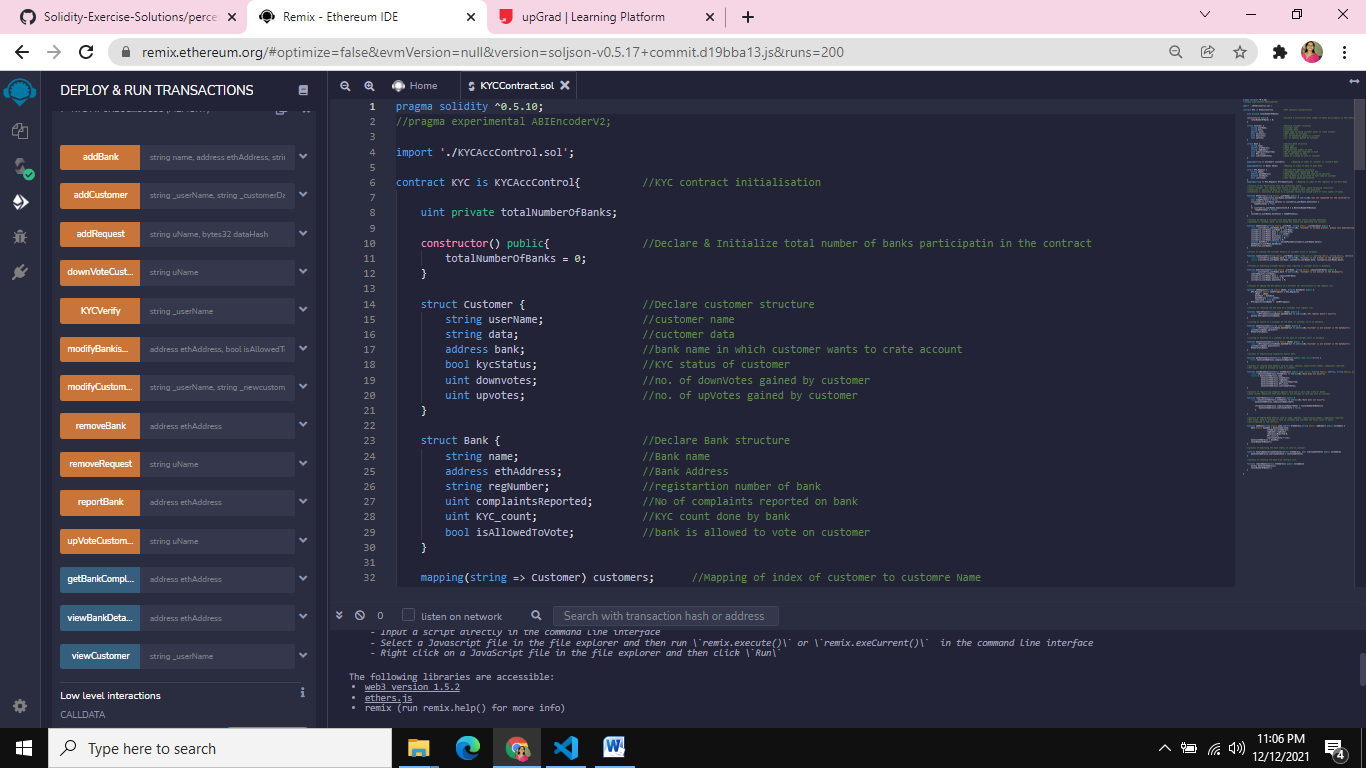
* **KYCAccControl.sol** - It contains modifier for ***AdminOnly***  interfaces.
* **KYCContract.sol** – It contains smart contract. Flow as follows.
* Deploy **KYCContact.sol** in solidity. Following is output screen after deploying.



Flow sequence:

Step 1: Add 5 accounts using AddBank() Button. It has to be invoked from admin address(First account).

addBank(string memory name,address ethAddress,string memory regNumber)

* addBank('SBI', 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2, '1RV22IS001');
* addBank('HDFC', 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db ,'1RV22IS002');
* addBank('KOTAK' , 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB , '1RV22IS003');
* addBank('ICICI' , 0x617F2E2fD72FD9D5503197092aC168c91465E7f2 , '1RV22IS004');
* addBank('AXIS' , 0x17F6AD8Ef982297579C203069C1DbfFE4348c372 , '1RV22IS005');

Step 2: Existing Bank details can be viewed using ViewBank details function.

viewBankDetails(address ethAddress)

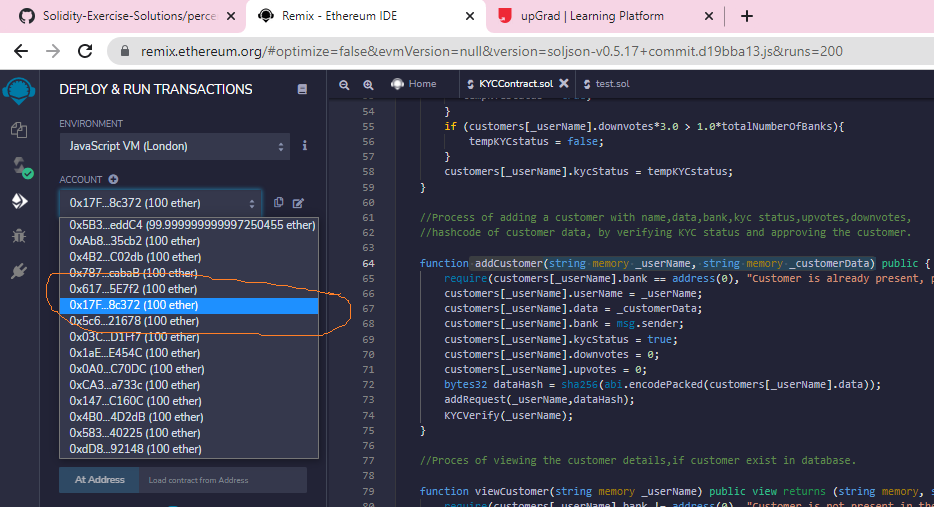
* viewBankDetails(0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2);

Step 3: To add customers to the particular bank

* First change the sender address to the bank address in Remix for the bank that wants to add the customer (as shown in snapshot below).

 addCustomer(string memory \_userName, string memory \_customerData)

Ex:  addCustomer(“Sharada”, “identification data”);



* Same function will invoke KYC request list

addRequest(string memory uName, bytes32 dataHash)

And initial KYC verification process.

KYCVerify(string memory \_userName)

* Further change in KYC status can be observed depending upon votes casted by other banks.

upVoteCustomer(string memory uName)

downVoteCustomer(string memory uName)

Step 4: customer details can be viewed using

viewCustomer(string memory \_userName)

 viewCustomer(“Sharada”);

Step 5: bank can modify customer information and remove request before modifying.

removeRequest(string memory uName)

 modifyCustomer(string memory \_userName, string memory \_newcustomerData)

modifyCustomer(“Sharada”, “identification data”);

Step 6: Admin can change the permission to vote for the bank (Admin access required).

modifyBankisAllowedToVote(address ethAddress, bool isAllowedToVote)

Admin can remove Bank (Admin access required)

removeBank(address ethAddress)

Step 7: If complaints against bank are more than 1/3rd of total number of banks, then such bank is not allowed dot vote on customer.

reportBank(address ethAddress)

Complaints are registered through

getBankComplaints(address ethAddress)