']'}
```

[21]:

```
h.redeem('9339350b-d16e-4a94-91ac-e363902172a8', from_node='Authority',␣
  ↪quantity=2)
```

```
 09-05 22:52:19 | I | Redeeming instrument id
'9339350b-d16e-4a94-91ac-e363902172a8' quantity '2'
 09-05 22:52:19 | I | Running action 'Redeem' on
'Authority/069C8B8CD60D/CN=Authority, OU=Test Dept, O=R3, L=London, C=GB'
 09-05 22:52:19 | D | Request type=post to
url='https://localhost:8888/api/v1/flow/069C8B8CD60D'
 09-05 22:52:19 | D | Original Request = {'clientRequestId':
'__REQUEST_NUMBER__', 'flowClassName': '__PACKAGE__.workflows.Redeem',
'requestBody': {'id': '__REDEEM_ASSET_ID__', 'quantity': 2}}
 09-05 22:52:19 | D | {'clientRequestId':
'76808.1683652939.960599', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '9339350b-d16e-4a94-91ac-e363902172a8', 'quantity': 2}}
 09-05 22:52:19 | D | After filling values = {'clientRequestId':
'76808.1683652939.960599', 'flowClassName':
'com.r3.developers.configurableInstrument.workflows.Redeem', 'requestBody':
{'id': '9339350b-d16e-4a94-91ac-e363902172a8', 'quantity': 2}}
 09-05 22:52:20 | D | 202
 09-05 22:52:20 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-05-09T17:22:20.003Z'}
 09-05 22:52:24 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:24 | D | 200
 09-05 22:52:24 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
```

```
 09-05 22:52:28 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:28 | D | 200
 09-05 22:52:28 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:32 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:32 | D | 200
 09-05 22:52:32 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:36 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:36 | D | 200
 09-05 22:52:36 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:40 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:40 | D | 200
 09-05 22:52:40 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:44 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:44 | D | 200
 09-05 22:52:44 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
```

```
 09-05 22:52:48 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:48 | D | 200
 09-05 22:52:48 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:52 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:52 | D | 200
 09-05 22:52:52 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:52:56 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:52:56 | D | 200
 09-05 22:52:56 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:53:00 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:53:00 | D | 200
 09-05 22:53:00 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:53:04 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:53:04 | D | 200
 09-05 22:53:04 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
```

```
 09-05 22:53:08 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:53:08 | D | 200
 09-05 22:53:08 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'RUNNING', 'flowResult':
None, 'flowError': None, 'timestamp': '2023-05-09T17:22:20.977Z'}
 09-05 22:53:12 | D | Request type=get to url='https://localhost:8
888/api/v1/flow/069C8B8CD60D/76808.1683652939.960599'
 09-05 22:53:12 | D | 200
 09-05 22:53:12 | D | {'holdingIdentityShortHash': '069C8B8CD60D',
'clientRequestId': '76808.1683652939.960599', 'flowId':
'4261d86e-8cc6-4cfb-842f-5b60e218a308', 'flowStatus': 'COMPLETED', 'flowResult':
'UtxoSignedTransactionImpl(id=SHA-
256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7, signature
s=[DigitalSignatureAndMetadata[signature=[304502202A2AC7A2A8C5CC8EBD17C31B3C9D9A
085125CF305944C695B4C5146C0CD7D4690221009CF759387E0F3F59099F5B71DE3E15473B35CA38
37522FFE1928D042792E17F4],
metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:27.505471Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3045022100B53F29E782CB834CA9
D9E35EB35E568BA60E3DE909805C7F8885CD68C9337FE70220493E393C4ADC4A3F6E6830555901B4
CC04711A180D4DB89247C51C359614B687],
metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:56.059090Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@535f4866]], wireTransaction=W
ireTransaction(id=SHA-
256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7,
privacySalt=[00449DCB5FA857C9F01E46CB74E19CE2688409EDB45FC2DA0E0DC82920B95ED4],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62812),
(size=868, sum=55221)], [(size=295, sum=11163), (size=295, sum=11163)],
[(size=2082, sum=123221)], [(size=556, sum=34280)], [], [(size=774, sum=41246)],
[], [(size=3358, sum=197055)], [(size=790, sum=52841)]]))', 'flowError': None,
'timestamp': '2023-05-09T17:23:09.909Z'}
```

```
 09-05 22:53:12 | I | UtxoSignedTransactionImpl(id=SHA-
256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7, signature
s=[DigitalSignatureAndMetadata[signature=[304502202A2AC7A2A8C5CC8EBD17C31B3C9D9A
085125CF305944C695B4C5146C0CD7D4690221009CF759387E0F3F59099F5B71DE3E15473B35CA38
37522FFE1928D042792E17F4],
metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:27.505471Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3045022100B53F29E782CB834CA9
D9E35EB35E568BA60E3DE909805C7F8885CD68C9337FE70220493E393C4ADC4A3F6E6830555901B4
CC04711A180D4DB89247C51C359614B687],
metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:56.059090Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@535f4866]], wireTransaction=W
ireTransaction(id=SHA-
256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7,
privacySalt=[00449DCB5FA857C9F01E46CB74E19CE2688409EDB45FC2DA0E0DC82920B95ED4],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62812),
(size=868, sum=55221)], [(size=295, sum=11163), (size=295, sum=11163)],
[(size=2082, sum=123221)], [(size=556, sum=34280)], [], [(size=774, sum=41246)],
[], [(size=3358, sum=197055)], [(size=790, sum=52841)]]))
```

```
[21]: ('76808.1683652939.960599',
       <Response [200]>,
       'UtxoSignedTransactionImpl(id=SHA-
      256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7, signature
      s=[DigitalSignatureAndMetadata[signature=[304502202A2AC7A2A8C5CC8EBD17C31B3C9D9A
      085125CF305944C695B4C5146C0CD7D4690221009CF759387E0F3F59099F5B71DE3E15473B35CA38
      37522FFE1928D042792E17F4],
      metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:27.505471Z,
      signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
      name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
      proof=null], DigitalSignatureAndMetadata[signature=[3045022100B53F29E782CB834CA9
      D9E35EB35E568BA60E3DE909805C7F8885CD68C9337FE70220493E393C4ADC4A3F6E6830555901B4
      CC04711A180D4DB89247C51C359614B687],
      metadata=DigitalSignatureMetadata[timestamp=2023-05-09T17:22:56.059090Z,
      signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
      name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
      batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
      batchMerkleTreeDigestAlgorithmName=SHA-256D,
      batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
      batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
      proof=net.corda.crypto.merkle.impl.MerkleProofImpl@535f4866]], wireTransaction=W
      ireTransaction(id=SHA-
      256D:5A97A6A78890E68C79B2EF0DA3B063B9C48A5DE039CC748E27DCB5DB5F6306A7,
      privacySalt=[00449DCB5FA857C9F01E46CB74E19CE2688409EDB45FC2DA0E0DC82920B95ED4],
      metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@65c412
      69, componentGroupLists=[[(size=1278, sum=108978)], [(size=1210, sum=62812),
      (size=868, sum=55221)], [(size=295, sum=11163), (size=295, sum=11163)],
      [(size=2082, sum=123221)], [(size=556, sum=34280)], [], [(size=774, sum=41246)],
      [], [(size=3358, sum=197055)], [(size=790, sum=52841)]]))')
```

```
[10]: pdf_report = h.print_to_pdf('Alice', '3d4c026b-7721-4f1d-97e4-5b823c61c568',␣
        ↪show=False)
      webbrowser.open("file:///{}".format(pdf_report))
```

```
[WDM] - Downloading: 100%|                              |
8.06M/8.06M [00:00<00:00, 19.4MB/s]
 17-05 14:09:25 | I | Time taken = 0:00:09.215990
```

```
[10]: True
```

```
[10]: pdf_report = h.print_to_pdf('Charlie', 'ab2a0ff2-92c4-4b73-af50-5fed62790214',␣
        ↪show=False)
      webbrowser.open("file:///{}".format(pdf_report))
```

```
[10]: True
```

```
[9]: h.transfer('6c764908-fdb0-40f3-af53-18f0159c6897', 'Charlie', 'Bob')
```

```
 09-05 11:57:24 | I | Tranfering instrument id
'6c764908-fdb0-40f3-af53-18f0159c6897' to 'Bob'
 09-05 11:57:24 | I | Running action 'Transfer' on
'Charlie/D728EEE9116F/CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 09-05 11:57:33 | I | {'type': 'FLOW_FAILED', 'message':
'net.corda.v5.ledger.utxo.ContractVerificationException: Verification of ledger
transaction with ID
SHA-256D:A2EBCEADF36EB733619F98987E3400B42E75CA93CE2998CF63FB2779D2A303BE
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:A2EBCEADF36EB733619F98987E3400B42E75CA93CE2998CF63FB2779D2A303BE.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'}
```

[9]:

```
('57457.1683613644.99713',
 <Response [200]>,
 {'type': 'FLOW_FAILED',
  'message': 'net.corda.v5.ledger.utxo.ContractVerificationException:
Verification of ledger transaction with ID
SHA-256D:A2EBCEADF36EB733619F98987E3400B42E75CA93CE2998CF63FB2779D2A303BE
failed: net.corda.v5.ledger.utxo.ContractVerificationException: Ledger
transaction contract verification failed for the specified transaction:
SHA-256D:A2EBCEADF36EB733619F98987E3400B42E75CA93CE2998CF63FB2779D2A303BE.\nThe
following contract verification requirements were not
met:\ncom.r3.developers.configurableInstrument.contracts.InstrumentContract:
Failed requirement: Instrument is not transferable.\n'})
```

[8]:

```
h.nodes
```

[8]:

```
{'Authority': {'x500': 'CN=Authority, OU=Test Dept, O=R3, L=London, C=GB',
  'hash': '069C8B8CD60D'},
 'Bob': {'x500': 'CN=Bob, OU=Test Dept, O=R3, L=London, C=GB',
  'hash': '83214F5048CD'},
 'Charlie': {'x500': 'CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB',
  'hash': 'D728EEE9116F'},
 'Alice': {'x500': 'CN=Alice, OU=Test Dept, O=R3, L=London, C=GB',
  'hash': '98A345A3E88B'}}
```

[ ]:

```

```

[5]:

```
h.issue('Dave', 'Alice', 'Government Bond 2025', quantity=10,␣
 ↪transferable=True, expiry=None, verifiable=True, attributes={'payments':␣
 ↪"['10Jun2023', '10Sep2023', '10Dec2023']"})
# ('67461.1681894831.8281841',
#  <Response [200]>,
#  'fe1818dd-0a03-40dd-b367-e71813aad485')
# h.issue('Dave', 'Charlie', 'BE Computers', quantity = None,␣
 ↪transferable=False, expiry=None, verifiable=True, attributes={'Year':␣
 ↪"2003", 'Name': 'Bob Gupta', 'Marks': '451/500'})
```

```
26-04 12:57:09 | I | Issuing instrument='Government Bond 2025' to
'Alice'
26-04 12:57:09 | I | Running action 'Issue' on
'Dave/272937EA30B2/CN=Dave, OU=Test Dept, O=R3, L=London, C=GB'
26-04 12:58:54 | I | 4df1ef96-eeb0-44bf-935e-48cef48753e6
```

[5]:
```
('45337.1682494029.1437619',
 <Response [200]>,
 '4df1ef96-eeb0-44bf-935e-48cef48753e6')
```

[6]:

```
Help on method transfer in module json_on_jupyter:

transfer(inst_id, from_node, to_node, quantity=None) method of
json_on_jupyter.JSON_ON_Jupyter instance
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[32]:

[26]:

[26]:
```
                                     id                 name  \
0  7e7bd79d-d5d6-4c9d-954b-9a47e2dc517d  Government Bond 2025

                                          owner  \
0  CN=Charlie, OU=Test Dept, O=R3, L=London, C=GB

                                       issuer  quantity  transferable expiry  \
0  CN=Dave, OU=Test Dept, O=R3, L=London, C=GB         3          True   None
```

```
   verifiable                                            attributes
0        True  {'payments': '['10Jun2023', '10Sep2023', '10Dec2023']'}
```

[19]:

```

```

[19]: 128

[6]:

```
# h.transfer('fdd47076-7f6a-4ea1-9d1f-c663d9daf97f', 'Alice', 'Bob') #␣
 ↪transferable=False
# h.transfer('0697bc47-b1f0-43f5-a37f-841fc353925e', 'Alice', 'Bob') #␣
 ↪transferable=True
h.transfer('ab2a0ff2-92c4-4b73-af50-5fed62790214', 'Alice', 'Charlie', 3)
```

```
 25-04 11:47:50 | I | Tranfering instrument id
'ab2a0ff2-92c4-4b73-af50-5fed62790214' to 'Charlie'
 25-04 11:47:50 | I | Running action 'Transfer' on
'Alice/45C63C50EA0B/CN=Alice, OU=Test Dept, O=R3, L=London, C=GB'
```

```
 25-04 11:49:28 | I | UtxoSignedTransactionImpl(id=SHA-
256D:017301B55A8F5B3FC454189BF45BCCC4F3744A23E61E16D80B6A62F83DEFB270, signature
s=[DigitalSignatureAndMetadata[signature=[304402203FA9A03DB18B95EB183B7DDDB12C5A
6EAF25603DB09149079869A7E1494F939102201C76857B6B47BE6C0D0F874AB66FF1BA327957129E
37F6CA693C9AF5E176C711],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:17:56.621447Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[30450220431OC67DFAB3B1FEB2FE
489A460AD24E5D5ADB4CBC6DFA033203FD9CDEEBFE00022100D8924D51BF688CF2D7F945D4735A65
0A3285FB9A5039545FBD80AC95C838325B],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:18:51.973001Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3046022100B9D372C348B3D53889
59DF90552B29B4E55B49EACFBE776055AEECE6C87E7BEC022100B3E551CC1870A1B7F43E01EB1BF6
63140ADB3A2DEC7A12374F1CC2B5005A06D7],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:19:06.702039Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@349c9e78]], wireTransaction=W
ireTransaction(id=SHA-
256D:017301B55A8F5B3FC454189BF45BCCC4F3744A23E61E16D80B6A62F83DEFB270,
privacySalt=[CDD748372F9DDB4D07512BD41BDD0227FCE6035062517389859543 3E48BF6DBE],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@5ce3ab
78, componentGroupLists=[[(size=1278, sum=109133)], [(size=1210, sum=62100),
(size=868, sum=55065)], [(size=295, sum=10379), (size=295, sum=10684)],
[(size=2082, sum=122509), (size=2082, sum=122509)], [(size=556, sum=34280)], [],
[(size=774, sum=39657)], [], [(size=3340, sum=194643), (size=3342, sum=194946)],
[(size=794, sum=52838)]]))
```

[6]:
```
('31099.1682403470.883976',
 <Response [200]>,
 'UtxoSignedTransactionImpl(id=SHA-
256D:017301B55A8F5B3FC454189BF45BCCC4F3744A23E61E16D80B6A62F83DEFB270, signature
s=[DigitalSignatureAndMetadata[signature=[304402203FA9A03DB18B95EB183B7DDDB12C5A
6EAF25603DB09149079869A7E1494F939102201C76857B6B47BE6C0D0F874AB66FF1BA327957129E
37F6CA693C9AF5E176C711],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:17:56.621447Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[304502204310C67DFAB3B1FEB2FE
489A460AD24E5D5ADB4CBC6DFA033203FD9CDEEBFE00022100D8924D51BF688CF2D7F945D4735A65
0A3285FB9A5039545FBD80AC95C838325B],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:18:51.973001Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565}],
proof=null], DigitalSignatureAndMetadata[signature=[3046022100B9D372C348B3D53889
59DF90552B29B4E55B49EACFBE776055AEECE6C87E7BEC022100B3E551CC1870A1B7F43E01EB1BF6
63140ADB3A2DEC7A12374F1CC2B5005A06D7],
metadata=DigitalSignatureMetadata[timestamp=2023-04-25T06:19:06.702039Z,
signatureSpec=SHA256withECDSA, properties={platformVersion=5000, cpiName=CPI
name, cpiVersion=CPI version, cpiSignerSummaryHash=46616B652D76616C7565,
batchMerkleTreeDigestProviderName=TweakableHashDigestProvider,
batchMerkleTreeDigestAlgorithmName=SHA-256D,
batchMerkleTreeDigestOptionsLeafPrefixB64=YmF0Y2hMZWFm,
batchMerkleTreeDigestOptionsNodePrefixB64=YmF0Y2hOb2Rl}],
proof=net.corda.crypto.merkle.impl.MerkleProofImpl@349c9e78]], wireTransaction=W
ireTransaction(id=SHA-
256D:017301B55A8F5B3FC454189BF45BCCC4F3744A23E61E16D80B6A62F83DEFB270,
privacySalt=[CDD748372F9DDB4D07512BD41BDD0227FCE6035062517389859543 3E48BF6DBE],
metadata=net.corda.ledger.common.data.transaction.TransactionMetadataImpl@5ce3ab
78, componentGroupLists=[[(size=1278, sum=109133)], [(size=1210, sum=62100),
(size=868, sum=55065)], [(size=295, sum=10379), (size=295, sum=10684)],
[(size=2082, sum=122509), (size=2082, sum=122509)], [(size=556, sum=34280)], [],
[(size=774, sum=39657)], [], [(size=3340, sum=194643), (size=3342, sum=194946)],
[(size=794, sum=52838)]]))')
```

[3]:
```
h.get("Dave", 30020)
```

```
17-04 11:18:18 | I | 200
```

```
17-04 11:18:18 | I | {'holdingIdentityShortHash': '272937EA30B2',
'clientRequestId': '30020', 'flowId': '5af38dc7-0991-4811-86dc-b2c4239a6bf3',
'flowStatus': 'COMPLETED', 'flowResult': '[{"id":"fdd47076-7f6a-4ea1-9d1f-
c663d9daf97f","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple Collage","university"
:"Apple"}},{"id":"abadf72d-a7b1-43e2-a122-
5749dffbee58","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple
Collage","university":"Apple"}}]', 'flowError': None, 'timestamp':
'2023-04-15T04:41:10.956Z'}
```

```
[3]: (30020, <Response [200]>)
```

```
[5]: h.log.info("apple")
```

```
17-04 11:03:19 | I | apple
```

```
[4]: (req, res, result) = h.post('Dave', queryRequest)
     h.get("Dave", 30020)
```

```
 17-04 11:18:39 | I | 202
 17-04 11:18:39 | I | {'holdingIdentityShortHash': '272937EA30B2',
'clientRequestId': '46048.1681710519.2324579', 'flowId': None, 'flowStatus':
'START_REQUESTED', 'flowResult': None, 'flowError': None, 'timestamp':
'2023-04-17T05:48:39.270Z'}
 17-04 11:18:44 | I | 200
 17-04 11:18:44 | I | {'holdingIdentityShortHash': '272937EA30B2',
'clientRequestId': '46048.1681710519.2324579', 'flowId':
'cc8c2422-1dc3-44a1-8421-dafc438f165c', 'flowStatus': 'COMPLETED', 'flowResult':
'[{"id":"fdd47076-7f6a-4ea1-9d1f-
c663d9daf97f","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple Collage","university"
:"Apple"}},{"id":"abadf72d-a7b1-43e2-a122-
5749dffbee58","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple
Collage","university":"Apple"}}]', 'flowError': None, 'timestamp':
'2023-04-17T05:48:41.621Z'}
```

```
 17-04 11:18:44 | I | 200
 17-04 11:18:44 | I | {'holdingIdentityShortHash': '272937EA30B2',
'clientRequestId': '30020', 'flowId': '5af38dc7-0991-4811-86dc-b2c4239a6bf3',
'flowStatus': 'COMPLETED', 'flowResult': '[{"id":"fdd47076-7f6a-4ea1-9d1f-
c663d9daf97f","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple Collage","university"
:"Apple"}},{"id":"abadf72d-a7b1-43e2-a122-
5749dffbee58","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple
Collage","university":"Apple"}}]', 'flowError': None, 'timestamp':
'2023-04-15T04:41:10.956Z'}
```

```
[4]: (30020, <Response [200]>)
```

```
[5]: (req, res) = h.get("Dave", 30020)
```

```
 17-04 11:19:21 | I | 200
 17-04 11:19:21 | I | {'holdingIdentityShortHash': '272937EA30B2',
'clientRequestId': '30020', 'flowId': '5af38dc7-0991-4811-86dc-b2c4239a6bf3',
'flowStatus': 'COMPLETED', 'flowResult': '[{"id":"fdd47076-7f6a-4ea1-9d1f-
c663d9daf97f","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple Collage","university"
:"Apple"}},{"id":"abadf72d-a7b1-43e2-a122-
5749dffbee58","name":"Certificate","owner":"Party(name=CN=Alice, OU=Test Dept,
O=R3, L=London, C=GB, owningKey=EC Public Key
[ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\\n            X:
f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\\n)","issuer
":"Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\\n)","transf
erable":false,"expiry":null,"attributes":{"college":"Apple
Collage","university":"Apple"}}]', 'flowError': None, 'timestamp':
'2023-04-15T04:41:10.956Z'}
```

[14]:
```
pd.DataFrame(json.loads(res.json()['flowResult']))
```

[14]:
```
                                     id         name  \
0  fdd47076-7f6a-4ea1-9d1f-c663d9daf97f  Certificate
1  abadf72d-a7b1-43e2-a122-5749dffbee58  Certificate

                                                                    owner  \
0  Party(name=CN=Alice, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\n
X: f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\n)
1  Party(name=CN=Alice, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [ab:e4:40:46:6a:26:a1:3d:60:70:1b:1e:68:a0:81:a9:ca:0f:d2:96]\n
X: f141c3b2973dce02e8cfeeb3ada735652ce78b55a348b417481c9c6857c6432c\n
Y: f766dfa43075e5af7c3c89c5aad7af883e9b8e5330a65432240d6d3151daea16\n)

                                                                 issuer  \
0  Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\n)
1  Party(name=CN=Dave, OU=Test Dept, O=R3, L=London, C=GB, owningKey=EC Public
Key [28:43:4d:10:b0:6a:bb:88:8a:38:8d:85:1c:81:93:62:88:be:1b:74]\n
X: 588a1e2d59785a7cf3e37de91f6fbfdcd3af47f5ae62ccf1a84bb4d9365d2a37\n
Y: 5443ac36886a87b17b61df56c2d18d84e15a0fcf3781ff9bea7ccf7890348cab\n)

   transferable expiry                                          attributes
0         False   None  {'college': 'Apple Collage', 'university': 'Apple'}
1         False   None  {'college': 'Apple Collage', 'university': 'Apple'}
```

[1]:

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In [1], line 3
      1 # example 1 -- Government Bond with defined coupon payments -- Type 1 /␣
  ↪Test 1 -- Transferable, Quantifiable Instrument
----> 3 h.issue('Dave', 'Alice', 'Government Bond 2024',
      4         quantity=10,
      5         transferable=True,
      6         expiry=None,
      7         verifiable=True,
      8         attributes={'payments': "['10Jun2023', '10Sep2023', '10Dec2023' ␣
  ↪'10Mar2024']"})

NameError: name 'h' is not defined
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: