

All Commands and Property-Registration Network flow:

1. **Generating certificates, genesis block, channel config, anchor peer, Generating and starting docker container.**

Joining peers to channel, Updating anchor peers,

===== All GOOD, Hyperledger Fabric Property Registration Network Is Now Up and Running! =====

Command to run script:

- Navigate to the property-registration DIR and then cd into network:
- **./fabricNetwork.sh up**

Output:

```

root@upgrad-VirtualBox:/home/upgrad/Desktop/property-registration/network# ./fabricNetwork.sh up
Starting for channel 'registrationchannel' with CLI timeout of '15' seconds and CLI delay of '5' seconds and chaincode version '1.1'
Continue? [Y/n]
proceeding ...
LOCAL_VERSION=1.4.2
DOCKER_IMAGE_VERSION=1.4.3
***** WARNING *****
Local fabric binaries and docker images are
out of sync. This may cause problems.
*****
/home/upgrad/Desktop/property-registration/network/bin/cryptogen

##### Generate certificates using cryptogen tool #####
##### Generate certificates using cryptogen tool #####
+ cryptogen generate --config=./crypto-config.yaml
registrars-property-registration-network.com
users-property-registration-network.com
+ res=0
+ set +x

/home/upgrad/Desktop/property-registration/network/bin/configtxgen
##### Generating Orderer Genesis block #####
##### Generating Orderer Genesis block #####
+ configtxgen -profile OrdererGenesis -channelID upgrad-sys-channel -outputBlock ./channel-artifacts/genesis.block
2020-08-31 01:11:58.482 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2020-08-31 01:11:58.556 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.556 IST [common.tools.configtxgen.localconfig] Load -> INFO 003 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.596 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 004 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.597 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 005 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.600 IST [common.tools.configtxgen] doOutputBlock -> INFO 006 Generating genesis block
2020-08-31 01:11:58.601 IST [common.tools.configtxgen] doOutputBlock -> INFO 007 Writing genesis block
+ res=0
+ set +x

##### Generating channel configuration transaction 'channel.tx' #####
##### Generating channel configuration transaction 'channel.tx' #####
+ configtxgen -profile RegistrationChannel -outputCreateChannelTx ./channel-artifacts/channel.tx -channelID registrationchannel
2020-08-31 01:11:58.716 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2020-08-31 01:11:58.762 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.837 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.837 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.837 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 005 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:58.837 IST [common.tools.configtxgen.encoder] NewChannelGroup -> WARN 006 Default policy emission is deprecated, please include policy specifications for the channel group in configtx.yaml
2020-08-31 01:11:58.839 IST [common.tools.configtxgen.encoder] NewChannelGroup -> WARN 007 Default policy emission is deprecated, please include policy specifications for the channel group in configtx.yaml
2020-08-31 01:11:58.845 IST [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 008 Generating new channel config
2020-08-31 01:11:58.845 IST [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 009 Writing new channel tx
+ res=0
+ set +x

##### Generating anchor peer update for registrar MSP #####
##### Generating anchor peer update for registrar MSP #####
+ configtxgen -profile RegistrationChannel -outputAnchorPeersUpdate ./channel-artifacts/registrarMSPanchors.tx -channelID registrationchannel -asOrg registrarMSP
2020-08-31 01:11:58.913 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2020-08-31 01:11:59.034 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.112 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.114 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.115 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2020-08-31 01:11:59.115 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

```

```

##### Generating anchor peer update for users MSP #####
##### Generating anchor peer update for users MSP #####
+ configtxgen -profile RegistrationChannel -outputAnchorPeersUpdate ./channel-artifacts/usersMSPanchors.tx -channelID registrationchannel -asOrg usersMSP
2020-08-31 01:11:59.290 IST [common.tools.configtxgen] main -> INFO 001 Loading configuration
2020-08-31 01:11:59.290 IST [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.343 IST [common.tools.configtxgen.localconfig] completeInitialization -> INFO 003 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.346 IST [common.tools.configtxgen.localconfig] LoadTopLevel -> INFO 004 Loaded configuration: /home/upgrad/Desktop/property-registration/network/configtx.yaml
2020-08-31 01:11:59.346 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 005 Generating anchor peer update
2020-08-31 01:11:59.346 IST [common.tools.configtxgen] doOutputAnchorPeersUpdate -> INFO 006 Writing anchor peer update
+ res=0
+ set +x

Creating network 'network_registration' with the default driver
Creating volume 'network_orderer.property-registration-network.com' with default driver
Creating volume 'network_peer0.registrar.property-registration-network.com' with default driver
Creating volume 'network_peer1.registrar.property-registration-network.com' with default driver
Creating volume 'network_peer0.users.property-registration-network.com' with default driver
Creating volume 'network_peer1.users.property-registration-network.com' with default driver
Creating volume 'network_peer2.users.property-registration-network.com' with default driver
Creating peer users.property-registration-network.com ... done
Creating peer orderer.property-registration-network.com ... done
Creating peer users.property-registration-network.com ... done
Creating peer0.registrar.property-registration-network.com ... done
Creating peer1.registrar.property-registration-network.com ... done
Creating peer0.users.property-registration-network.com ... done
Creating peer1.users.property-registration-network.com ... done
Creating channelcode ... done
Creating peer0.registrar.property-registration-network.com ... done
Creating peer1.registrar.property-registration-network.com ... done
Creating peer0.users.property-registration-network.com ... done
Creating peer1.users.property-registration-network.com ... done
Creating cli ... done
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
924f6f35c9          hyperledger/fabric-tools:latest    "/bin/bash"        2 seconds ago       Up less than a second    0.0.0.0:7051->7051/tcp, 0.0.0.0:7053->7053/tcp    cli
924f6f35c9          hyperledger/fabric-peer:latest     "peer node start -" 11 seconds ago     Up 2 seconds          0.0.0.0:7051->7051/tcp, 0.0.0.0:7053->7053/tcp    peer0.registrar.property-registration-network.com
0514309cd4          hyperledger/fabric-cscli:latest    "/bin/bash -c 'sleep" 11 seconds ago     Up 5 seconds          0.0.0.0:8051->8051/tcp, 0.0.0.0:8053->8053/tcp    channelcode
a751cd15b87         hyperledger/fabric-peer:latest     "peer node start -" 11 seconds ago     Up 3 seconds          0.0.0.0:9051->9051/tcp, 0.0.0.0:9053->9053/tcp    peer1.registrar.property-registration-network.com
a60b2ab0d4          hyperledger/fabric-peer:latest     "peer node start -" 11 seconds ago     Up 4 seconds          0.0.0.0:11051->11051/tcp, 0.0.0.0:11053->11053/tcp    peer0.users.property-registration-network.com
a0f3a0d6d5          hyperledger/fabric-peer:latest     "peer node start -" 11 seconds ago     Up 5 seconds          0.0.0.0:12051->12051/tcp, 0.0.0.0:12053->12053/tcp    peer1.users.property-registration-network.com
0037801597d         hyperledger/fabric-ca:latest       "sh -c 'fabric-ca-se" 11 seconds ago     Up 4 seconds          0.0.0.0:8054->7054/tcp                        ca.users.property-registration-network.com
1004b0d0d6          hyperledger/fabric-orderer:latest  "orderer"          11 seconds ago     Up 6 seconds          0.0.0.0:7050->7050/tcp                        orderer.property-registration-network.com
81c5bf77b3e         hyperledger/fabric-ca:latest       "sh -c 'fabric-ca-se" 11 seconds ago     Up 3 seconds          0.0.0.0:7054->7054/tcp                        ca.registrar.property-registration-network.com
2735b26318e        hyperledger/fabric-peer:latest     "peer node start -" 11 seconds ago     Up 2 seconds          0.0.0.0:10051->10051/tcp, 0.0.0.0:10053->10053/tcp    peer1.users.property-registration-network.com
Sleeping 10s to allow cluster to complete booting

```

START

Setting up Hyperledger Fabric Network

```

Channel name : registrationchannel
Creating channel...
+ peer channel create -o orderer.property-registration-network.com:7050 -c registrationchannel -f ./channel-artifacts/channel.tx
+ res=0
+ set +x
2020-08-30 19:42:26.843 UTC [channelMgr] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:26.715 UTC [cli.commands] readBlock -> INFO 001 Received block: 0
***** Channel 'registrationchannel' created *****
Having all peers join the channel...
+ peer channel join -o registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:26.864 UTC [channelMgr] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:26.870 UTC [cli.commands] SubmitProposal -> INFO 001 Successfully submitted proposal to join channel
***** peer0.registrar.registration-network.com joined channel 'registrationchannel' *****

```

```

===== Channel 'registrationchannel' created =====

Having all peers join the channel...
+ peer channel join -b registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:26.866 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:26.930 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.registrar.registration-network.com joined channel 'registrationchannel' =====

+ peer channel join -b registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:32.082 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:32.138 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer1.registrar.registration-network.com joined channel 'registrationchannel' =====

+ peer channel join -b registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:37.271 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:37.332 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer0.users.registration-network.com joined channel 'registrationchannel' =====

+ peer channel join -b registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:42.484 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:42.533 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
===== peer1.users.registration-network.com joined channel 'registrationchannel' =====

+ peer channel join -b registrationchannel.block
+ res=0
+ set +x
2020-08-30 19:42:47.682 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:47.739 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
Updating anchor peers for REGISTRAR...
+ peer channel update -o orderer.property-registration-network.com:7050 -c registrationchannel -f ./channel-artifacts/registrarMSPanchors.tx
+ res=0
+ set +x
2020-08-30 19:42:47.863 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:47.880 UTC [channelCmd] update -> INFO 002 Successfully submitted channel update
===== Anchor peers updated for org 'registrarMSP' on channel 'registrationchannel' =====

Updating anchor peers for USERS...
+ peer channel update -o orderer.property-registration-network.com:7050 -c registrationchannel -f ./channel-artifacts/usersMSPanchors.tx
+ res=0
+ set +x
2020-08-30 19:42:53.014 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2020-08-30 19:42:53.035 UTC [channelCmd] update -> INFO 002 Successfully submitted channel update
===== Anchor peers updated for org 'usersMSP' on channel 'registrationchannel' =====

===== All GOOD, Hyperledger Fabric Property Registration Network Is Now Up and Running! =====

END

```

2. Open another terminal and **connect to CHAINCODE container**:

Then first make sure all node dependency are in place and then start the (start-dev)script.

Command :

- docker exec -it chaincode /bin/bash
- npm install
- npm run start-dev

Output:

```

root@0514e399c9ed:/opt/gopath/src/chaincode# ls -ll
total 44
-rwxrwxrwx  1 root root  5072 Aug 30 10:28 cert.crt
-rw-rw-rw-  1 1000 1000   191 Aug 29 09:55 index.js
drwxrwxrwx 169 root root  4096 Aug 29 21:46 node_modules
-rw-rw-rw-  1 1000 1000  1174 Aug 30 19:26 package.json
-rw-rw-rw-  1 1000 1000  6360 Aug 30 19:36 registrarContract.js
-rw-rw-rw-  1 1000 1000 14771 Aug 30 09:50 userContract.js
root@0514e399c9ed:/opt/gopath/src/chaincode#

```

```

root@0514e399c9ed:/opt/gopath/src/chaincode# npm run start-dev
> regnet@0.0.1 start-dev /opt/gopath/src/chaincode
> fabric-chaincode-node start --peer.address peer0.registrar.property-registration-network.com:7053 --chaincode-id-name regnet:1.1

Configuring yargs through package.json is deprecated and will be removed in the next major release, please use the JS API instead.
Configuring yargs through package.json is deprecated and will be removed in the next major release, please use the JS API instead.
Configuring yargs through package.json is deprecated and will be removed in the next major release, please use the JS API instead.
2020-08-30T19:54:10.074Z info [shim:contracts-spi/bootstrap.js] No metadata file supplied in contract, introspection will generate all the data
2020-08-30T19:54:10.076Z info [shim:contracts-spi/chaincodefromcontract.js] Using serializers ["transaction","jsonSerializer","serializers":{}]
Configuring yargs through package.json is deprecated and will be removed in the next major release, please use the JS API instead.
2020-08-30T19:54:10.171Z info [shim:lib/chaincode.js] Registering with peer peer0.registrar.property-registration-network.com:7053 as chaincode "regnet:1.1"

Command succeeded

2020-08-30T19:54:10.219Z info [shim:lib/handler.js] Successfully registered with peer node. State transferred to "established"
2020-08-30T19:54:10.225Z info [shim:lib/handler.js] Successfully established communication with peer node. State transferred to "ready"
Regnet Smart Contract Instantiated
2020-08-30T19:57:10.346Z info [shim:lib/handler.js] [registrationchannel-5a3e53cc] Calling chaincode Init() succeeded. Sending COMPLETED message back to peer

```

3. Install and Instantiate Chaincode:

Command :

➤ ./fabricNetwork.sh install

Output :

```
root@upgrad-VirtualBox:/home/upgrad/desktop/property-registration/network# ./fabricNetwork.sh install
Installing chaincode for channel 'registrationchannel' with CLI timeout of '15' seconds and CLI delay of '5' seconds and chaincode version '1.1'
Continue? [Y/n]
proceeding ...
LOCAL_VERSION=1.4.2
DOCKER_IMAGE_VERSION=1.4.3
===== REGNET =====
Local fabric binaries and docker images are
out of sync. This may cause problems.
=====

START

Deploying Chaincode REGNET On Certification Network

Channel name : registrationchannel
# peer chaincode install -n regnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
Installing chaincode on peer0.registrar.property-registration-network.com ...
# res=0
# set xx
2020-08-30 19:57:09.459 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 001 Using default escc
2020-08-30 19:57:09.459 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 002 Using default vsc
2020-08-30 19:57:09.638 UTC [chaincodeCmd] install -> INFO 003 Installed remotely response=status:200 payload:"OK" >
===== Chaincode is installed on peer0.registrar =====
Installing chaincode on peer1.registrar.property-registration-network.com ...
# peer chaincode install -n regnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
# res=0
# set xx
2020-08-30 19:57:09.761 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 001 Using default escc
2020-08-30 19:57:09.761 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 002 Using default vsc
2020-08-30 19:57:09.923 UTC [chaincodeCmd] install -> INFO 003 Installed remotely response=status:200 payload:"OK" >
===== Chaincode is installed on peer1.registrar =====
Installing chaincode on peer0.users.property-registration-network.com ...
# peer chaincode install -n regnet -v 1.1 -l node -p /opt/gopath/src/github.com/hyperledger/fabric/peer/chaincode/
# res=0
# set xx
2020-08-30 19:57:10.024 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 001 Using default escc
2020-08-30 19:57:10.024 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 002 Using default vsc
2020-08-30 19:57:10.198 UTC [chaincodeCmd] install -> INFO 003 Installed remotely response=status:200 payload:"OK" >
===== Chaincode is installed on peer0.users =====
Instantiating chaincode on channel using peer0.registrar.property-registration-network.com ...
# peer chaincode instantiate -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -l node -v 1.1 -c '{"Args":["org.property-registration-network.user.instantiate"]}' -P 'OR ("registrarRSP.member","usersRSP.member")'
# res=0
# set xx
2020-08-30 19:57:10.312 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 001 Using default escc
2020-08-30 19:57:10.312 UTC [chaincodeCmd] checkChaincodeCmdParams -> INFO 002 Using default vsc
===== Chaincode is instantiated on peer0.registrar on channel 'registrationchannel' =====

===== ALL GOOD, Chaincode REGNET is Now Installed & Instantiated On Registration Network =====

END
```

4. Now we could connect to the CLI container and start invoking our functions: **Invoking instantiate function**

Command:

- `docker exec -it cli /bin/bash`
- `peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user.instantiate"]}'`
- `peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar.instantiate"]}'`

Output:

From CLI container:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user.instantiate"]}'
2020-08-30 20:00:59.641 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

From Chaincode :

```
2020-08-30T19:54:10.219Z info [shim:lib/handler.js] Successfully registered with peer node. State transferred to "established"
2020-08-30T19:54:10.225Z info [shim:lib/handler.js] Successfully established communication with peer node. State transferred to "ready"
Regnet Smart Contract Instantiated
2020-08-30T19:57:10.346Z info [shim:lib/handler.js] [registrationchannel-5a3e53cc] Calling chaincode Int() succeeded. Sending COMPLETED message back to peer
Regnet Smart Contract Instantiated
2020-08-30T20:00:59.637Z info [shim:lib/handler.js] [registrationchannel-a3ae8d4a] Calling chaincode Invoke() succeeded. Sending COMPLETED message back to peer
```

```

2020-08-30T20:08:59.637Z info [shim:lib/handler.js] [registrationchannel-a3ae8d4a] Calling chaincode Invoke() succeeded. Sending COMPLETED message back to peer
Registrar Smart Contract Instantiated
2020-08-30T20:13:59.077Z info [shim:lib/handler.js] [registrationchannel-13101190] Calling chaincode Invoke() succeeded. Sending COMPLETED message back to peer

```

From peer0.registrar :

```

2020-08-30T20:08:59.626530674Z 2020-08-30 20:08:59.626 UTC [endorser] callChaincode -> INFO 074 [registrationchannel][a3ae8d4a] Entry chaincode: name="regnet"
2020-08-30T20:08:59.639436018Z 2020-08-30 20:08:59.639 UTC [endorser] callChaincode -> INFO 075 [registrationchannel][a3ae8d4a] Exit chaincode: name="regnet" (13ms)
2020-08-30T20:10:59.639275532Z 2020-08-30 20:08:59.639 UTC [comm.grpc.server] 1 -> INFO 076 unary call completed grpc.service=protos.Endorser grpc.method=ProcessProposal grpc_peer_address=172.21.0.11:5307
8 grpc.code=OK grpc.call_duration=14.426347ms
2020-08-30T20:09:01.654048160Z 2020-08-30 20:09:01.653 UTC [gossip.privdata] StoreBlock -> INFO 077 [registrationchannel] Received block [4] from buffer
2020-08-30T20:09:01.654652869Z 2020-08-30 20:09:01.654 UTC [committer.txvalidator] Validate -> INFO 078 [registrationchannel] Validated block [4] in 0ms
2020-08-30T20:09:01.671344012Z 2020-08-30 20:09:01.671 UTC [endorser] commitWithHashes -> INFO 079 [registrationchannel] Committed block [4] with 1 transaction(s) in 16ms (state_validation=0ms block_and_
privdata_commit=11ms state_commit=3ms) commitHash=[dffd43e31827767df9829c28f86261833dad1dbb6efee279a87cd97de464f127]

```

Important Note:

- If a registrar member tries to invoke a Functions which can only be **initiated by peer0.users** , we will return a String “**Users should be used to raise a user or property registration request**”
- And similarly, **only REGISTRAR peer’s can access all functions in org.property-registration-network.registrar smart contract**
- If any peer who is not member registrar return string “**Registrar member can only approve user or property request**”
- For all the functions related to users => use the below steps:
 1. Connect to chaincode container: **docker exec -it chaincode /bin/bash**
 2. command : **npm run start-dev** (as this script is pointing to a peer0.users)
 3. Then open another terminal and connect to peer0.users.property-registration-network.com: **docker exec -it peer0.users.property-registration-network.com /bin/bash**
 4. execute all the function where initiator is USER from the above terminal.
- For all the functions related to registrar => use the below steps:
 1. Connect to chaincode container: **docker exec -it chaincode /bin/bash**
 2. command : **npm run start-dev** (as this script is pointing to a peer0.registrar)
 3. Then open another terminal and connect to peer0.registrar.property-registration-network.com: **docker exec -it peer0.registrar.property-registration-network.com /bin/bash**
 4. execute all the function where initiator is REGISTRAR from the above terminal.



```
{
  "name": "regnet",
  "version": "0.0.1",
  "description": "Registration Network",
  "main": "index.js",
  "engines": {
    "node": ">=8",
    "npm": ">=5"
  },
  "scripts": {
    "lint": "eslint .",
    "pretest": "npm run lint",
    "test": "nyc mocha test --recursive",
    "start-dev": "fabric-chaincode-node start --peer.address peer0.registrar.property-registration-network.com:7053 --chaincode-id-name regnet:1.1",
    "start-dev1": "fabric-chaincode-node start --peer.address peer0.users.property-registration-network.com:9053 --chaincode-id-name regnet:1.1",
    "start": "fabric-chaincode-node start",
    "mocha": "mocha test --recursive"
  },
  "engineStrict": true,
  "author": "Avi K",
  "dependencies": {
    "fabric-contract-api": "~1.4.2",
    "fabric-shim": "~1.4.2"
  }
}
```

EXAMPLE :

1. To invoke a function from **user contract using peer0.users**:

STEP 1 : connect to chaincode container :

docker exec -it chaincode /bin/bash

STEP 2 : Run the script

npm run start-dev1 (as this script is pointing to peer0.users)

STEP 3: On another terminal connect to peer0.users.property-registration-network.com:

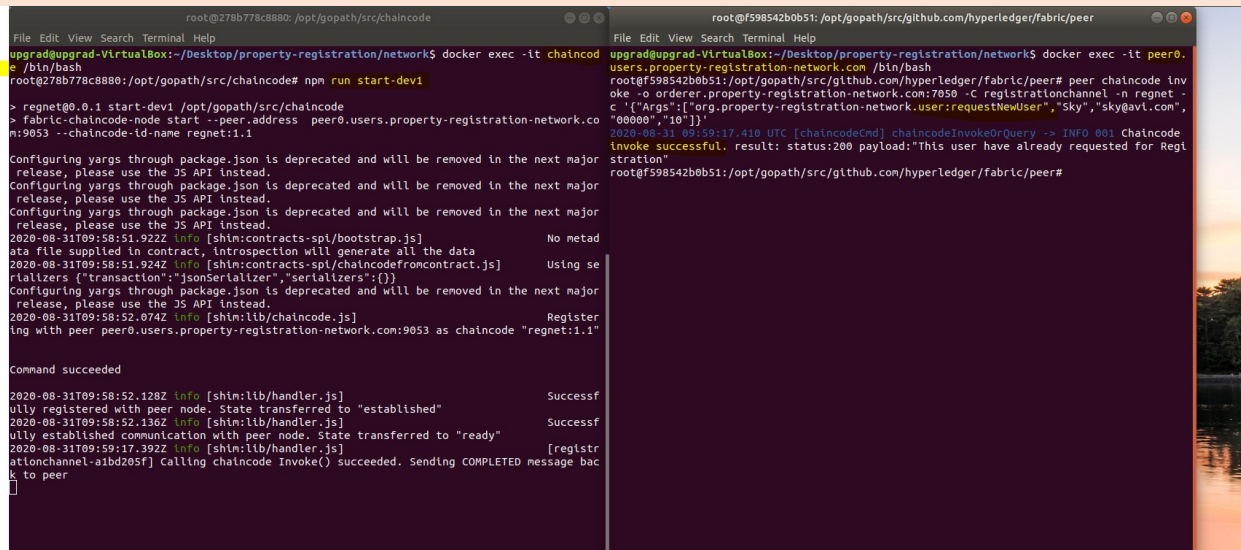
docker exec -it peer0.users.property-registration-network.com /bin/bash

STEP 4 : invoke any function from the user contract:

Example:

peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:requestNewUser","Sky","sky@avi.com","00000","10"]}'

OUTPUT:

The image shows two terminal windows side-by-side. The left window is titled 'root@278b778c8880: /opt/gopath/src/chaincode' and shows the execution of 'npm run start-dev' for a chaincode named 'regnet'. It displays messages about configuring yargs, introspection, and successful registration with a peer. The right window is titled 'root@f598542b0b51: /opt/gopath/src/github.com/hyperledger/fabric/peer' and shows the execution of 'peer chaincode invoke' to call the 'requestNewUser' function on the 'regnet' contract. It shows the successful invocation result.

EXAMPLE :

1. To invoke a function from **registrar** contract using **peer0.registrar**:

STEP 1 : connect to chaincode container :

docker exec -it chaincode /bin/bash

Stop the npm run start-dev script (ctrl +c) then,

STEP 2 : Run the script

npm run start-dev (as this script is pointing to peer0.registrar)

STEP 3: On another terminal connect to peer0.registrar.property-registration-network.com:

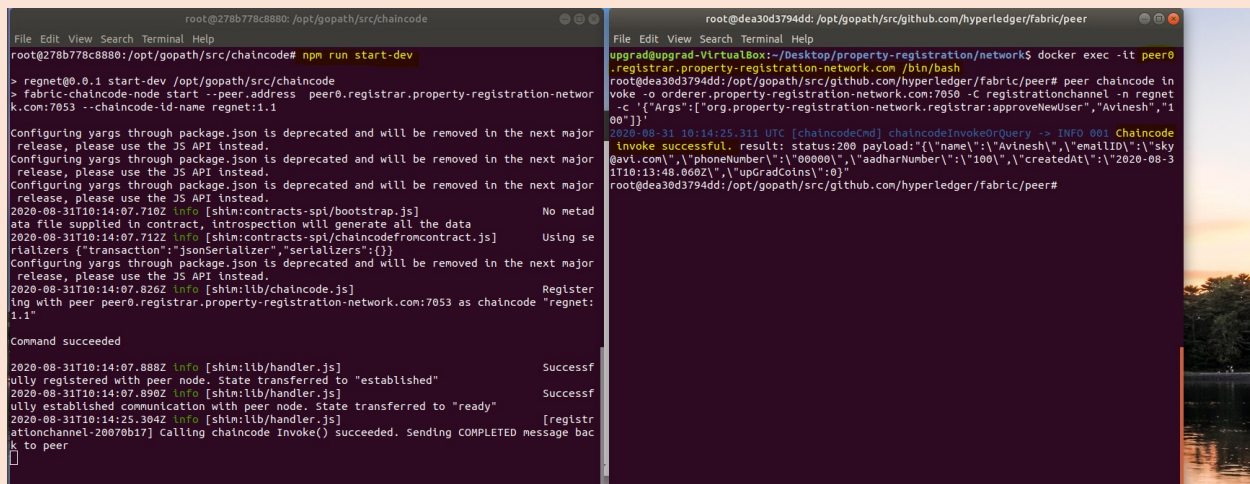
docker exec -it peer0.registrar.property-registration-network.com /bin/bash

STEP 4 : invoke any function from the user contract:

Example:

peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:approveNewUser","Sky","10"]}'

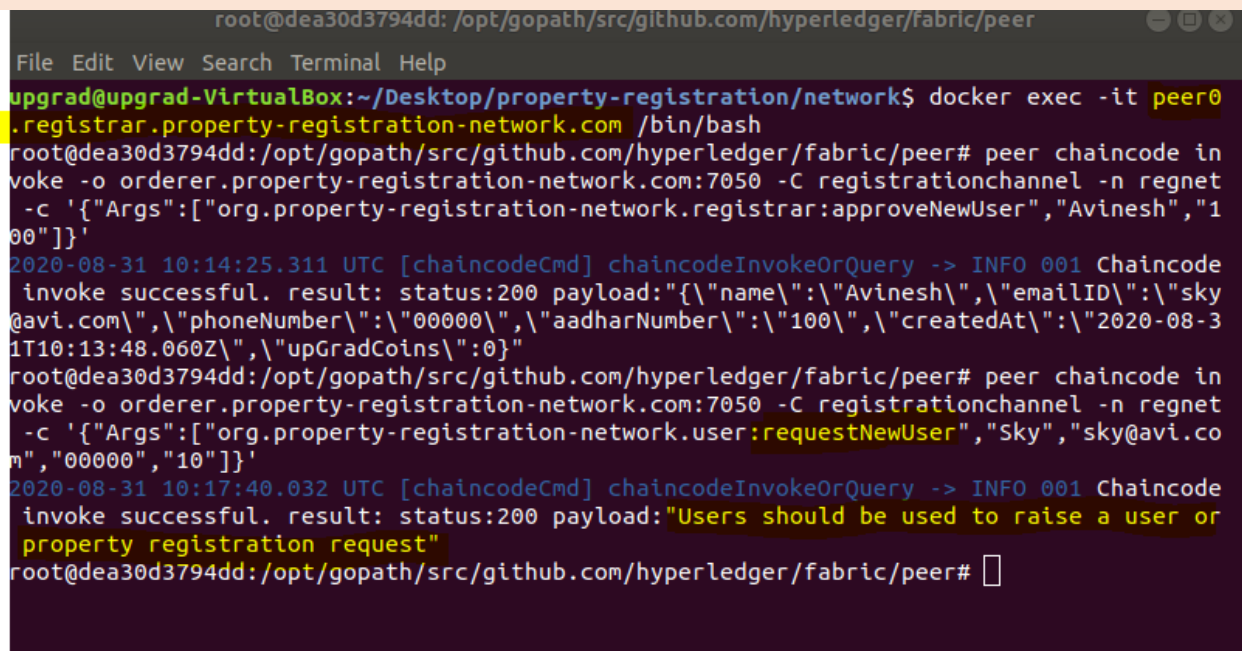
OUTPUT:

The image shows two terminal windows side-by-side. The left window is titled 'root@278b778c8880: /opt/gopath/src/chaincode' and shows the command 'npm run start-dev' being executed. It displays logs for the 'regnet@0.0.1 start-dev' command, including configuration details and a successful chaincode invocation. The right window is titled 'root@dea30d3794dd: /opt/gopath/src/github.com/hyperledger/fabric/peer' and shows the command 'docker exec -it peer0 .registrar.property-registration-network.com /bin/bash'. It then shows the 'peer chaincode invoke' command being executed, with a successful result indicating the chaincode was invoked successfully.

If we try to invoke a function which is not supposed to be initiated by a member peer error :

Output:

Example: User or property request can only be raised by peer0.users

The image shows a terminal window titled 'root@dea30d3794dd: /opt/gopath/src/github.com/hyperledger/fabric/peer'. It shows the command 'docker exec -it peer0 .registrar.property-registration-network.com /bin/bash' being executed. Then, the 'peer chaincode invoke' command is run with arguments for the 'approveNewUser' function. The output shows a successful status but then an error message: 'Users should be used to raise a user or property registration request'. This indicates that the invocation failed because the user is not a registrar.

Example: Approval of user or property registration can only be done by Registrar.

```
root@f598542b0b51: /opt/gopath/src/github.com/hyperledger/fabric/peer
File Edit View Search Terminal Help
root@dea30d3794dd: /opt/gopath/src/github.com/hyperledger/fabric/peer# exit
exit
upgrad@upgrad-VirtualBox: ~/Desktop/property-registration-network$ docker exec -it peer0
.users.property-registration-network.com /bin/bash
root@f598542b0b51: /opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode in
voke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet
-c '{"Args":["org.property-registration-network.registrar:approveNewUser","Sky","10"]}'
,
2020-08-31 10:19:06.042 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode
invoke successful. result: status:200 payload:"Registrar's can only approve user or pr
operty request"
root@f598542b0b51: /opt/gopath/src/github.com/hyperledger/fabric/peer#
```

Commands for invoking all functions in user and registrar contract

5. Request User:

Command:

- `peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:requestNewUser","Avi","avi@avi.com","123456","1"]}'`

Output:

```
root@264ffb335c99: /opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registra
tionchannel -n regnet -c '{"Args":["org.property-registration-network.user:requestNewUser","Avi","avi@avi.com","123456","1"]}'
2020-08-30 20:16:57.535 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"name":"Avi",
"emailID":"avi@avi.com","phoneNumber":"123456","aadharNumber":"1","createdAt":"2020-08-30T20:16:57.522Z"}
root@264ffb335c99: /opt/gopath/src/github.com/hyperledger/fabric/peer#
```

6. Approve User Request:

Command:

- `peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:approveNewUser","Avi","1"]}'`

Output:

```
root@264ffb335c99: /opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registra
tionchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:approveNewUser","Avi","1"]}'
2020-08-30 20:19:27.287 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"name":"Avi",
"emailID":"avi@avi.com","phoneNumber":"123456","aadharNumber":"1","createdAt":"2020-08-30T20:16:57.522Z","upGradCoins":0}
root@264ffb335c99: /opt/gopath/src/github.com/hyperledger/fabric/peer#
```

7. Recharge account:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:rechargeAccount","Avi","1","upg1000"]}'**

Output:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:rechargeAccount","Avi","1","upg1000"]}'
2020-08-30 20:20:50.109 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"name":"Avi",
"emailID":"avi@avi.com","phoneNumber":"123456","aadharNumber":"1","createdAt":"2020-08-30T20:16:57.522Z","upGradCoins":1000}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

8. View Approved user: From Registrar and user smartcontract:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:viewUser","Avi","1"]}'**
- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:viewUser","Avi","1"]}'**

Output:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:viewUser","Avi","1"]}'
2020-08-30 20:22:51.836 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"name":"Avi",
"emailID":"avi@avi.com","phoneNumber":"123456","aadharNumber":"1","createdAt":"2020-08-30T20:16:57.522Z","upGradCoins":1000}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:viewUser","Avi","1"]}'
2020-08-30 20:23:01.094 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"name":"Avi",
"emailID":"avi@avi.com","phoneNumber":"123456","aadharNumber":"1","createdAt":"2020-08-30T20:16:57.522Z","upGradCoins":1000}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

9. Property registration request:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:propertyRegistrationRequest","999","Avi","1000","registered","Avi","1"]}'**

Output:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:propertyRegistrationRequest","999","Avi","1000","registered","Avi","1"]}'
2020-08-30 20:27:02.424 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"propertyId":"999","owner":"","u0000org.property-registration-network.regnet.approvedUser\\u0000Avi-1\\u0000","price":"1000","status":"registered"}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

10. Property approve:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:approvePropertyRegistration","999"]}'**

Output:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:approvePropertyRegistration","999"]}'
2020-08-30 20:28:12.228 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"propertyId":"999","owner":"","price":"1000","status":"registered"}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

11. View property: as user or registrar

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:viewProperty","999"]}'**
- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:viewProperty","999"]}'**

Output:

```
root@92df64f5668c:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:viewProperty","999"]}'
2020-08-30 20:49:09.392 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"propertyId":"999","owner":"","price":"1000","status":"registered"}
root@92df64f5668c:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.registrar:viewProperty","999"]}'
2020-08-30 20:49:10.684 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"propertyId":"999","owner":"","price":"1000","status":"registered"}
root@92df64f5668c:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

12. Update property:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:updateProperty","999","Avi","1","onSale"]}'**

Output:

```
root@92df64f5668c:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:updateProperty","999","Avi","1","onSale"]}'
2020-08-30 20:51:46.855 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"propertyId":"999","owner":"","price":"1000","status":"onSale"}
root@92df64f5668c:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

13. Purchase property:

Command:

- **peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:purchaseProperty","999","Sky","10"]}'**

Output:

```
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c '{"Args":["org.property-registration-network.user:purchaseProperty","999","Sky","10"]}'
2020-08-30 20:59:34.819 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:{"OldOwner": "Avi", "NewOwner": "Sky", "price": "1000", "status": "registered", "propertyId": "999"}
root@264ffb335c99:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

