**Instructions to be followed to test smart contract on Remix IDE.**

Step 1:Open [www.remix.ethereum.org](http://www.remix.ethereum.org)

Step 2:In FileExplorer create a new file by name KYC.sol

Step 3:Copy the code from KYC.sol file present in Phase-2 folder to KYC.sol file in remix IDE.

Step 4:Navigate to solidity compiler ,select compiler version,language as solidity and compile KTC.sol smart contract.

Step 5:Navigate to Deploy and Run transaction, select environment as JavascriptVM and select any one account ,which plays the role of admin and deploy the smart contract.

Step 6:The address which deploys the smart contract becomes the admin.

Step 7:In this address as an admin one can add bank,remove bank and modify bank kyc permission.

Step 8:For example at this address“0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c” the contract is deployed then it becomes admin address.

Step 9:without changing account execute addbank function by providing bank name, bank address and registration number.

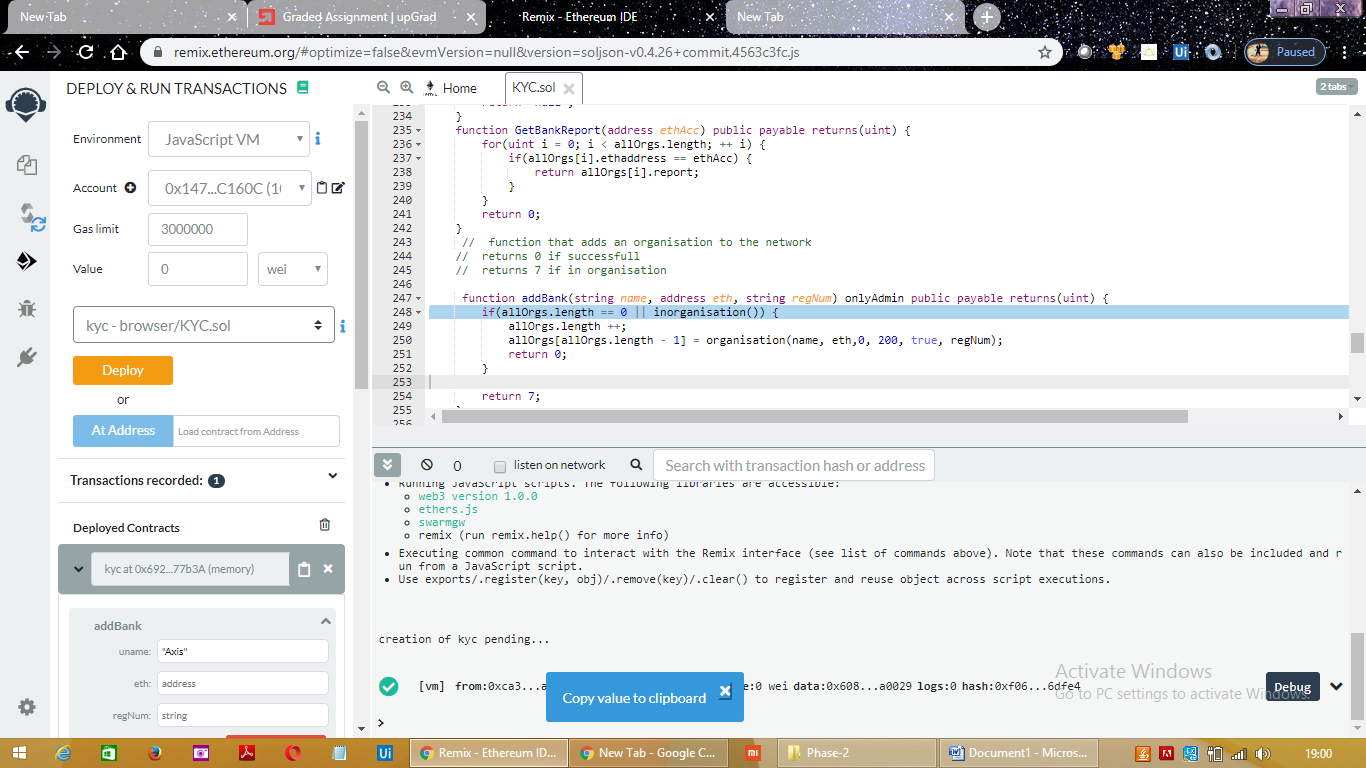
Admin address: 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c

Name: “Axis”

Eth: “0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C”

Regnum:”axc123bbvh”

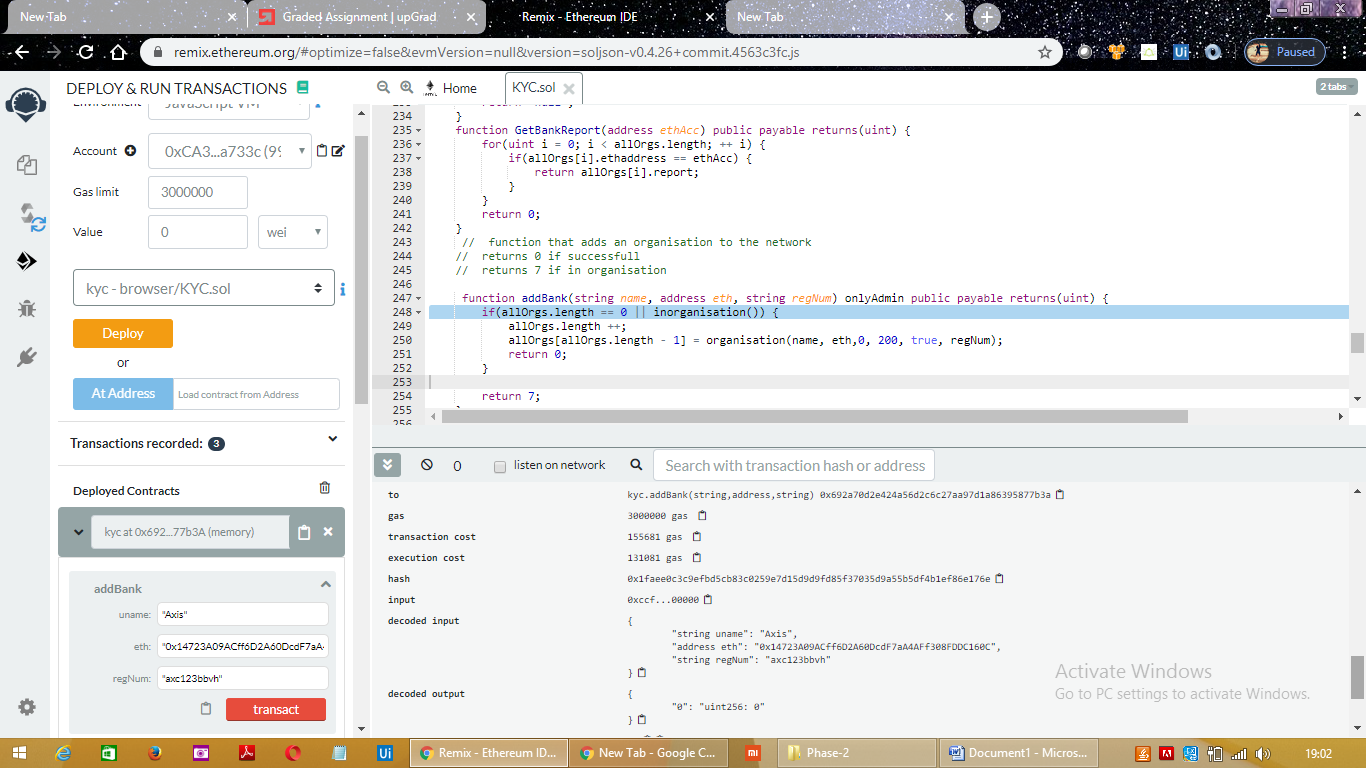
Copy the account address from account which we want to add as bank .



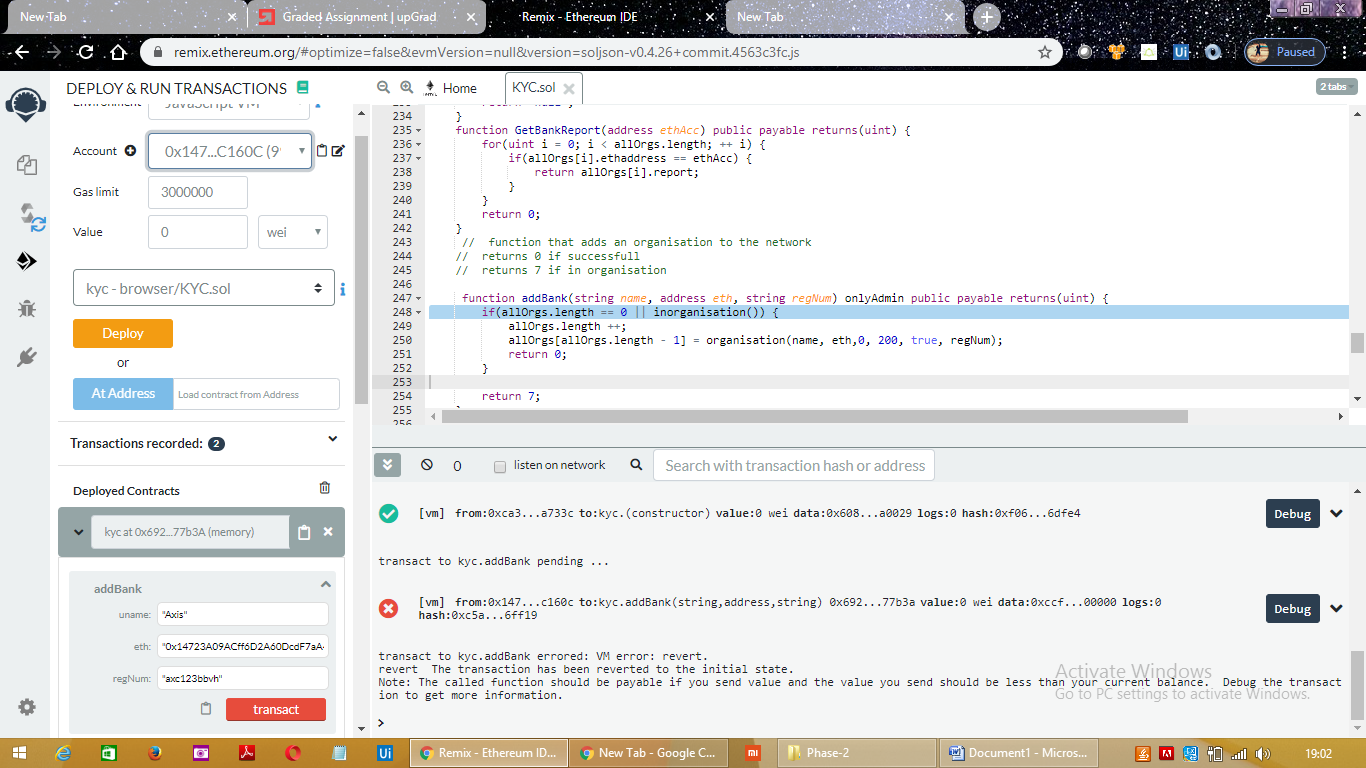
And paste in eth field under addBank ,and hit on transact.



If bank is added successfully it returns 0.



If other than admin if any other bank tries to addbank to the network it is not allowed. Only admin can add or remove bank and can modify KycPermission.



Step 10:Now change the account to Bank account address .The address one which was added by admin as bank.

Here in this case the bank added by admin is Axis,

Name: “Axis”

Eth: “0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C”

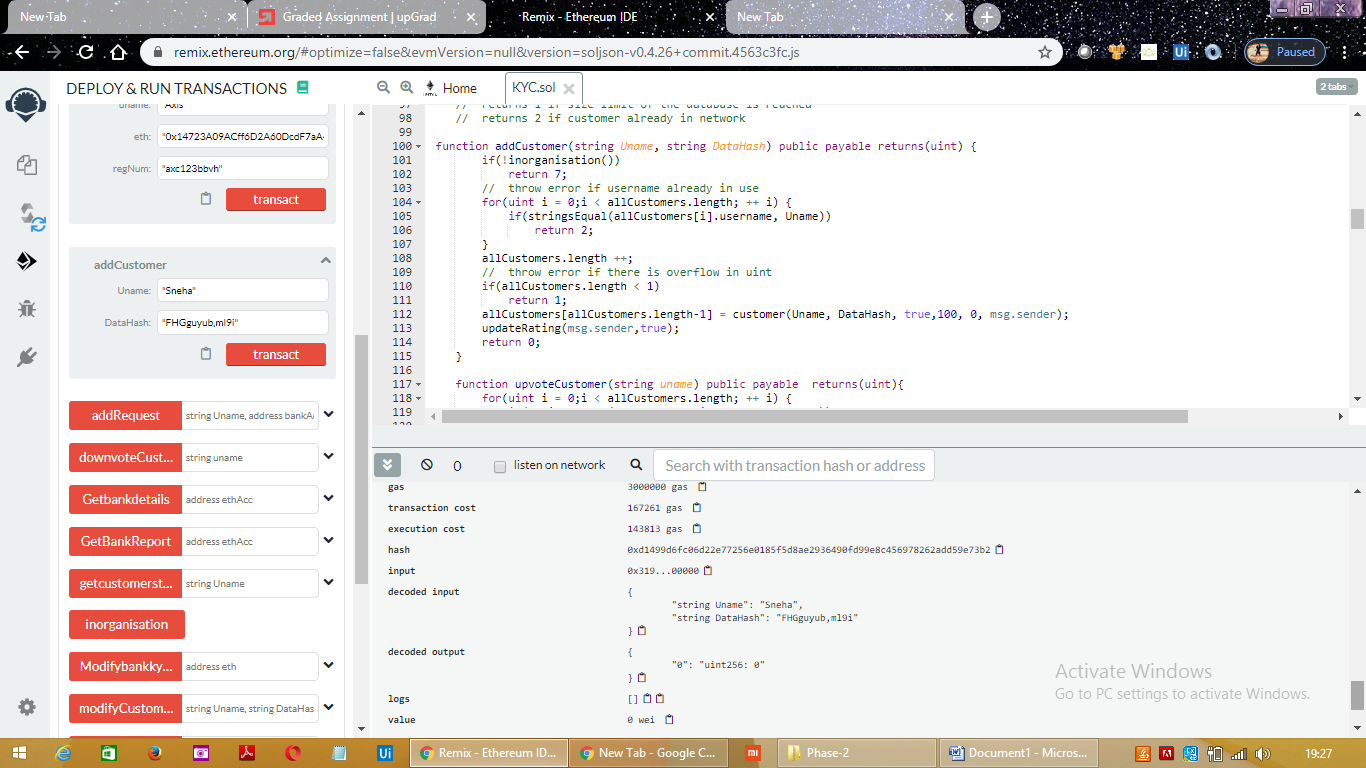
Regnum:”axc123bbvh”



Now Axis bank ,with bank address “0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C” can perform following operations like

Step 11:Add Customer

Bank can add customer to the customer list by providing customer name rand customer data



Ex:

Uname:”Sneha”

Customerdata: "FHGguyub,ml9i"

Hit on transact.

Addcustomer returns=0 if customer is added successfully.

=7 if bank is not a part of organisation

=1 if there is overflow in uint

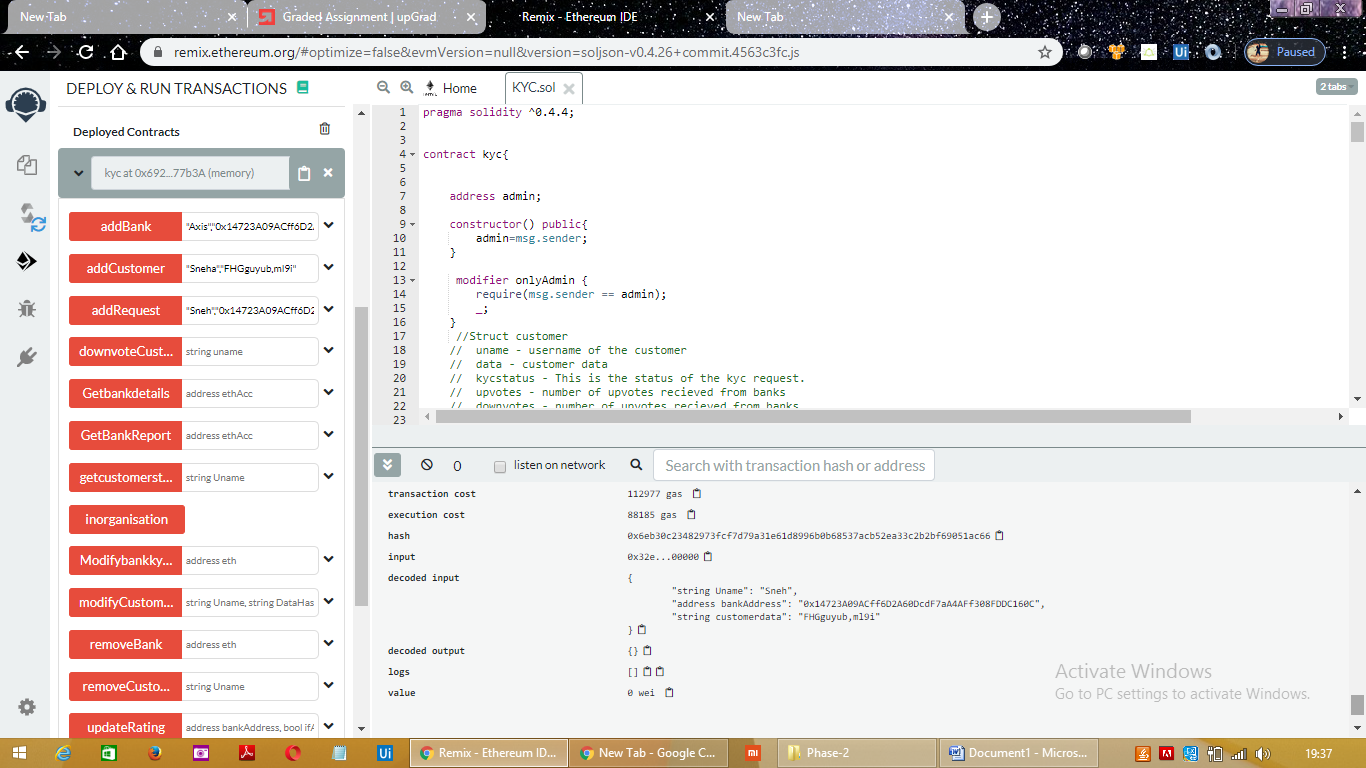
=2 if customer is already present in customer list.

Step 12:Bank can add kyc request by providing username of customer, bankaddress and customerdata.

Uname:”Sneha”

Address:” “0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C”

Customerdata: "FHGguyub,ml9i"

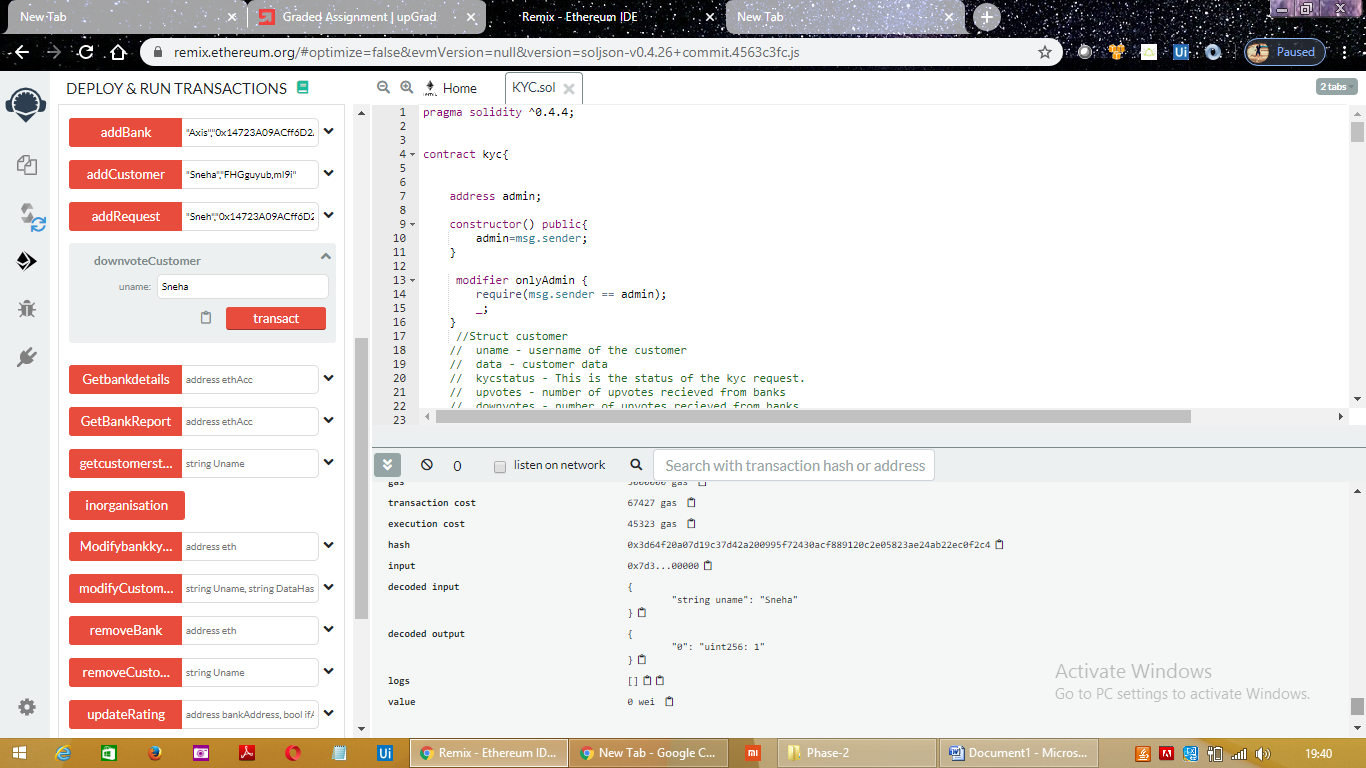


Step 13:Bank can upvote or downvote customer by providing customer name.

Upvote:



Downvote:

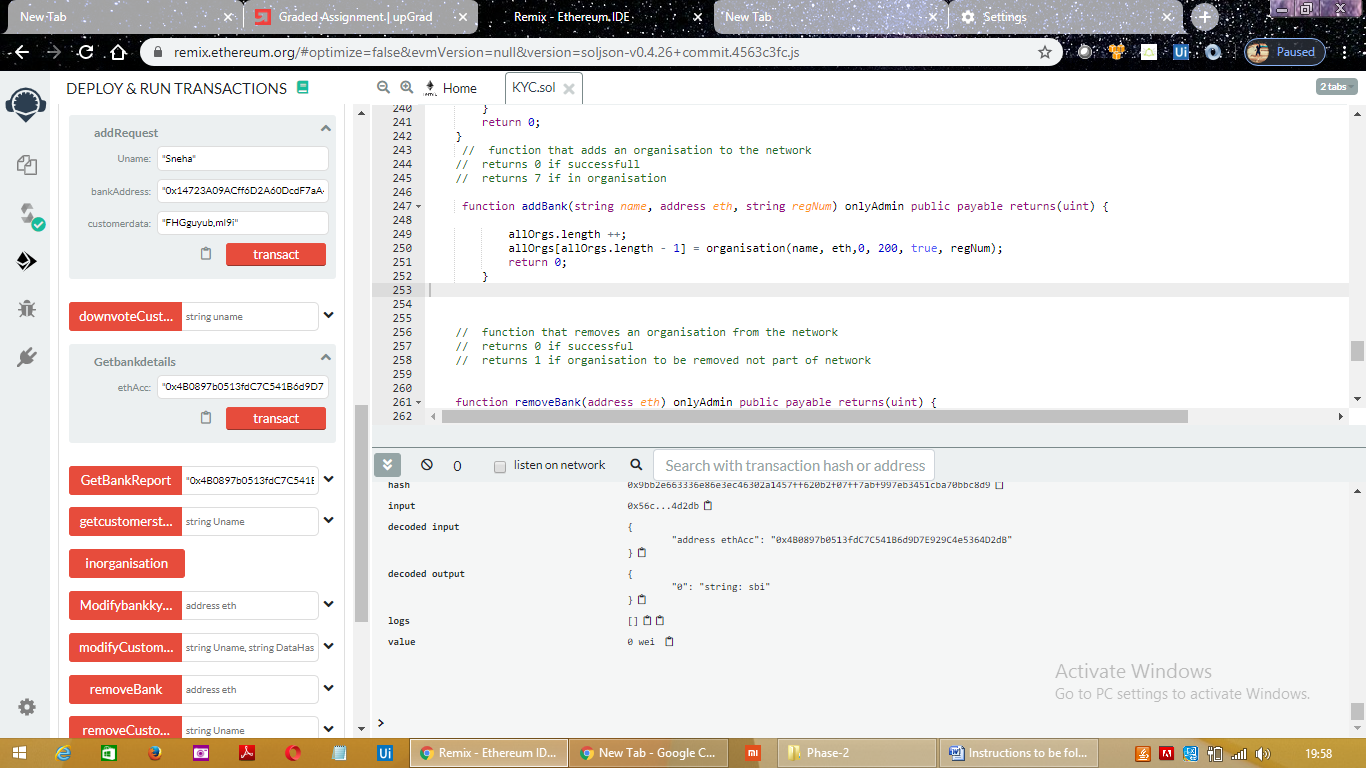


Step 14:GetBankDetails

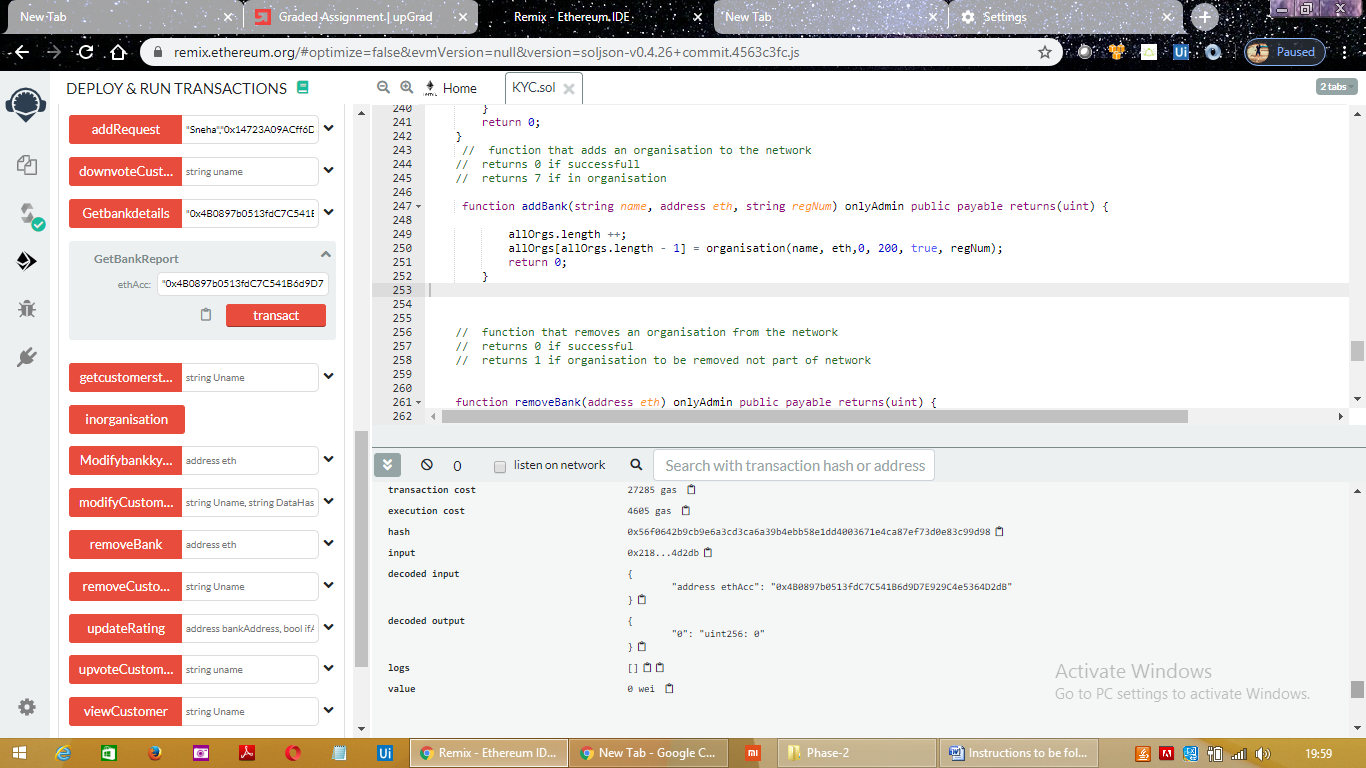
Other banks can get any bank detail present in network by providing bank address

Ex: "0x4B0897b0513fdC7C541B6d9D7E929C4e5364D2dB"

Returns Bank name

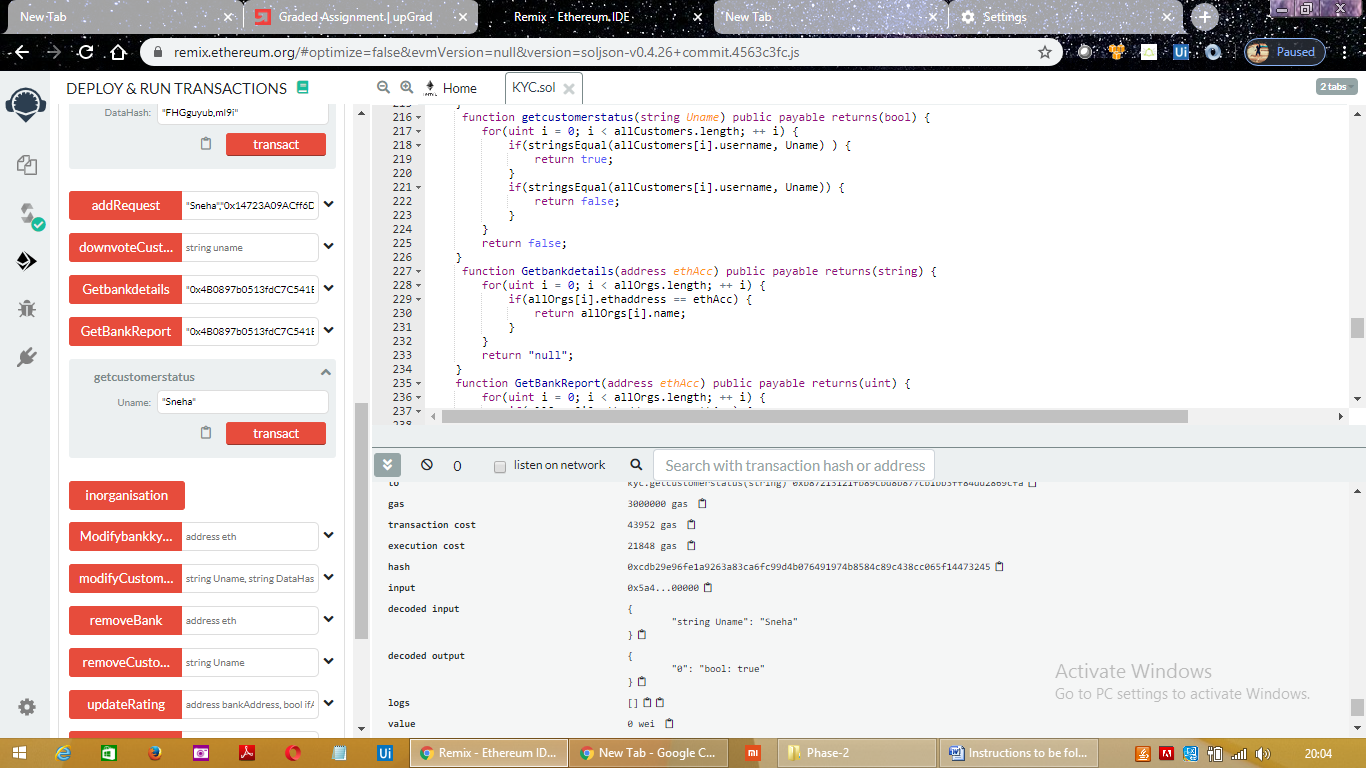


Step 15 :GetBankReport: returns bank report

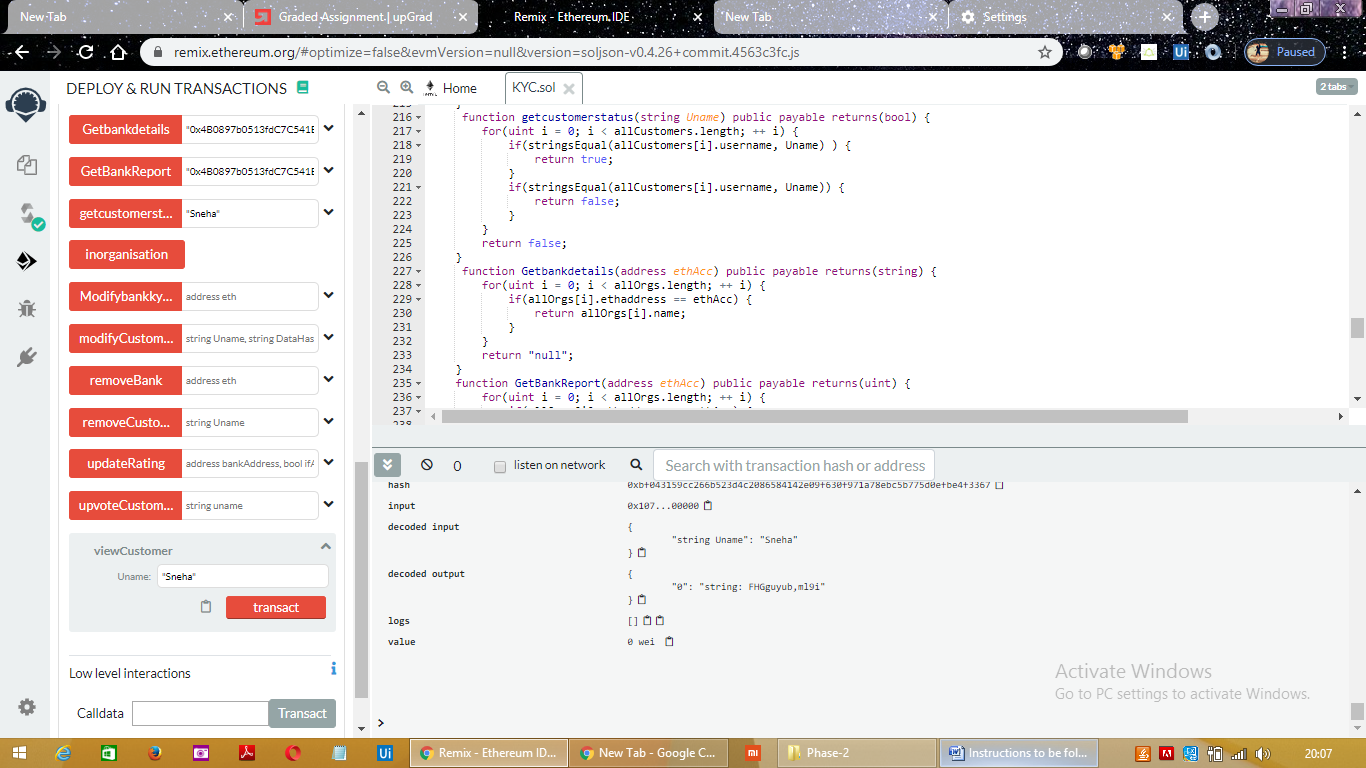


Step 16:Get Customer status:Returns customer kyc status from the smart contract. If true then the customer is verified.

It takes customer name as input.

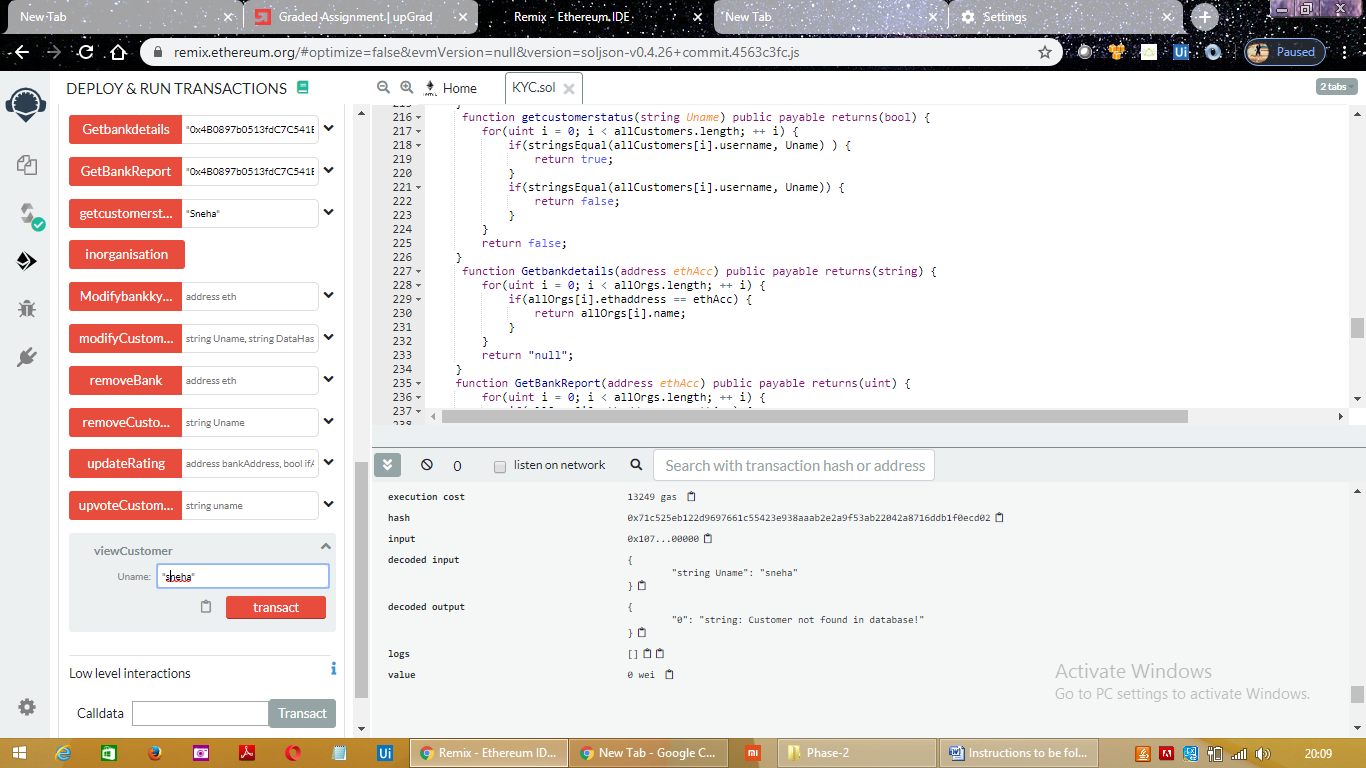


Step 17:View Customer



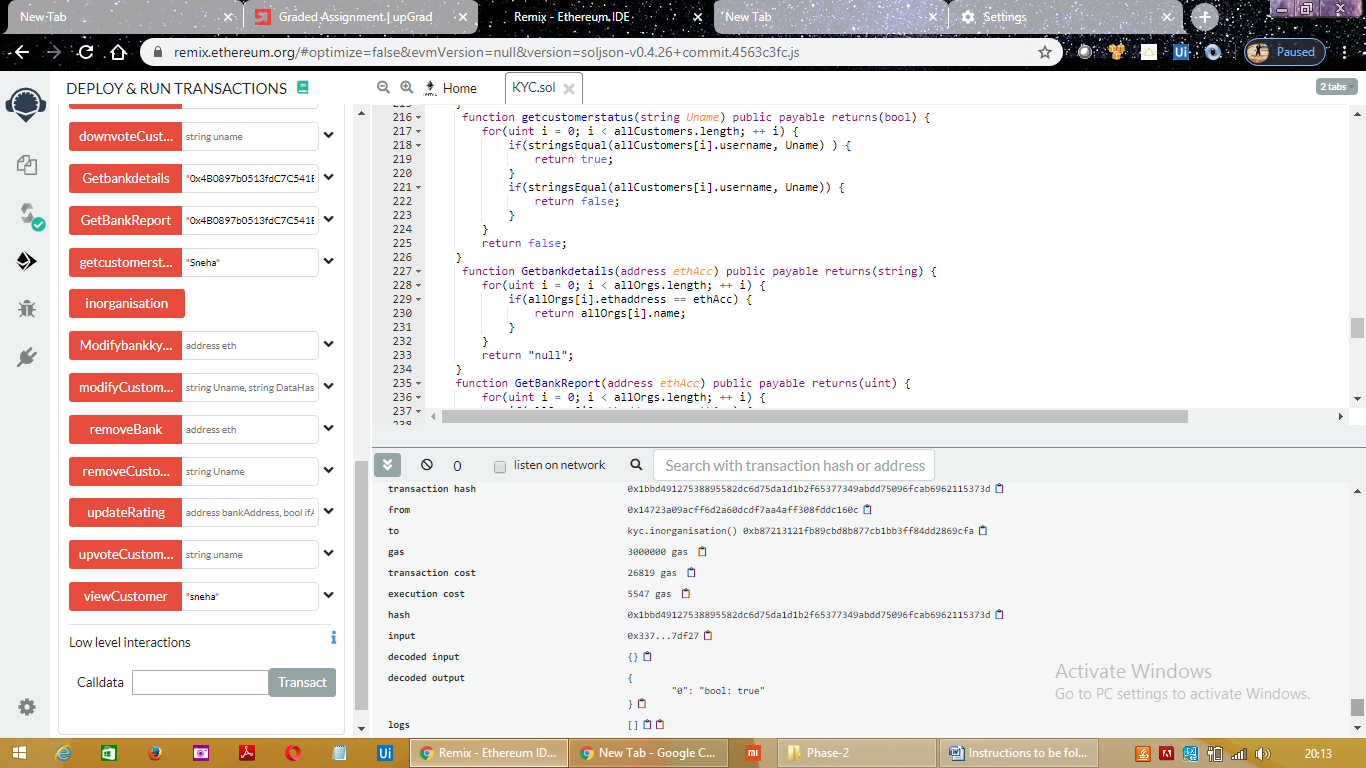
It takes username as input and and displays customer data and username if present.

If customer is not present displays as”Customer not present”



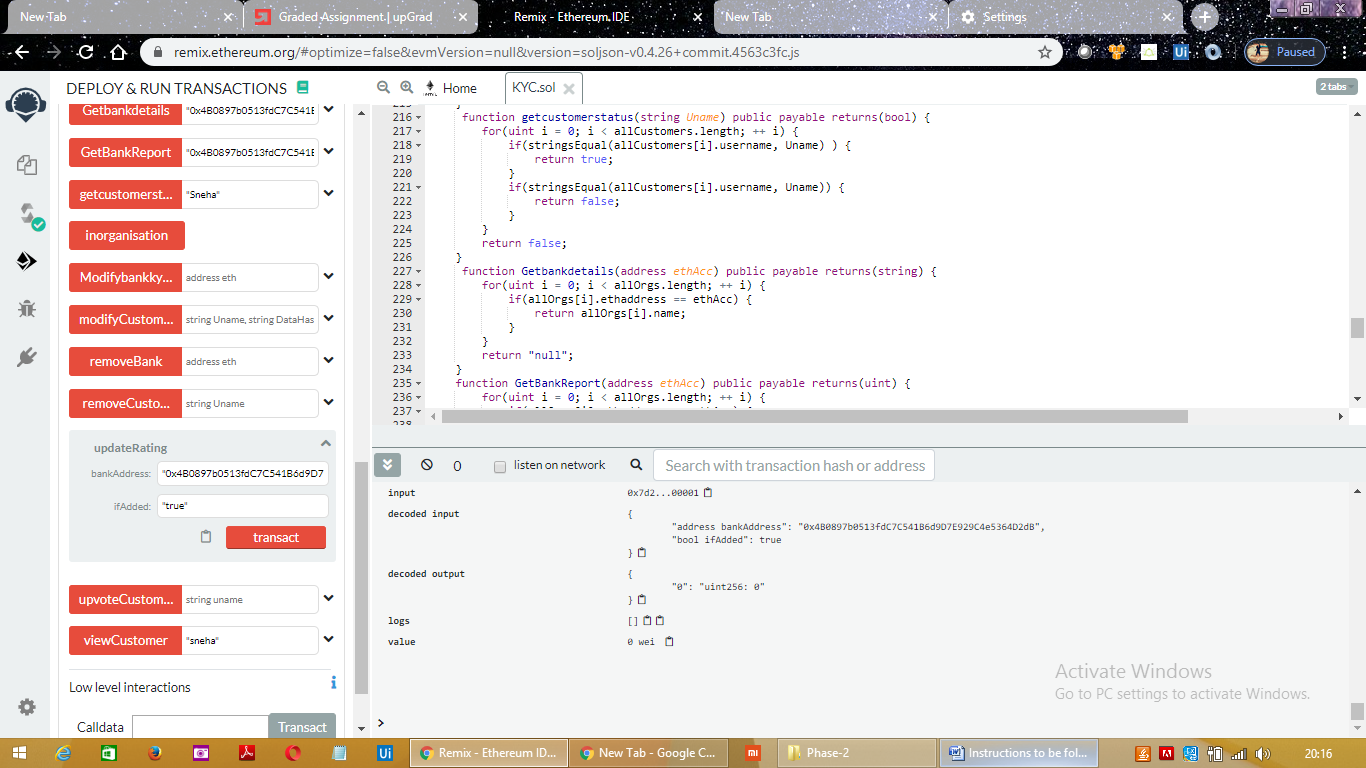
Step 18:inorganosation

It returns true if bank is a part of organisation



Step 19:Updateratings

Banks can update ratings of other banks by providing bank address and if bank is added in network.



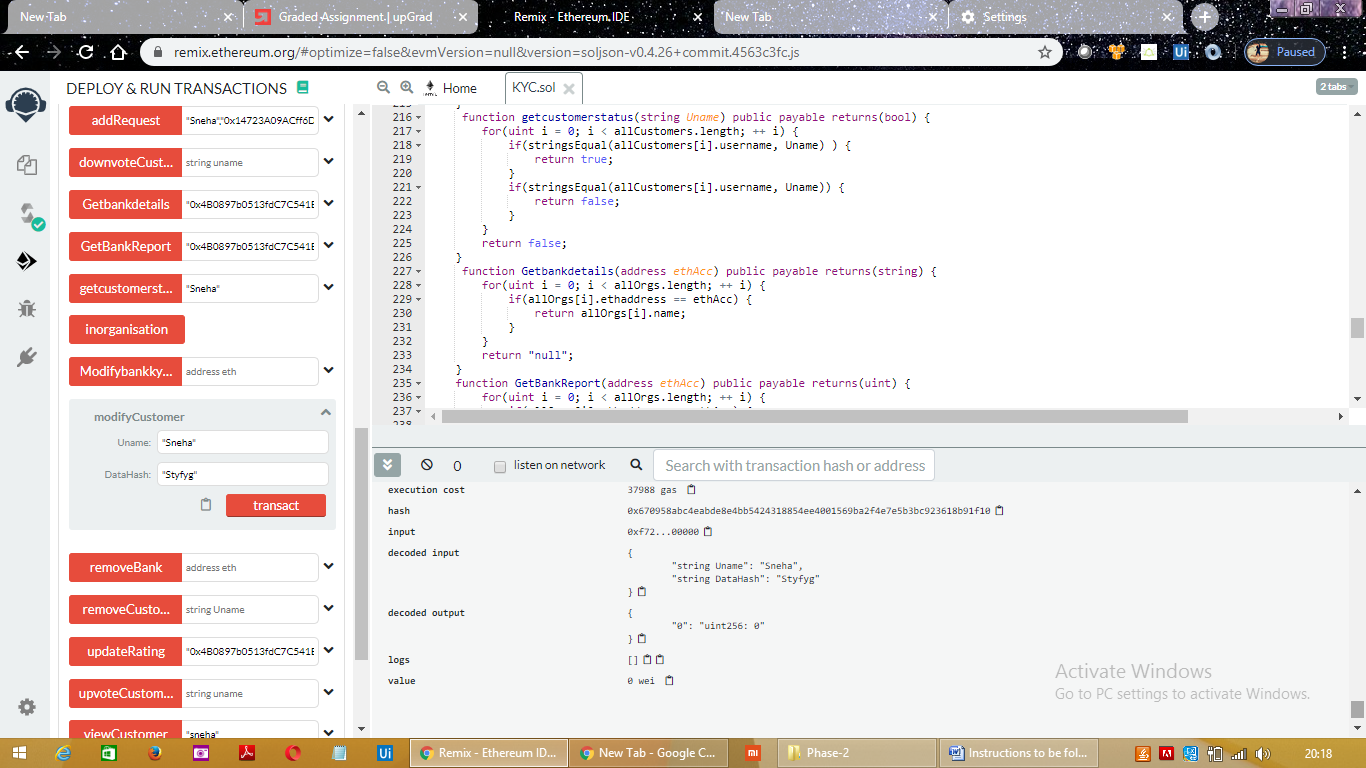
Step 20: ModifyCustomer

Bank can modify customer data by providing customer name and datahash

returns 0 if successful

returns 7 if not in organisation

returns 1 if customer profile not in database



Step 21:RemoveCustomer

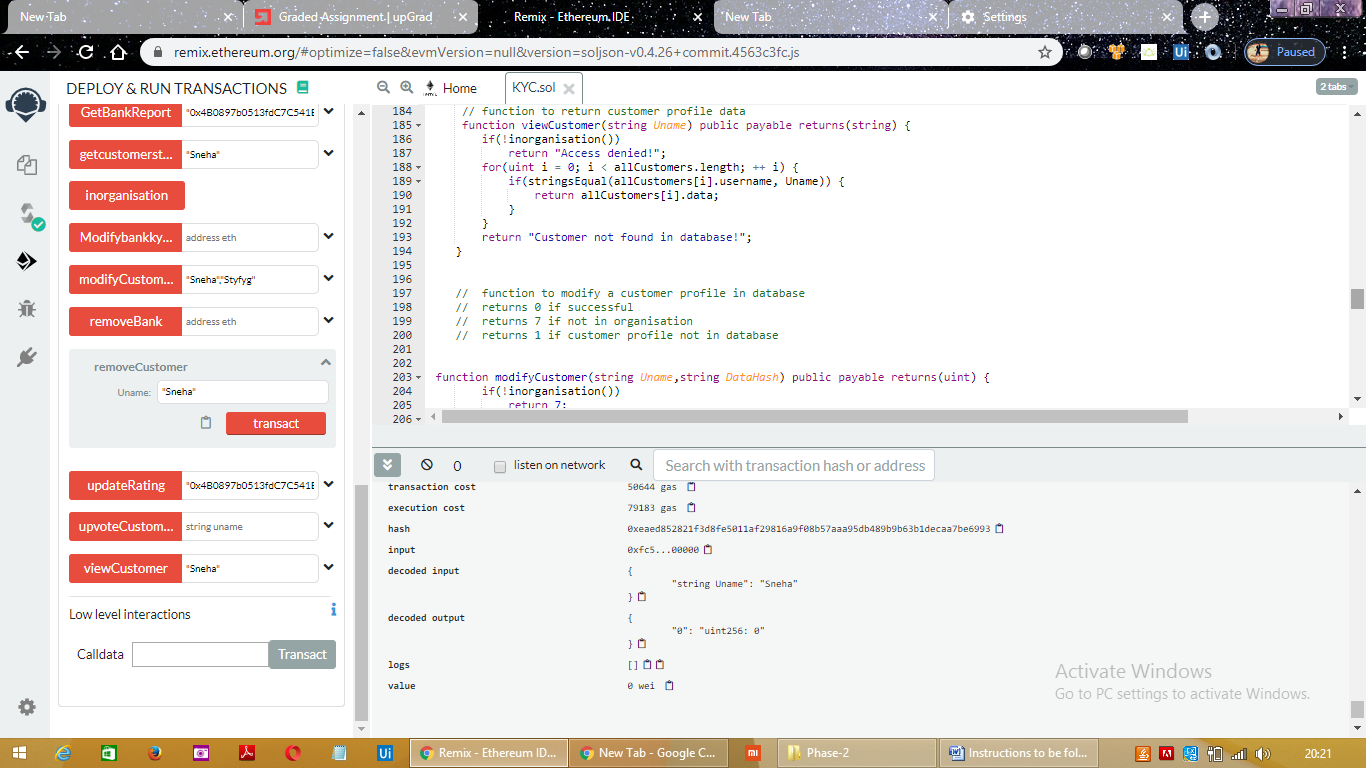
function to remove customer

returns 0 if successful

returns 7 if bank not in organisation

returns 1 if customer profile not in database

Takes usename as input



Step 22:Admin can modify bank KycPermission by providing bank address.

If Bank KycPermission is set to true ,admin can change to false

Or

If Bank KycPermission is set to false ,admin can change to true

Step 23:RemoveBank

Admin can remove bank from the network by providing bankaddress.

Ex: 0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C

returns 0 if successful

returns 1 if organisation to be removed not part of network

