Solution for Hyperledger Fabric Project

This is a step-by-step solution on how to download and install the hyperledger project for upgrad.com

1. Downloading the project files

The project file is available to download from this link. We can go to **this link** and download the *course2assignment_Rohit_Kundliwall.zip* file that contains the property-registration folder and solution.pdf file.

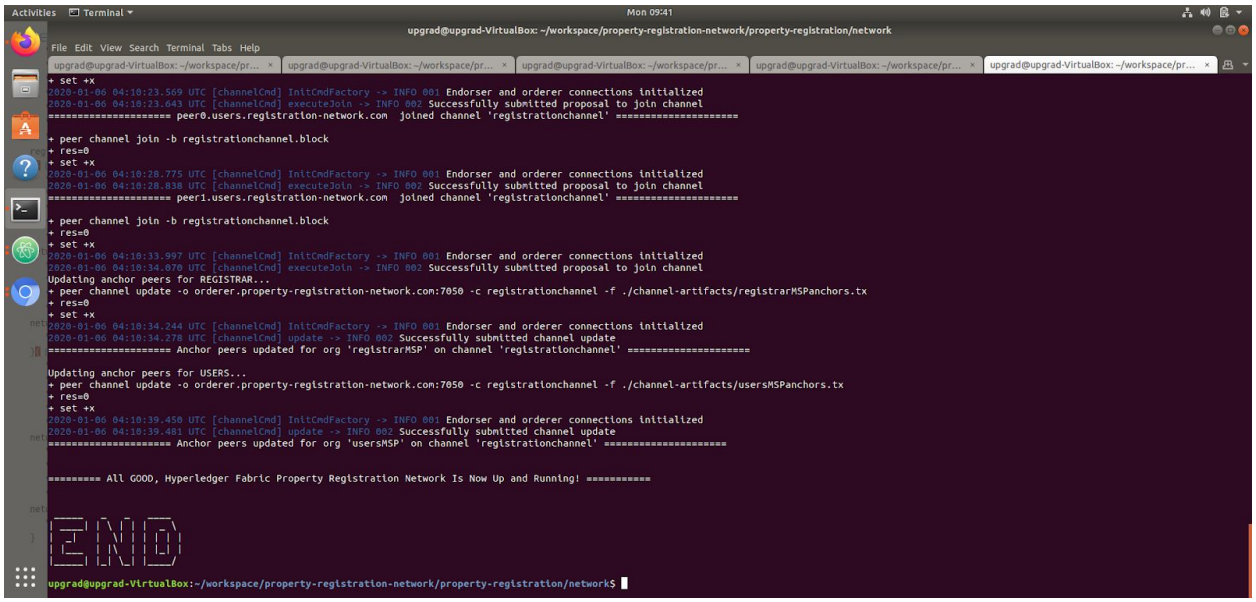
2. Starting the network and instantiating the chaincodes

To start the network, make sure you have all the prerequisites installed on the computer. **This is the URL** to a document that I have created to help anyone install Hyperledger Fabric prerequisites on your computer.

After you have installed all the prerequisites, you need to create a folder in the home directory with the name 'workspace'. Unzip the files of course2assignment_Rohit_Kundliwal.zip into this folder.

Now you need to open the terminal and cd into the network folder of the property-registration folder inside the workspace directory:
`cd ~/workspace/property-registration/network`

Once in the network directory, you can now start, stop or restart the fabric network.
To start `./fabricNetwork.sh up`
To stop `./fabricNetwork.sh down`
To restart `./fabricNetwork.sh restart`
Let us start the network by `./fabricNetwork.sh up` and you should see the below screen:

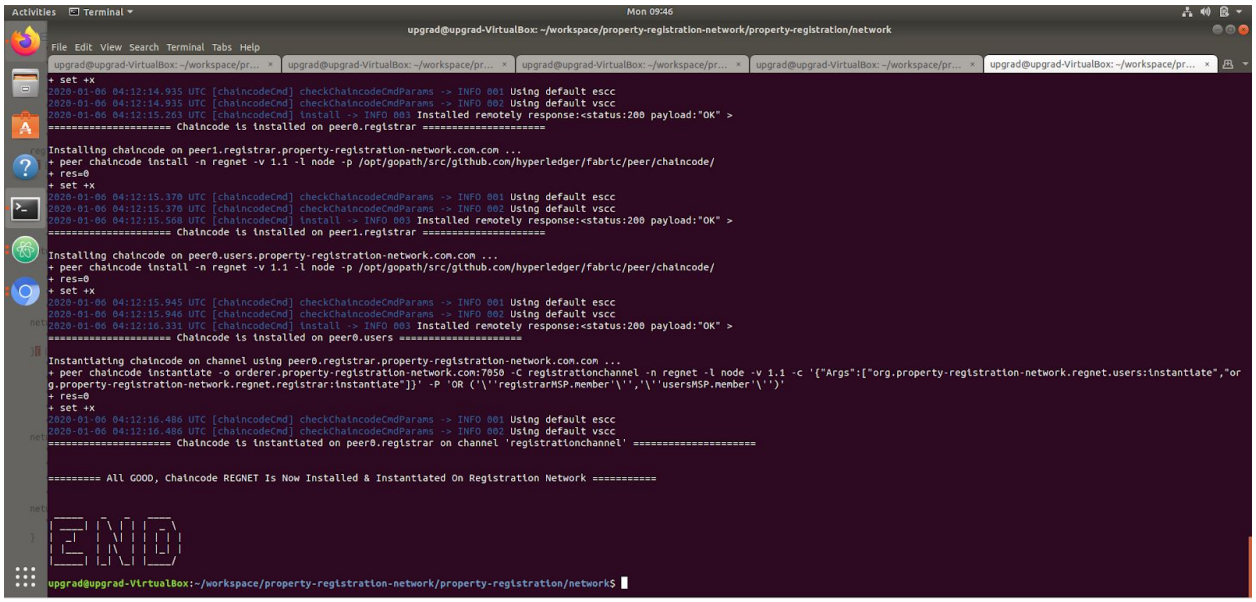


The above screenshot shows that the hyperledger fabric property registration network is up and running.

Troubleshooting: If you face problems in starting the network, there may be permission issues with the files generated by cryptogen or configtxgen tool. You can solve this problem by going to **this URL to a document** which I have created to help troubleshoot network launch problems by providing a step-by-step guide.

Once your network is up and running, you can install the chaincodes present in the chaincode folder of the property-registration directory.

To install and instantiate the chaincodes, run `./fabricNetwork.sh install` in the terminal. On successful installation and instantiation, you should see the below screen:



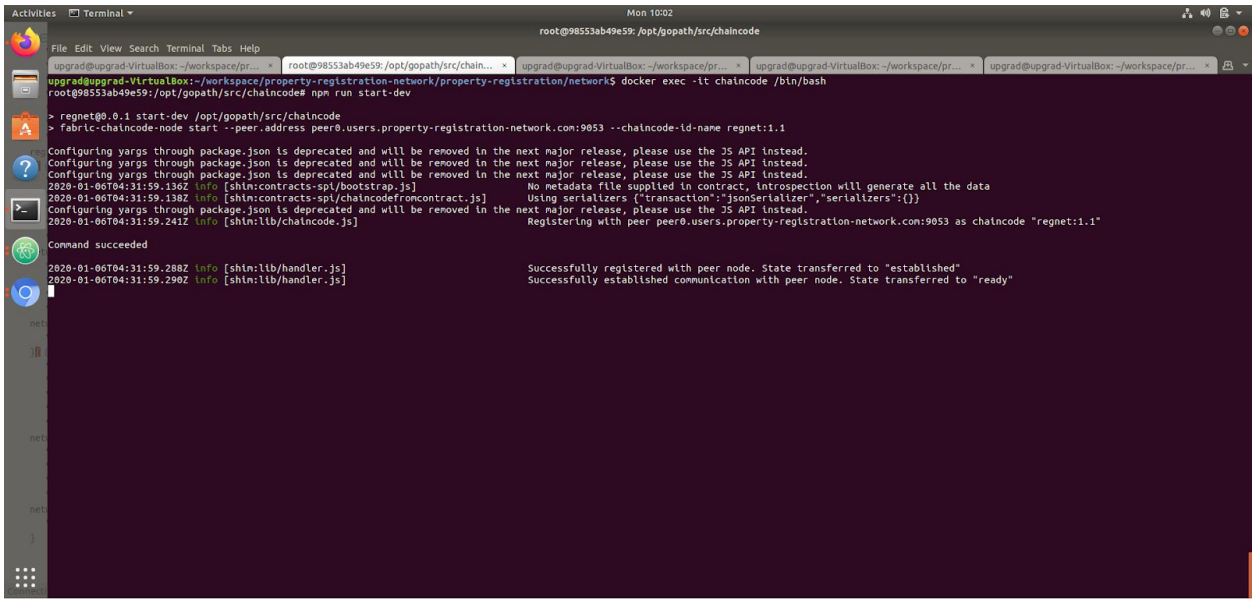
The above screenshot shows the successful installation and instantiation of chaincode REGNET on the Registration channel.
If you are unable to start the chaincode container, you can go to **this URL to a document** that I have created that will help you in starting the chaincode containers.
If you face problems in installation and instantiation of REGNET chaincode, please inspect the function installChaincode inside utils.sh file inside the scripts folder in the network directory.

On successful starting of hyperledger network and installation and instantiation of the REGNET chaincodes, we are now ready to see the functionalities of this network.

Invoking the functions of REGNET chaincode

First, we need to start the chaincode node application. We will SSH into chaincode container and install npm

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



Above screenshot is what you should see after entering the 3 commands.

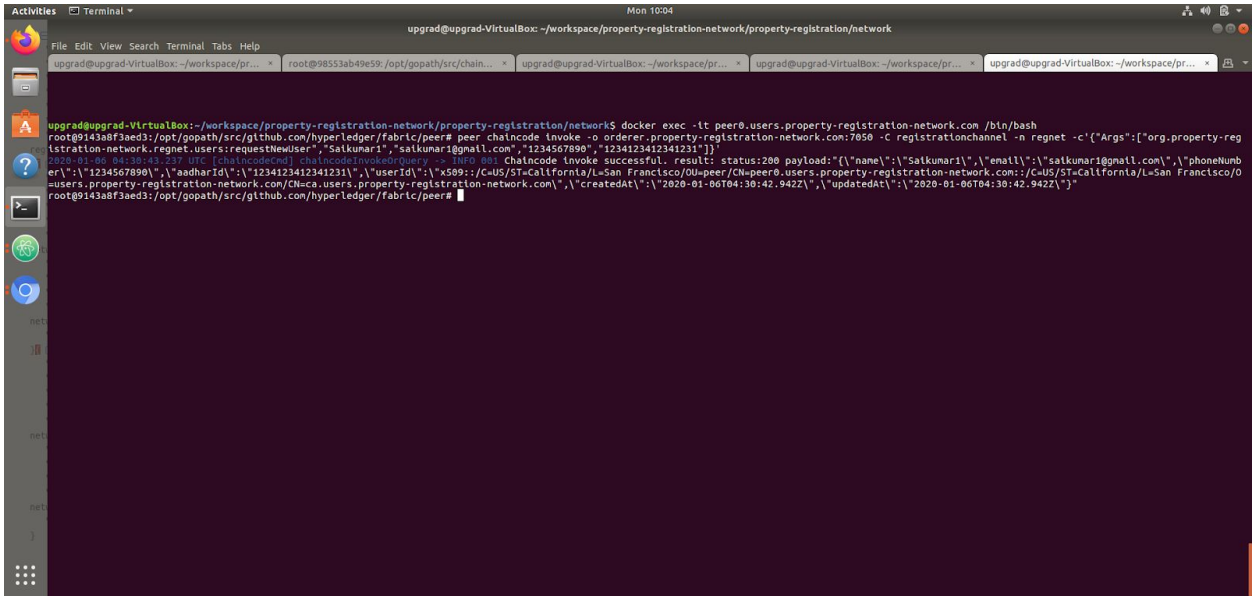
Now we will open another terminal and SSH into peer0 of user organisation

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

We will call this terminal '*user terminal*' from now.
We can now execute the chaincode commands.

Function 1: requestNewUser
Go to user terminal and let's run the requestNewUser command for Saikumar1, a new user

```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet
-c'{"Args":["org.property-registration-network.regnet.users:requestNewUser","Saikumar1","Saikumar1@gmail.c
om","1234567890","1234123412341231"]}'
```



Function 2: approveNewUser

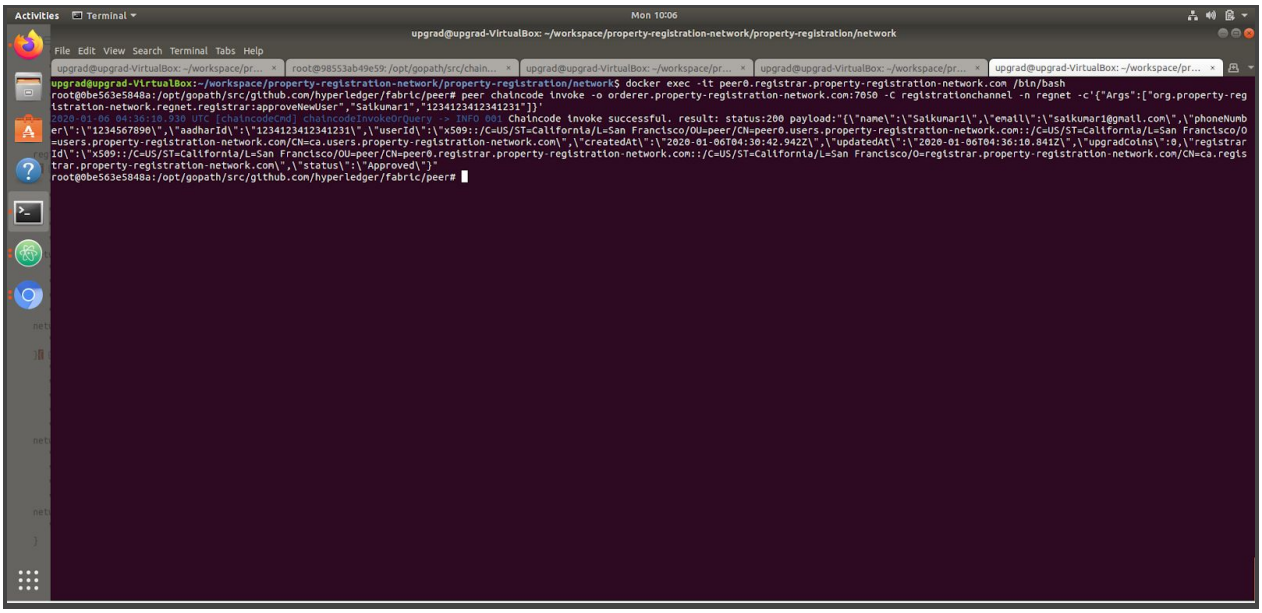
Open new terminal and enter peer0 of registrar organisation

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

We will call this terminal the '*registrar terminal*' from now.

Now execute the command to approve new user

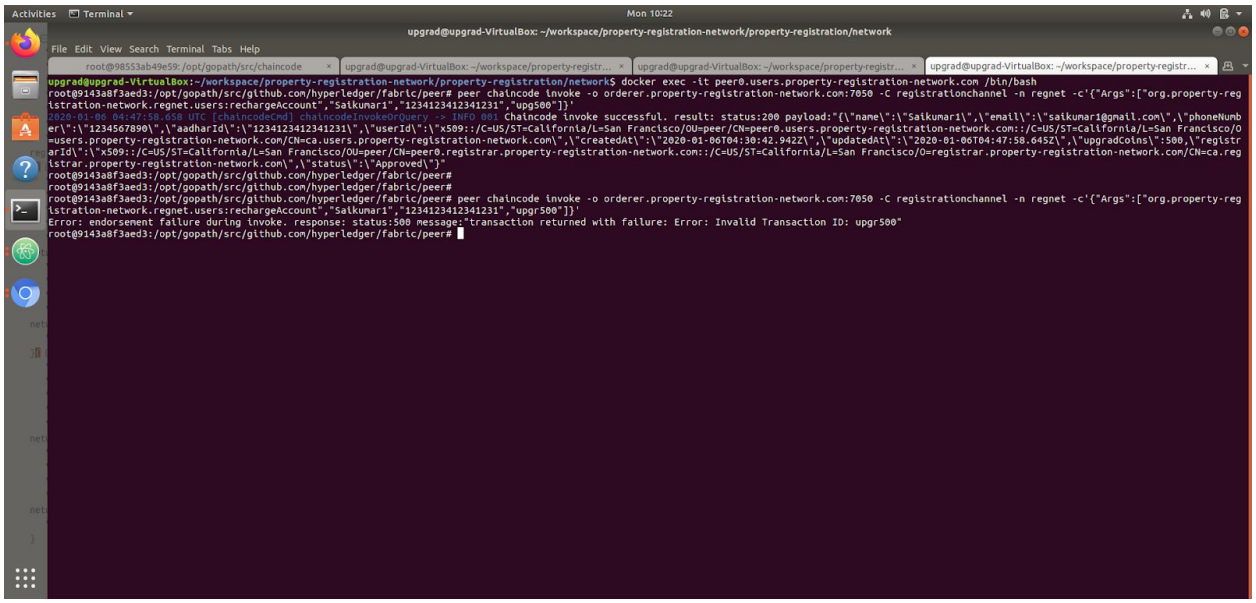
```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet
-c'{"Args":["org.property-registration-network.regnet.registrar:approveNewUser","Saikumar1","1234567890"]}
'
```



Function 3: rechargeAccount

Go to the user terminal and execute the recharge account function

```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:rechargeAccount","Saikumar1","123456789","upg500"]}'
```

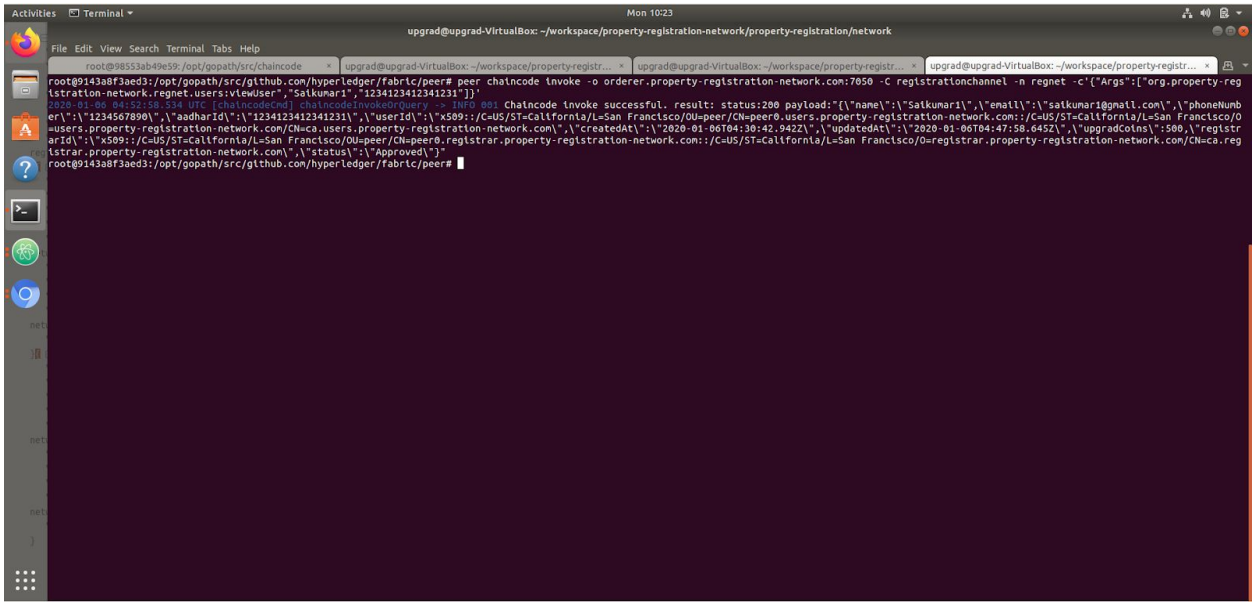


Above screenshot shows the success and failure functionality of rechargeAccount function

Function 4: viewUser

You can access this function from both user and registrar terminal. Go to user terminal and execute the view user function for Saikumar1 user

```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:viewUser","Saikumar1","123456789"]}'
```

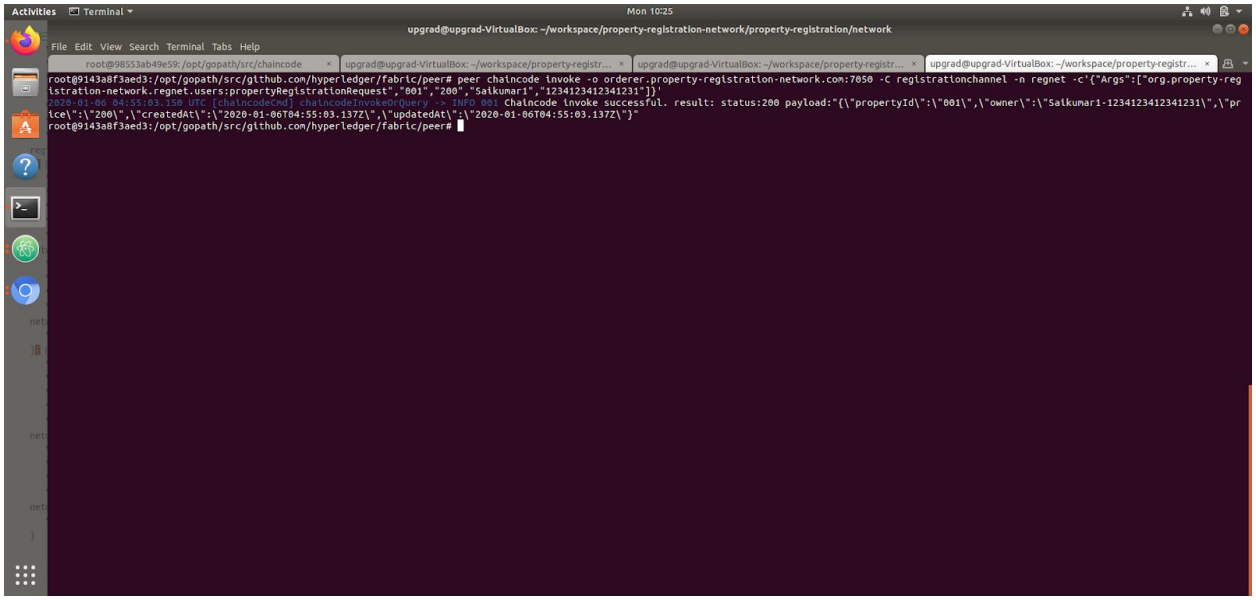


Note that ideally in the real world, the user should not be able to see the sensitive information of other users.

Function 5: propertyRegistrationRequest

Here the user Saikumar1 is requesting a property registration of ID 001 on the network.

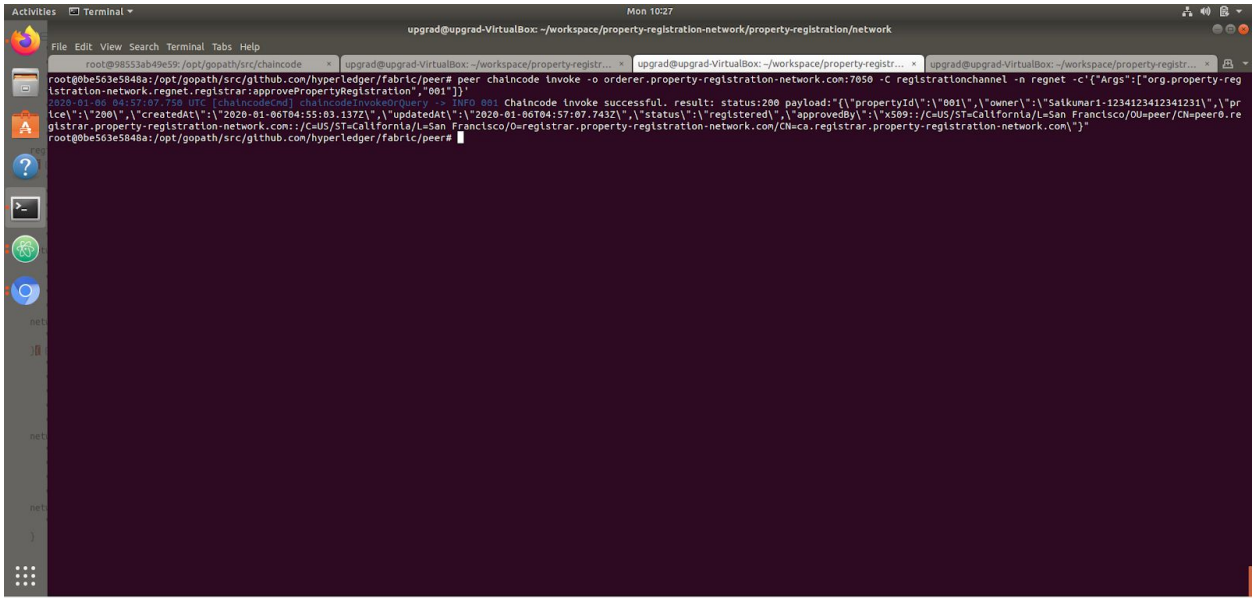
```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:propertyRegistrationRequest","001","200","Saikumar1","987654321"]}'
```



Function 6: approvePropertyRegistration

Go to the registrar terminal and execute the approve property registration function.

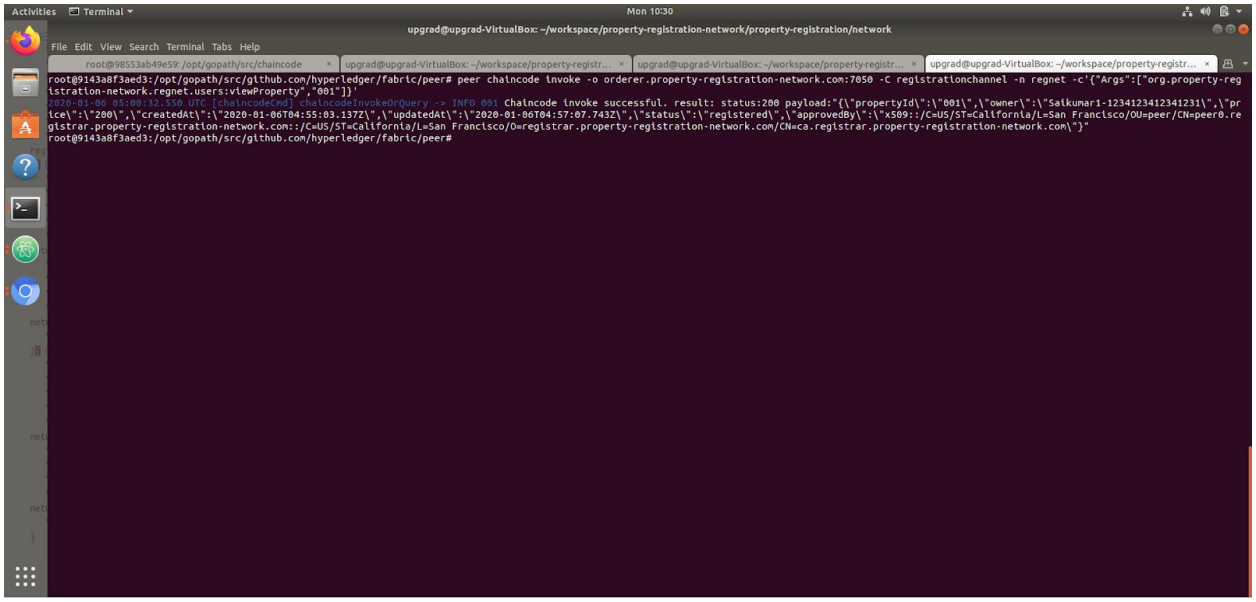
```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.registrar:approvePropertyRegistration","001"]}'
```



Function 7: viewProperty

You can access this function from both user and registrar terminal. Execute this command from either registrar or user terminal. We will use the user terminal and access the property ID 001

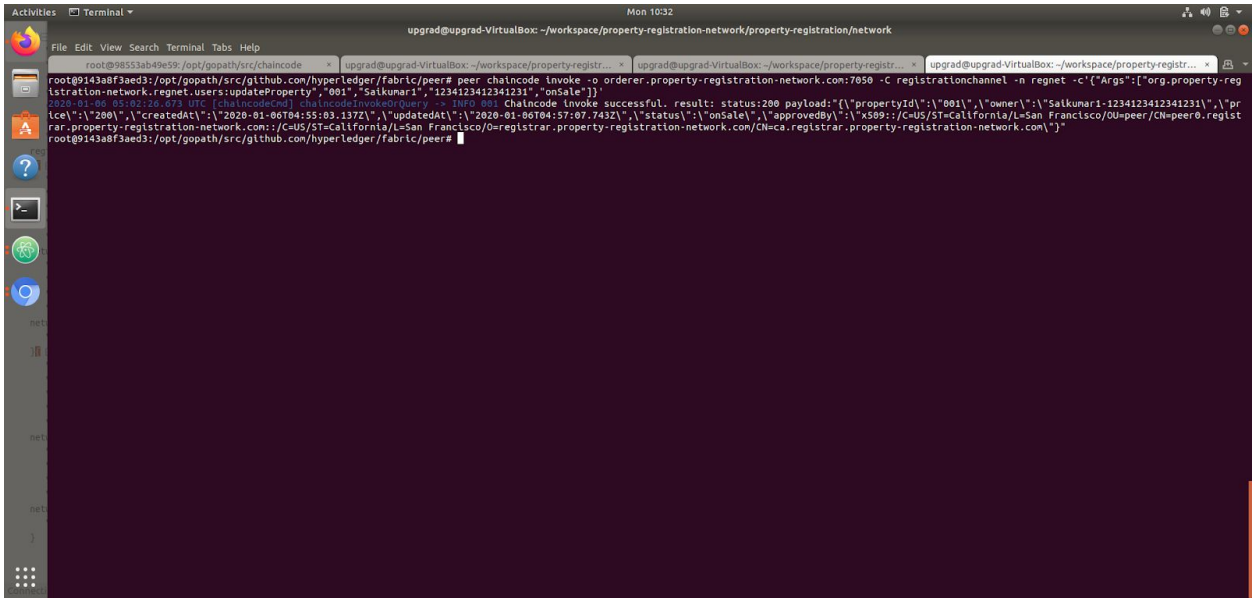
```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:viewProperty","001"]}'
```



Function 8: updateProperty

Go to the user terminal and execute update property function by entering username Saikumar1 and aadhar number 987654321 and updating the status of the property as “onSale”

```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:updateProperty","001","Saikumar1","123123123123231","onSale"]}’
```

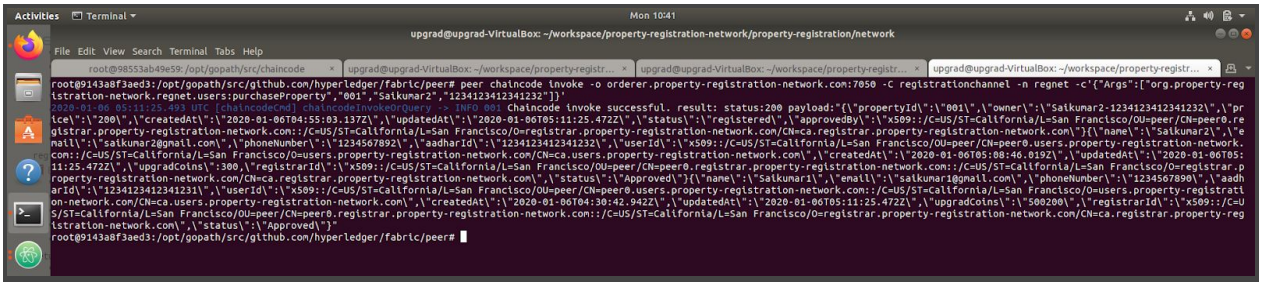


Function 9: purchaseProperty

For this, first we will have to make another user called Saikumar2 who will try to purchase the property of Saikumar1. Hence follow the function 1 and 2 to request and approve a new user with name Saikumar2.

Now, go to the user terminal and we will make user Saikumar1 purchase the property ID 001. This will reduce the 500 upgradcoins from Saikumar1 user and add the same in Saikumar2 user.

```
peer chaincode invoke -o orderer.property-registration-network.com:7050 -C registrationchannel -n regnet -c'{"Args":["org.property-registration-network.regnet.users:purchaseProperty","001","Saikumar2","1234123412341232"]}'
```



You have now successfully been able to test all the 9 functionalities of chaincode installed on the property registration network. Congrats!