



MAULANA AZAD NATIONAL URDU UNIVERSITY

- BLOCKCHAIN technology
LAB RECORDS

Name: JAMSHED ALAM

Roll No.: 20MMCA002HY

Course: MCA-3rd semester

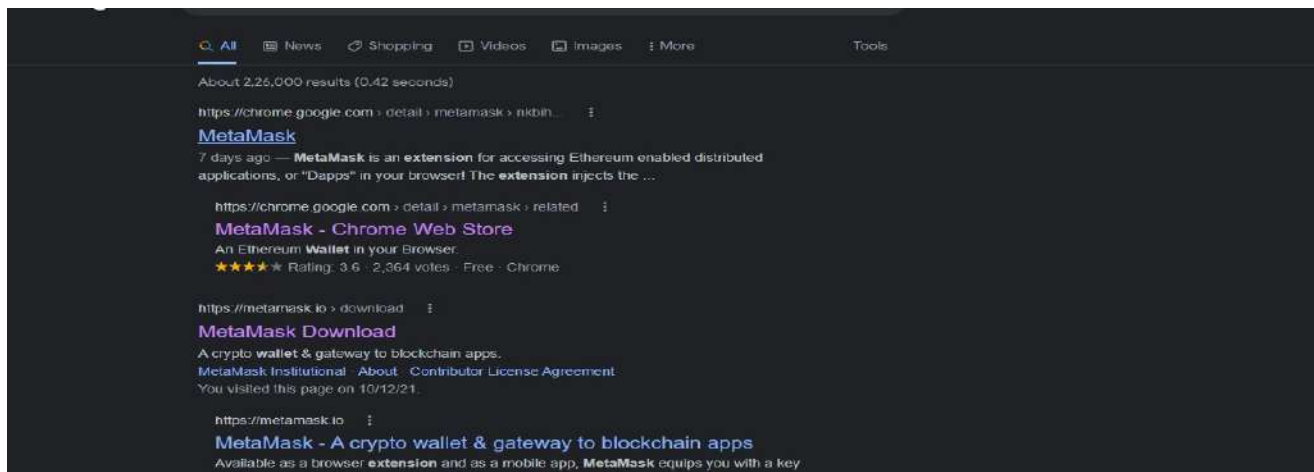
Department: CS&IT

SUBJECT: BCT LAB

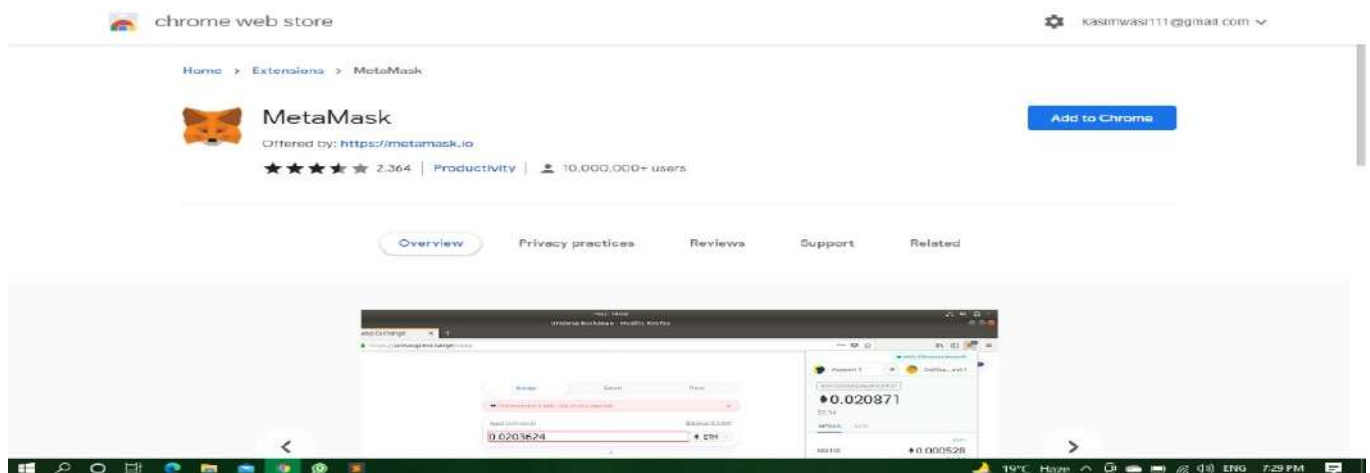
Practical No.1

MetaMask Wallet

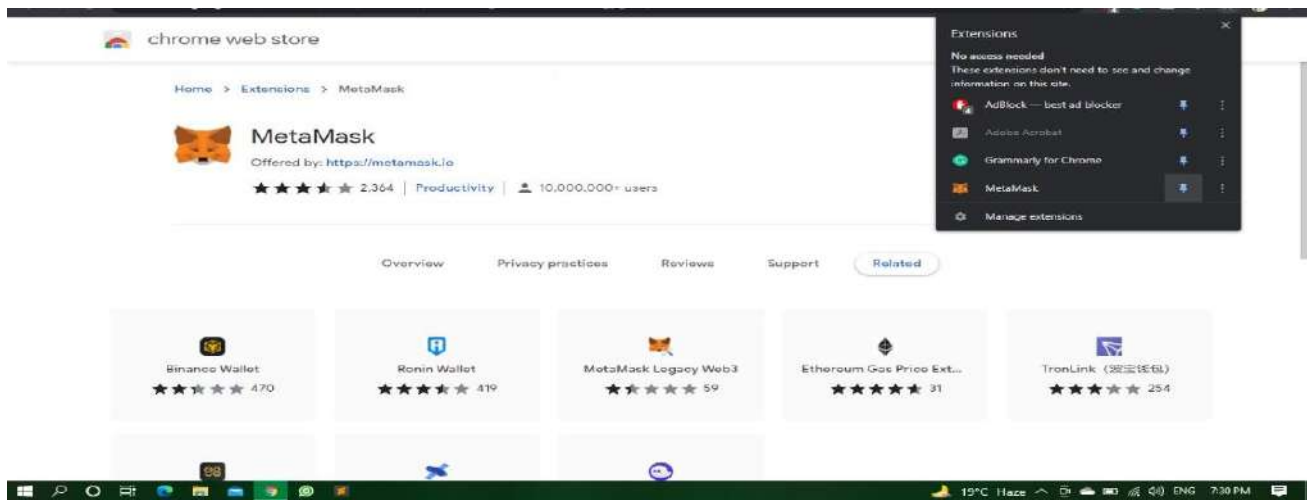
Step-1: Open Chrome Browser and search for “Metamask Wallet” and add extension download and open the first link.



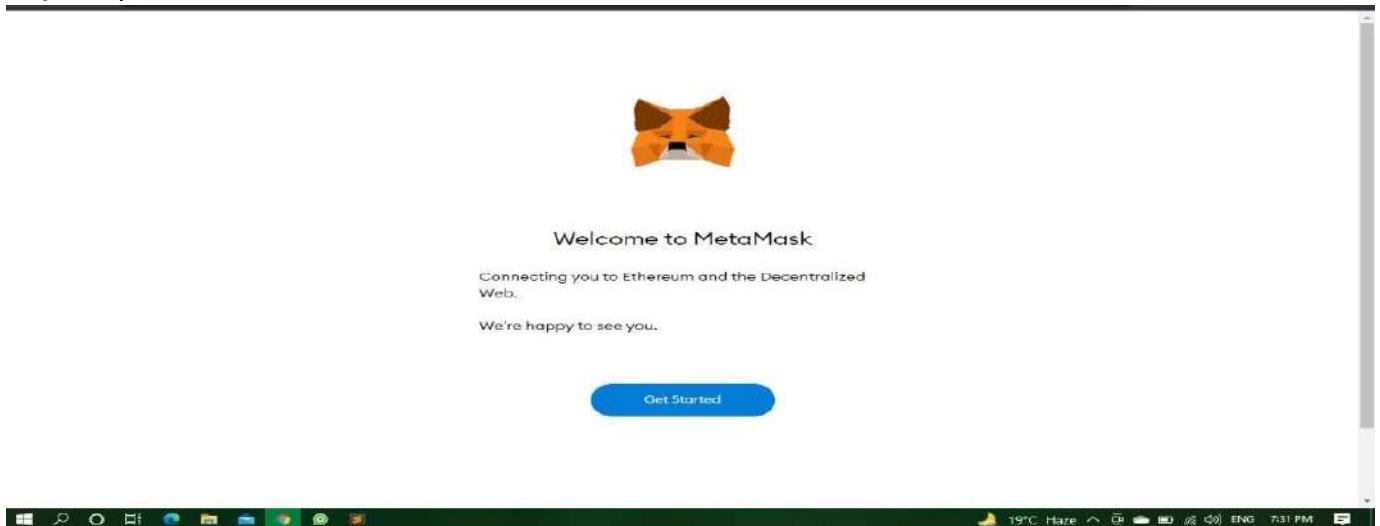
Step-2: Click on “Add to Chrome” and simply add it in your extension list.



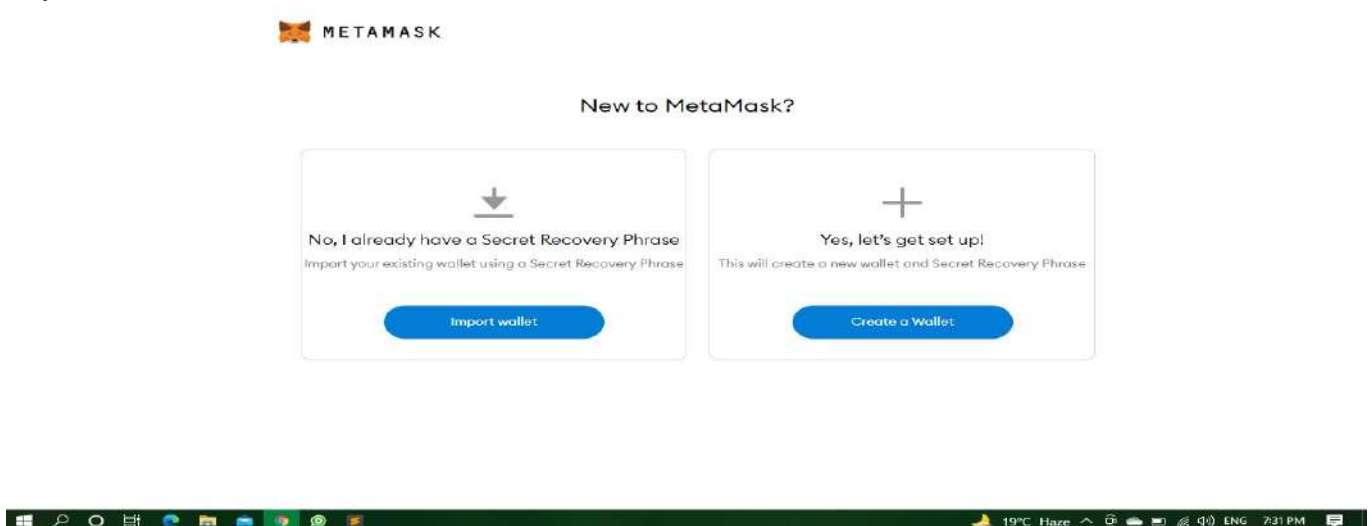
Step-3: click on extension option and pin the Metamask Wallet extension.



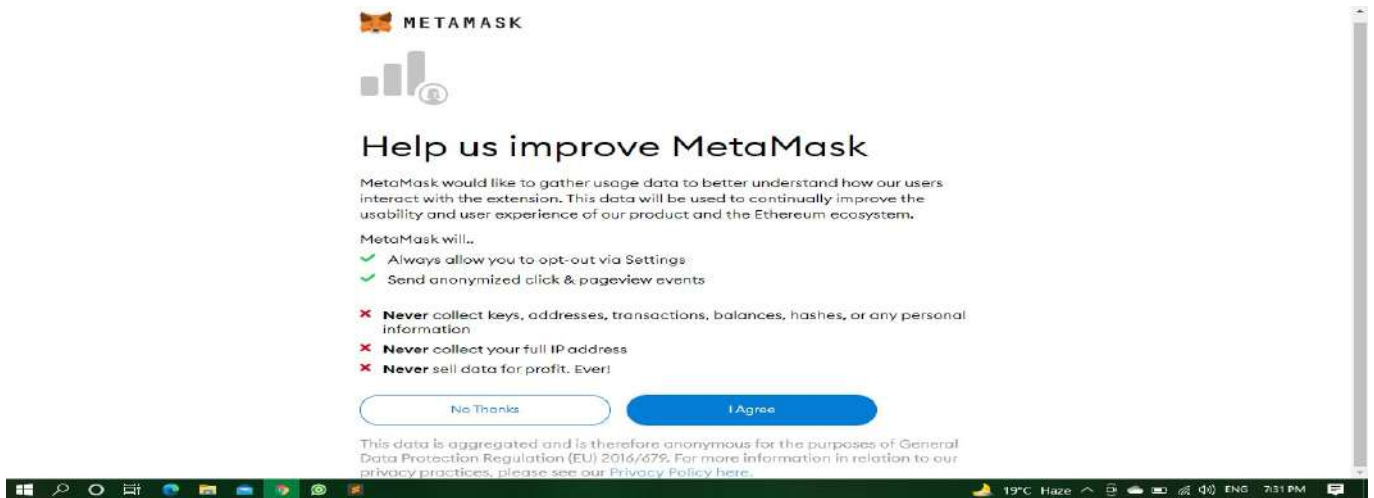
Step-4: Open Metamask Wallet and click “Get Started”.



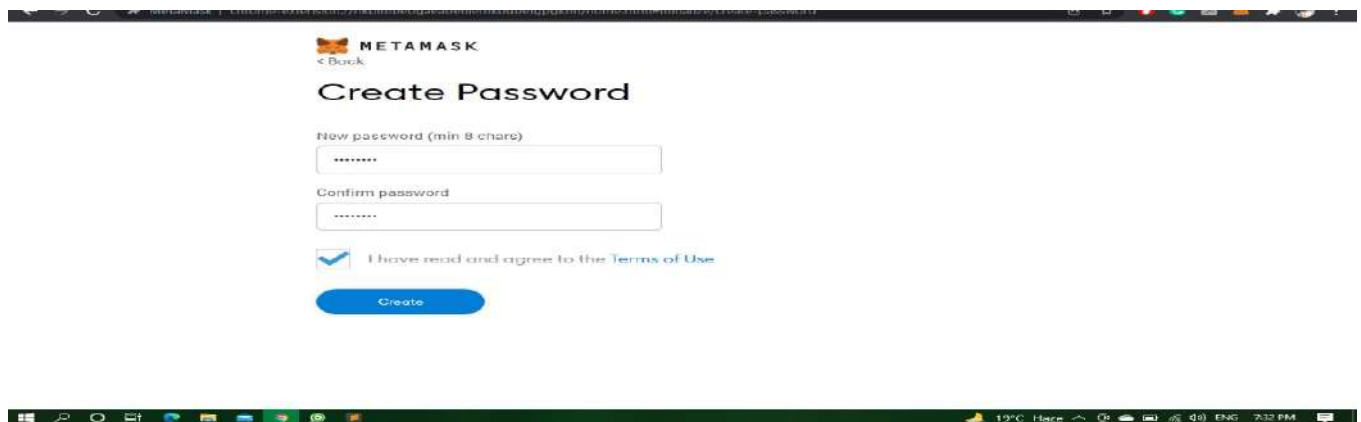
Step-5: Click on “Create a Wallet”.



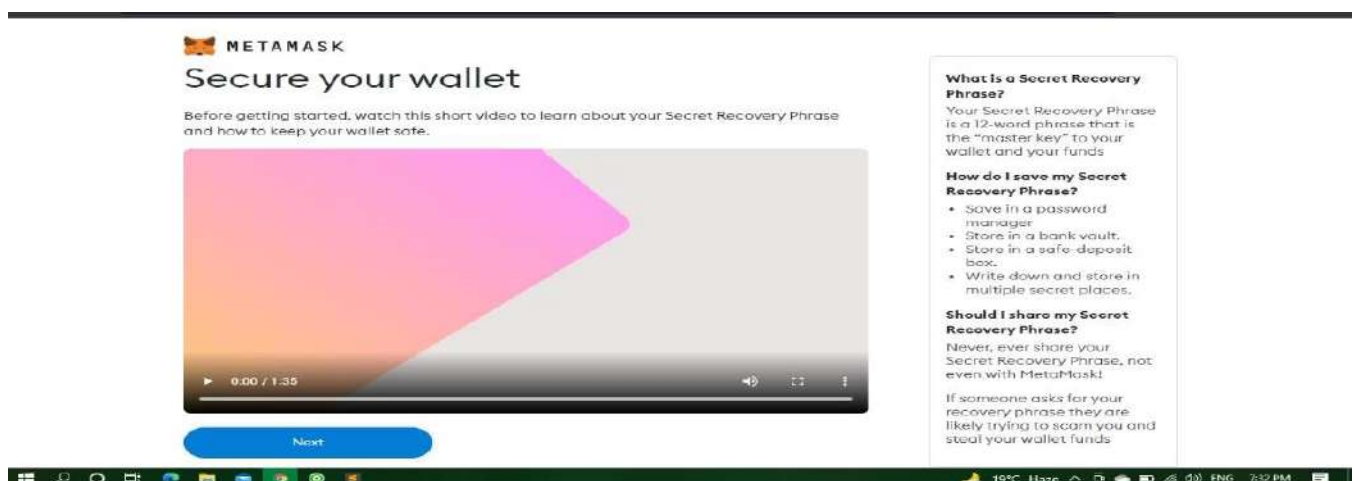
Step-6: Click on “I agree”.



Step-7:now Create Password



Step-8: Click on "Next" tab.



Step-9: Reveal the Secret Words then copy it and paste it in any document file and click on "Next".

Secret Recovery Phrase

Your Secret Recovery Phrase makes it easy to back up and restore your account.

WARNING: Never disclose your Secret Recovery Phrase. Anyone with this phrase can take your Ether forever.

hunt indicate diamond shine purse
spin dinner weather oppose
mosquito twist client

Remind me later

Next

Tips:

Store this phrase in a password manager like 1Password.

Write this phrase on a piece of paper and store in a secure location. If you want even more security, write it down on multiple pieces of paper and store each in 2 - 3 different locations.

Memorize this phrase.

Download this Secret Recovery Phrase and keep it stored safely on an external encrypted hard drive or storage medium.

Step-10: Confirm Secret Recovery Phrase accordingly.

Confirm your Secret Recovery Phrase

Please select each phrase in order to make sure it is correct.

hunt	indicate	diamond	shine
purse	spin	dinner	weather
oppose	mosquito	twist	client

client	diamond	dinner	hunt
indicate	mosquito	oppose	purse
shine	spin	twist	weather

Confirm

Step-11: After confirming the Secret Phrase simply click on "All Done", and your account has been created.



Congratulations

You passed the test - keep your Secret Recovery Phrase safe, it's your responsibility!

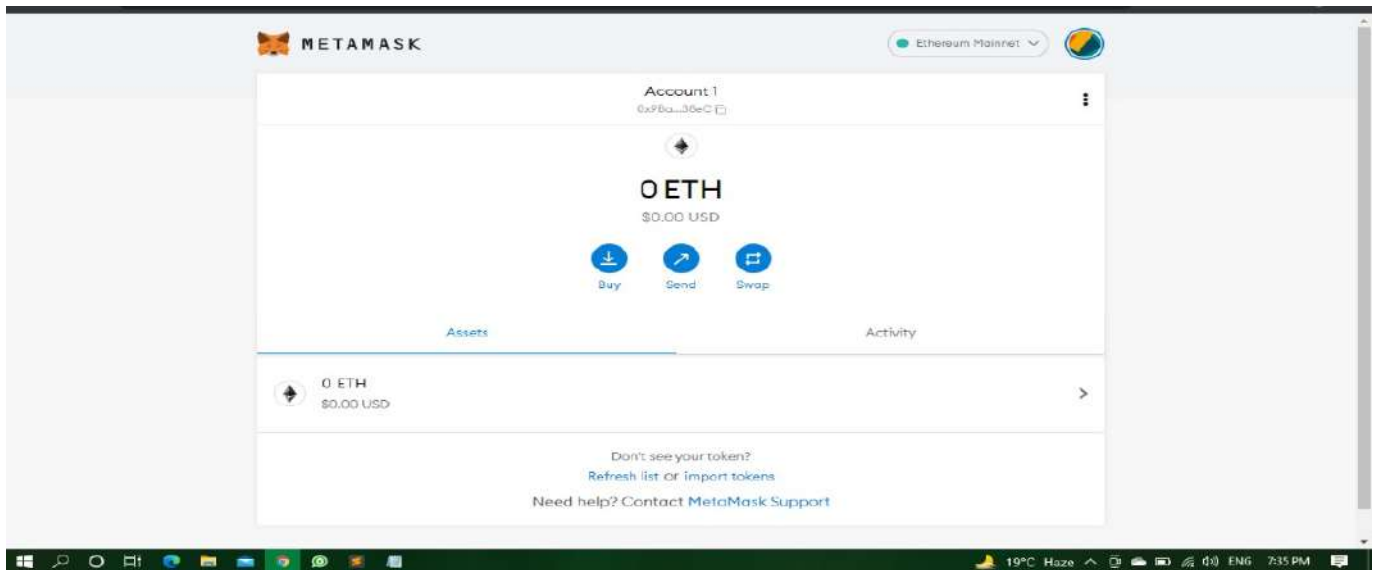
Tips on storing it safely

- Save a backup in multiple places.
- Never share the phrase with anyone.
- Be careful of phishing! MetaMask will never spontaneously ask for your Secret Recovery Phrase.
- If you need to back up your Secret Recovery Phrase again, you can find it in Settings -> Security.
- If you ever have questions or see something fishy, contact our support [here](#).

*MetaMask cannot recover your Secret Recovery Phrase. [Learn more](#).

All Done

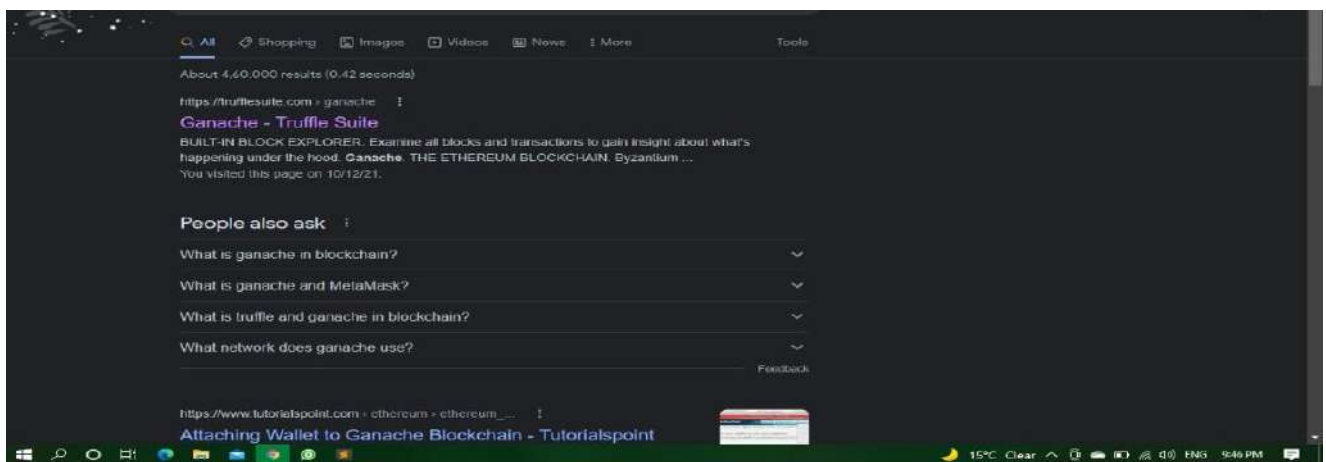
Metamask Wallet will look like.



Practical No.2

Installation of Ganache and Import the Ethereum to Metamask wallet.

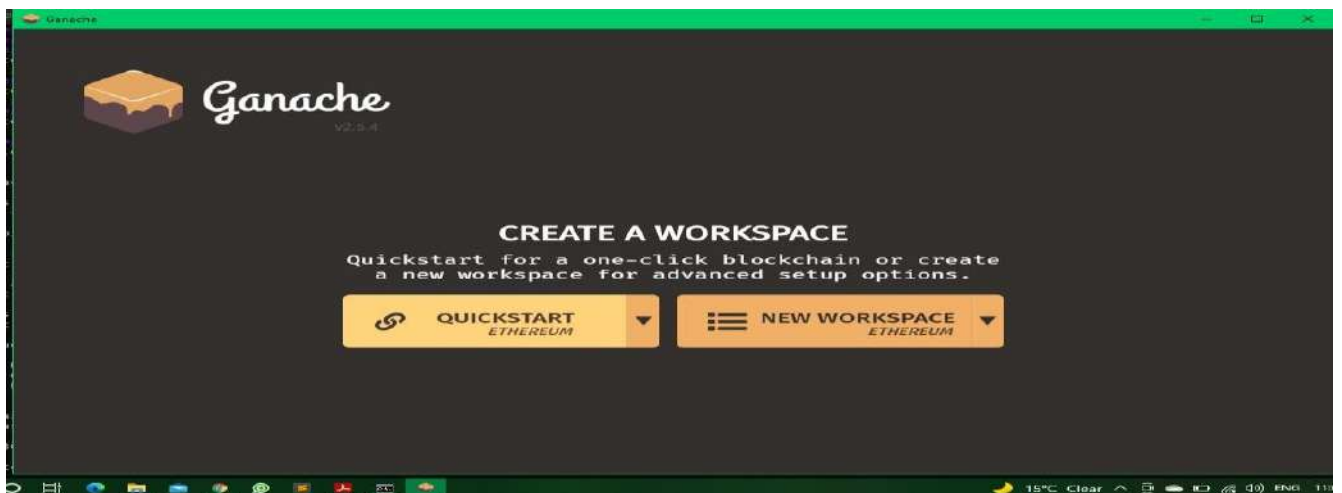
Step-1: Open your Chrome Browser and search for Ganache Wallet and Click on first link.



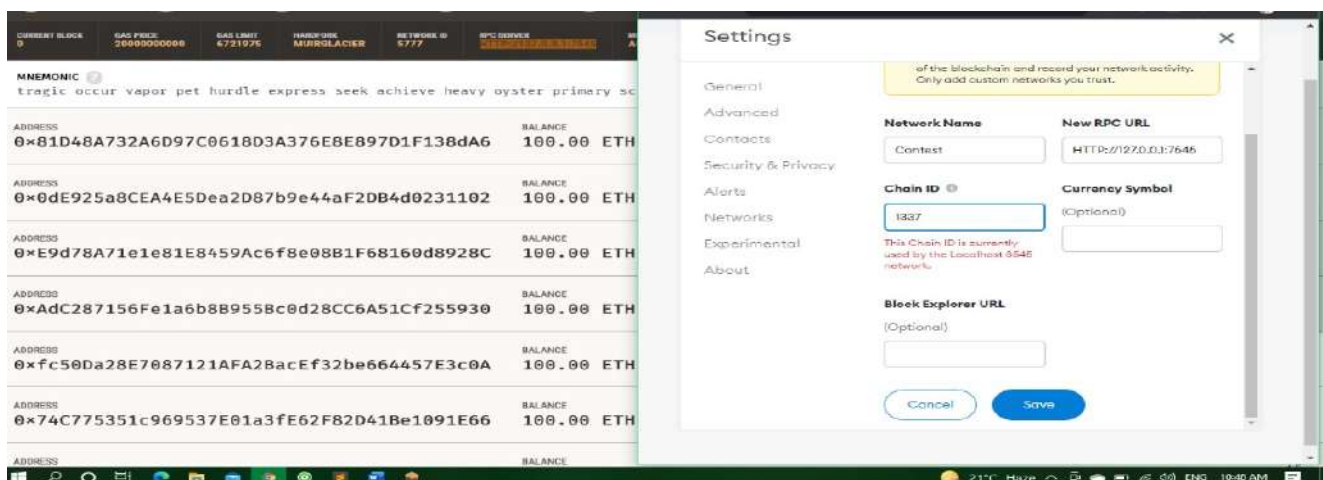
Step-2: Download Ganache for Windows.



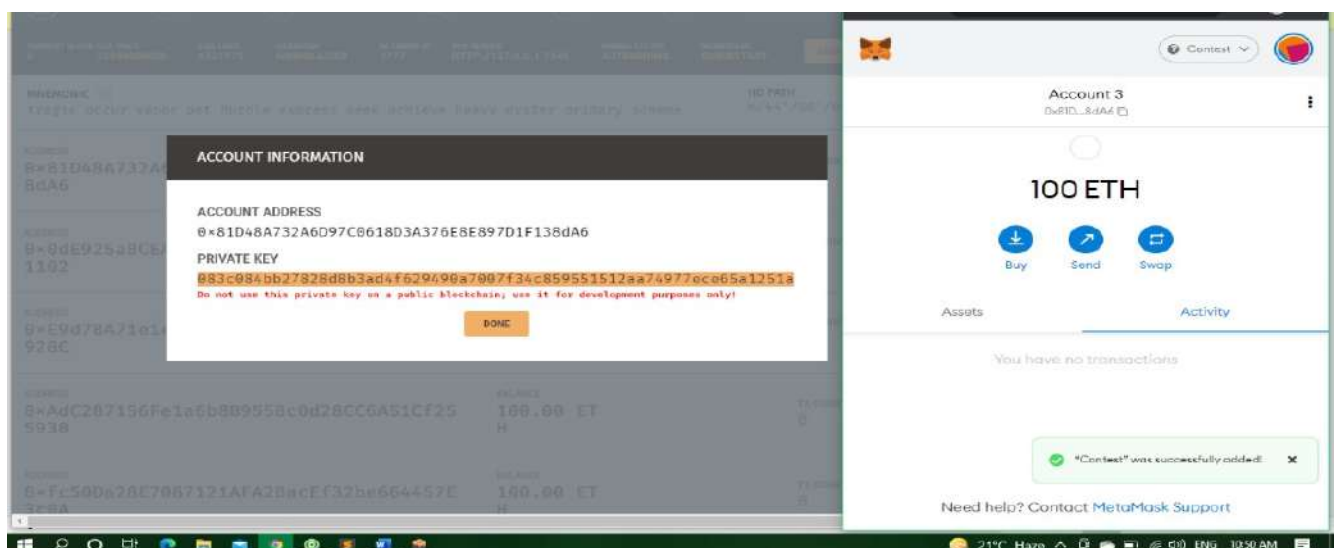
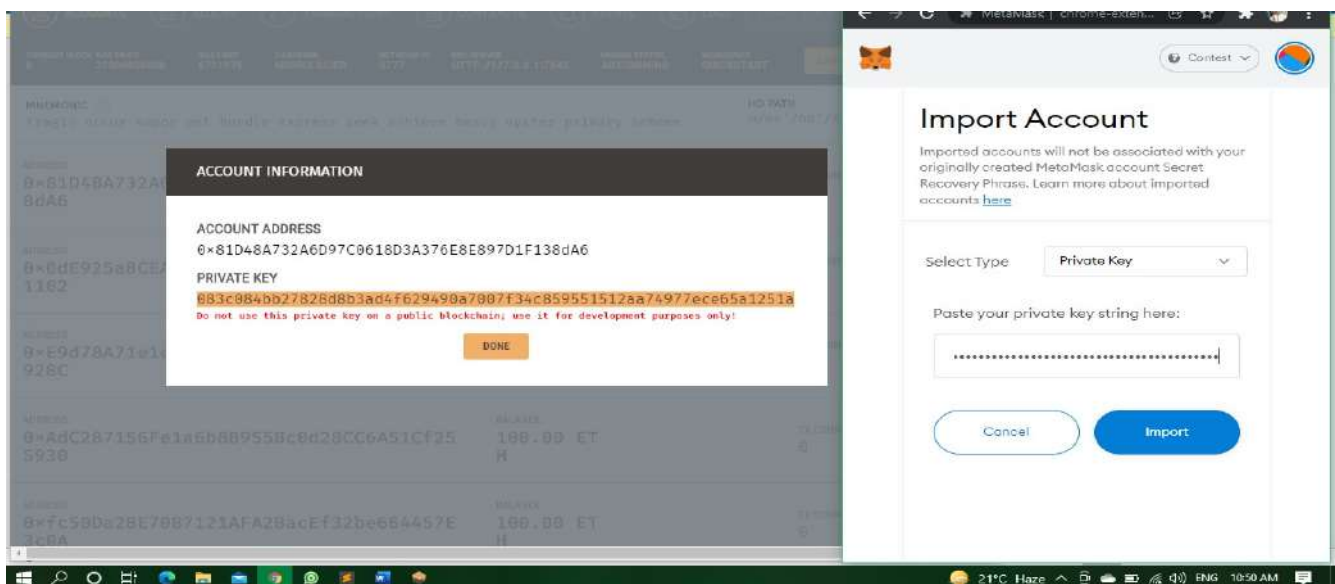
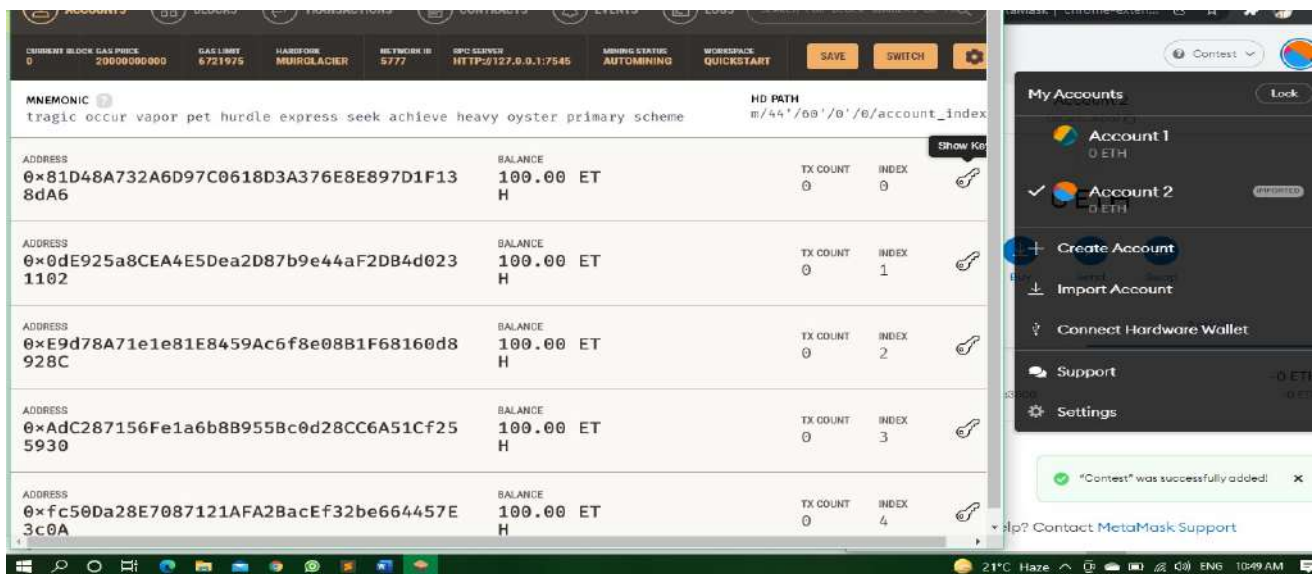
Step-3: Open ganache and click on “Quickstart”



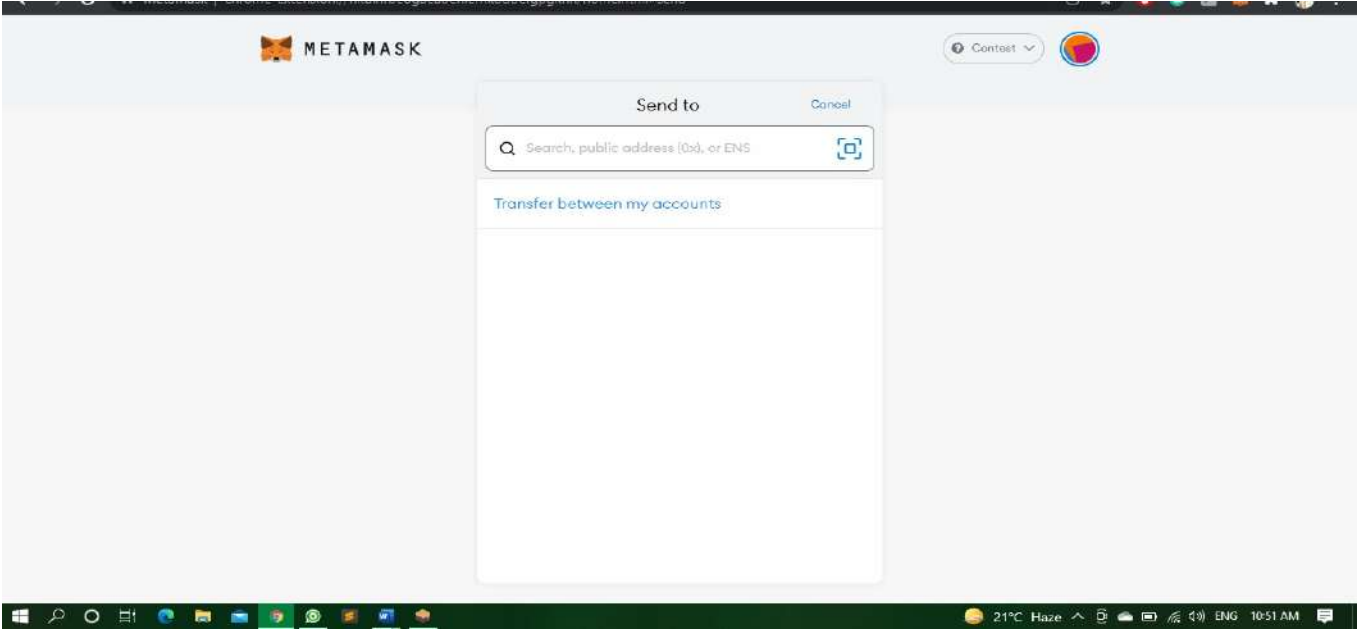
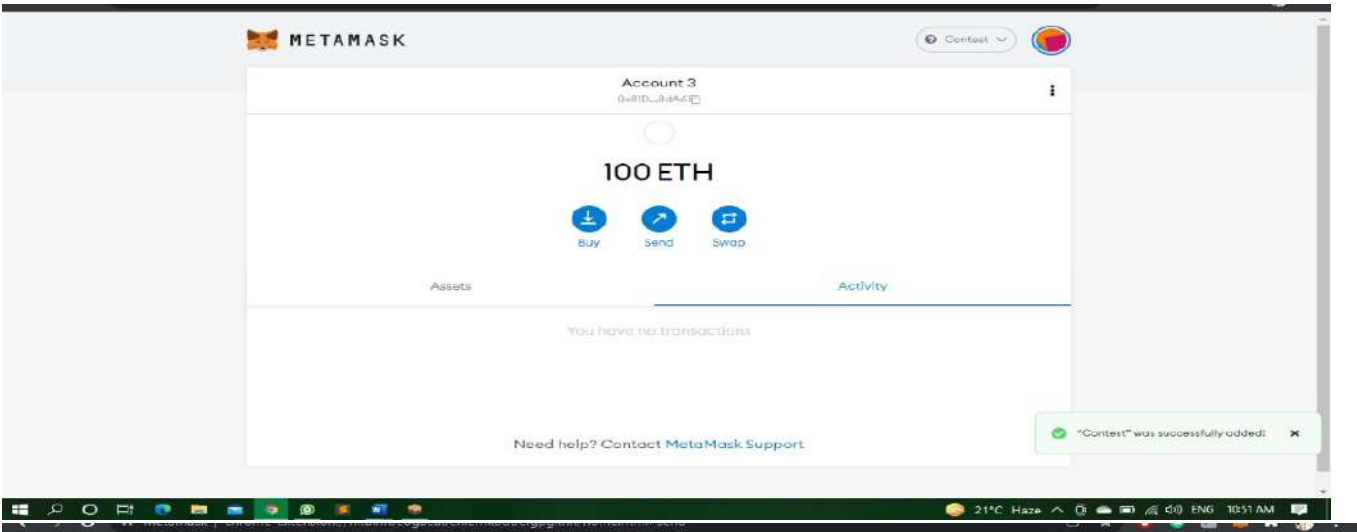
Step-4: Copy the “RPC Server” from Ganache and create a new Network in Metamask by using the RPC server.

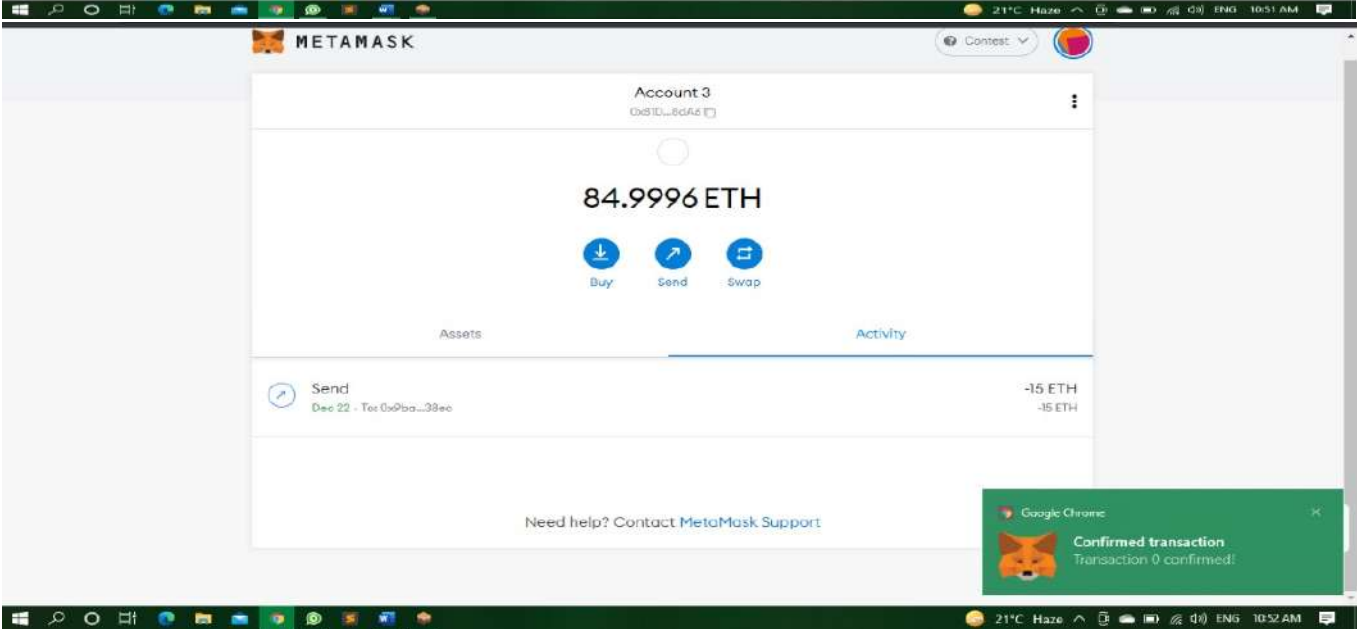
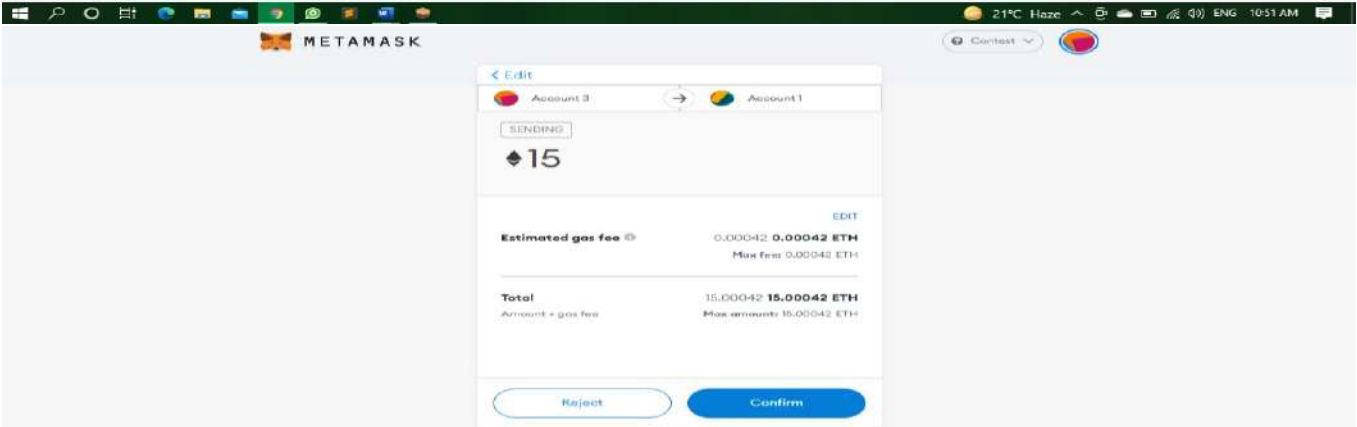
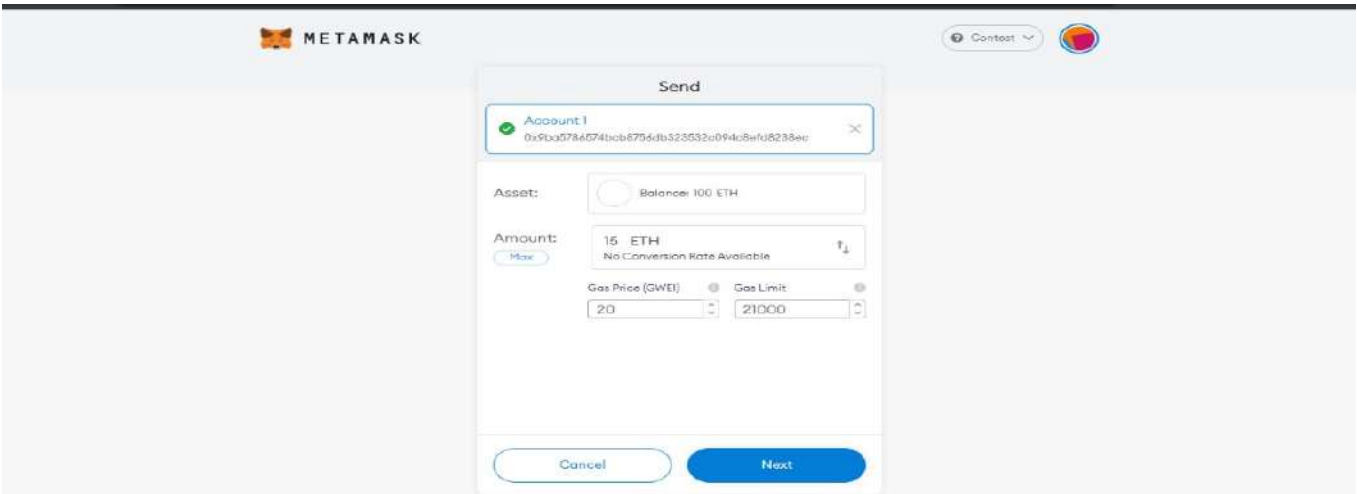


Step-5: Click on Key symbol in Ganache and Copy the Private Key and Import a new account in Metamask by using this Private key, after click on “Import” you will see 100 ETH in your account.



Step-6: Now click on "Send" and transfer between account and choose an account to send some ether and confirm the transaction.

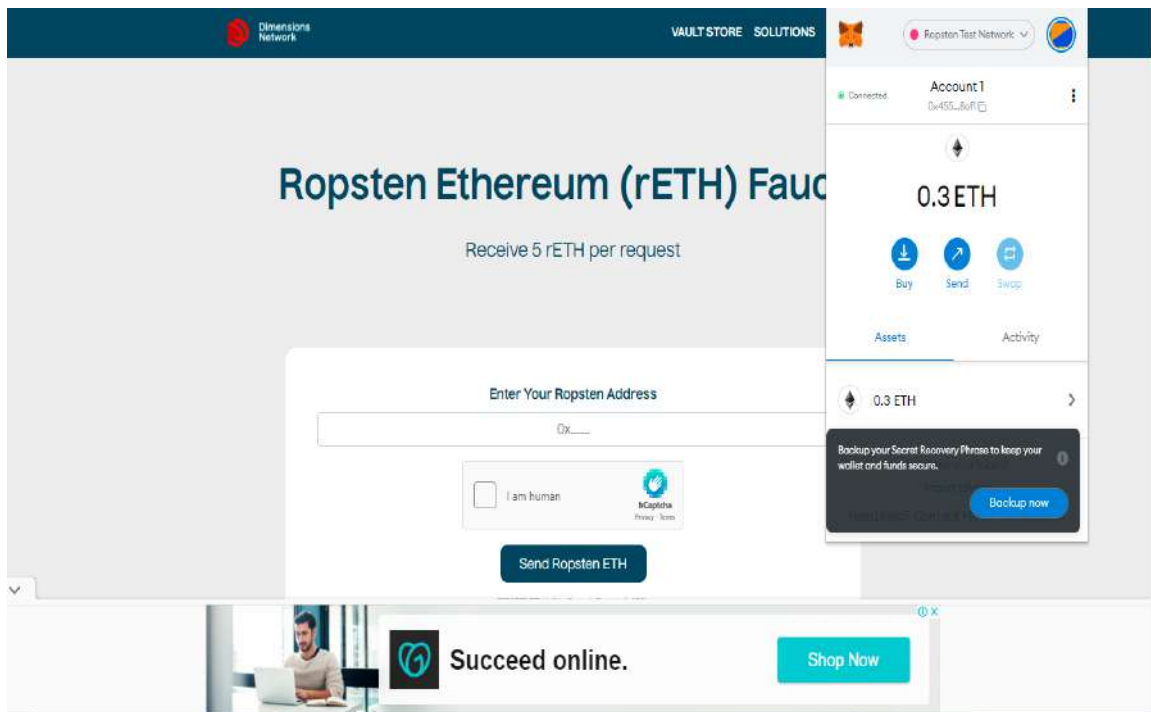




Transaction has been successfully completed.

PRACTICAL -3

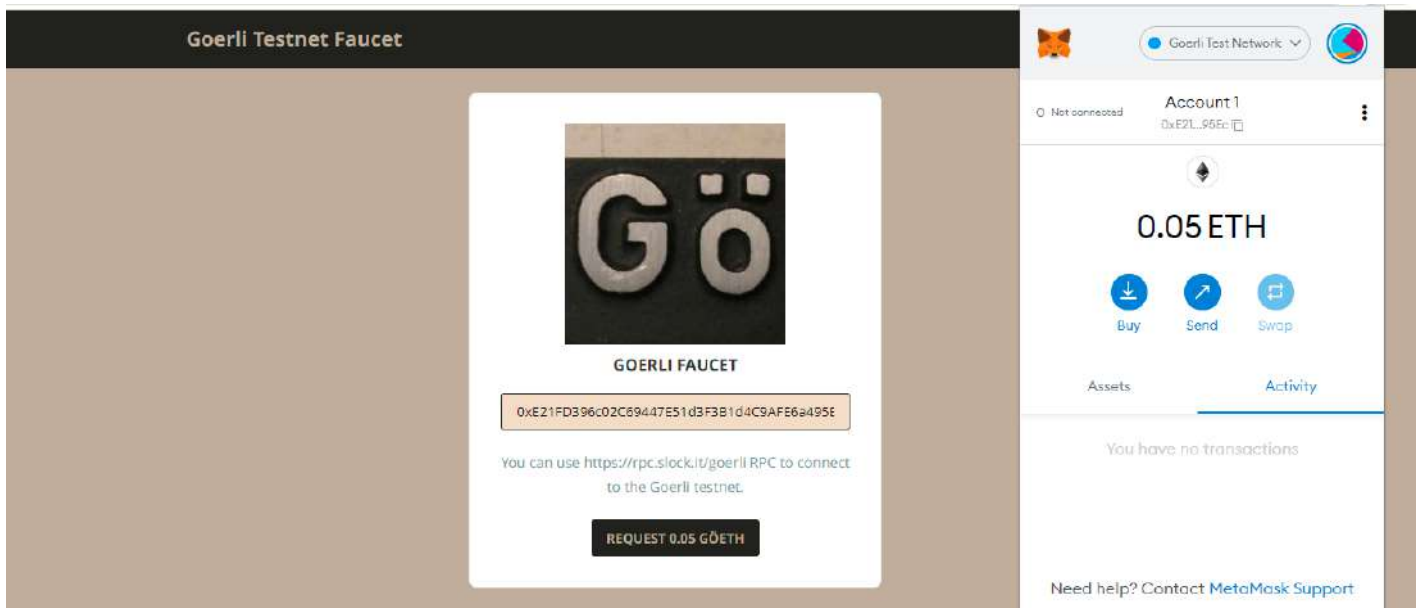
- 9 Firstly open your wallet and select ropston network .After that you click on buy ETH and then



click on get ether


9

→ AFTER ropston network ,open goerli network and click on buy button and after that you click on get ether




Practical No.5 Enable/Add the solidity compiler in Notepad++

Step-1: Download and Install Notepad++ .



Current Version 8.2

- Home
- Download
- News
- Online Help



Adobe Creative Cloud for Teams starting at \$33.99 per month. ADS VIA CARTON


Downloads

- Download Notepad++ v8.2
- Download Notepad++ v8.1.9.3
- Download Notepad++ v8.1.9.2
- Download Notepad++ v8.1.9.1
- Download Notepad++ v8.1.9
- Download Notepad++ v8.1.8

Practical Bitcoin BL...pdf Blockchain.py


22°C Light rain ENG 11:05 PM

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information you've provided to them or they've collected from your use of their services. [Got it!](#)




Current Version 8.2

- Home
- Download
- News
- Online Help
- Resources



Limited time offer: Get 10 free Adobe Stock images. ADS VIA CARTON

Download 64-Bit X64




DOWNLOAD

• [Installer](#) | [GPG Signature](#)

[e \(zip\)](#) | [GPG Signature](#)

SPONSORED



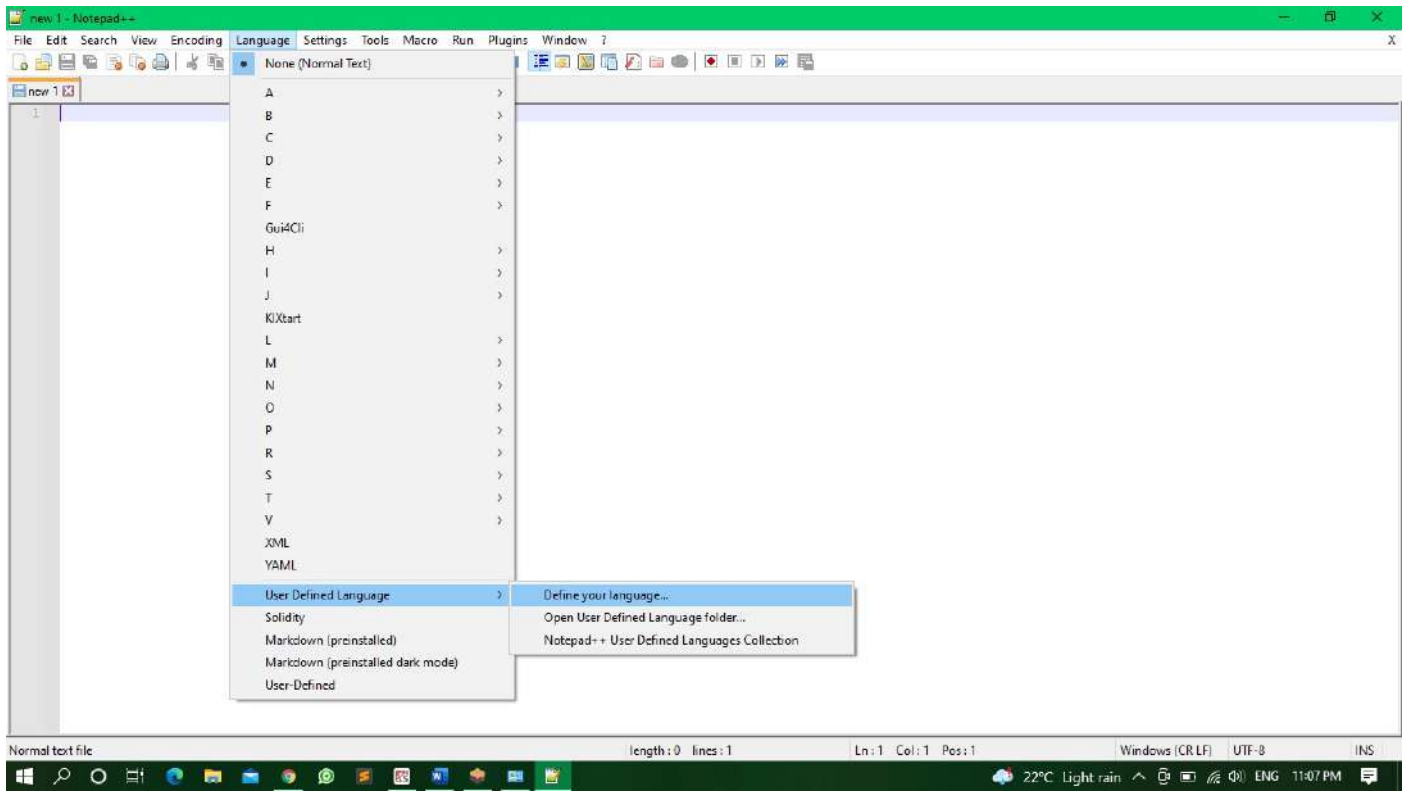
OVHcloud Your Alternative Cloud Provider in Australia with the Best Price-Performance ratio on Servers & Storage.

Practical Bitcoin BL...pdf Blockchain.py

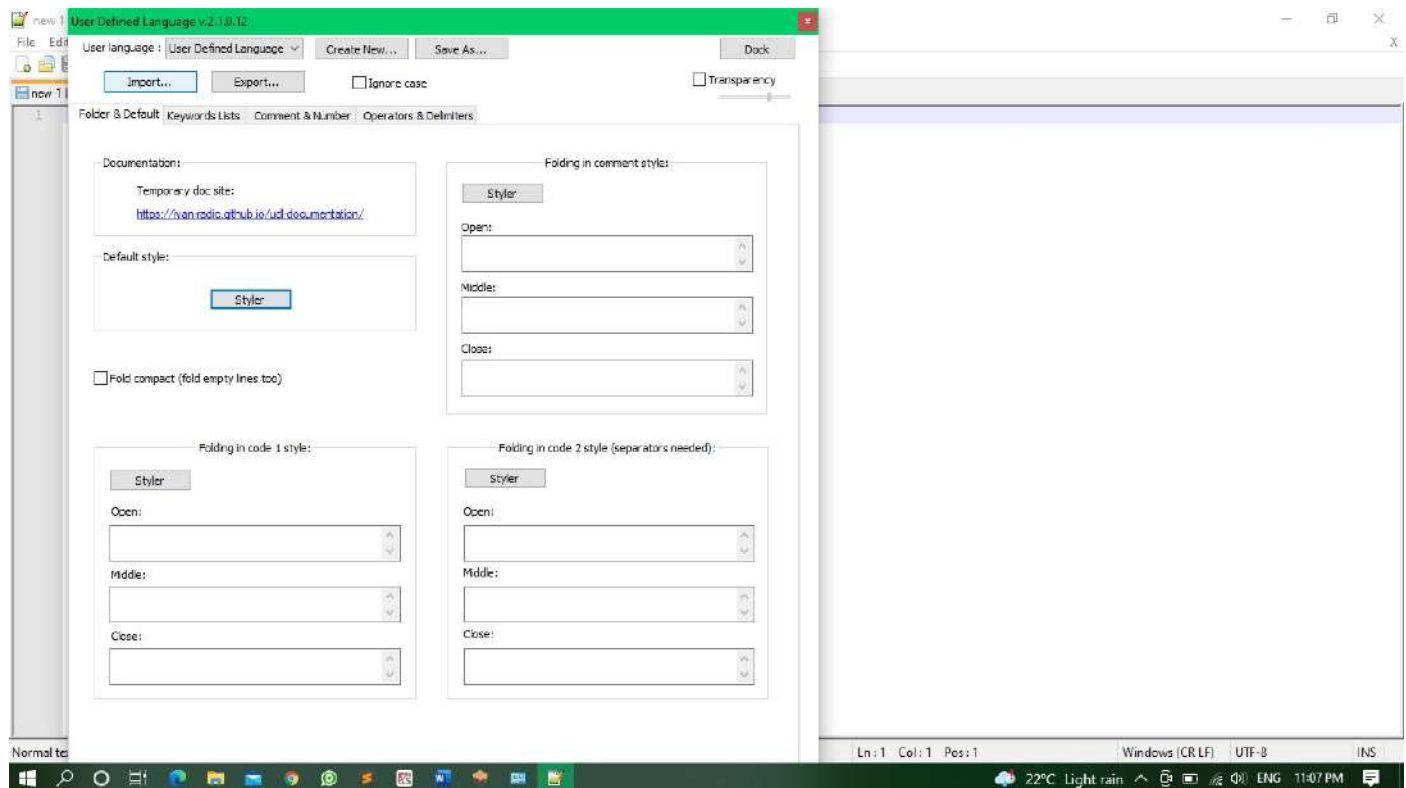
22°C Light rain ENG 11:05 PM

Step-2: Now open Notepad++

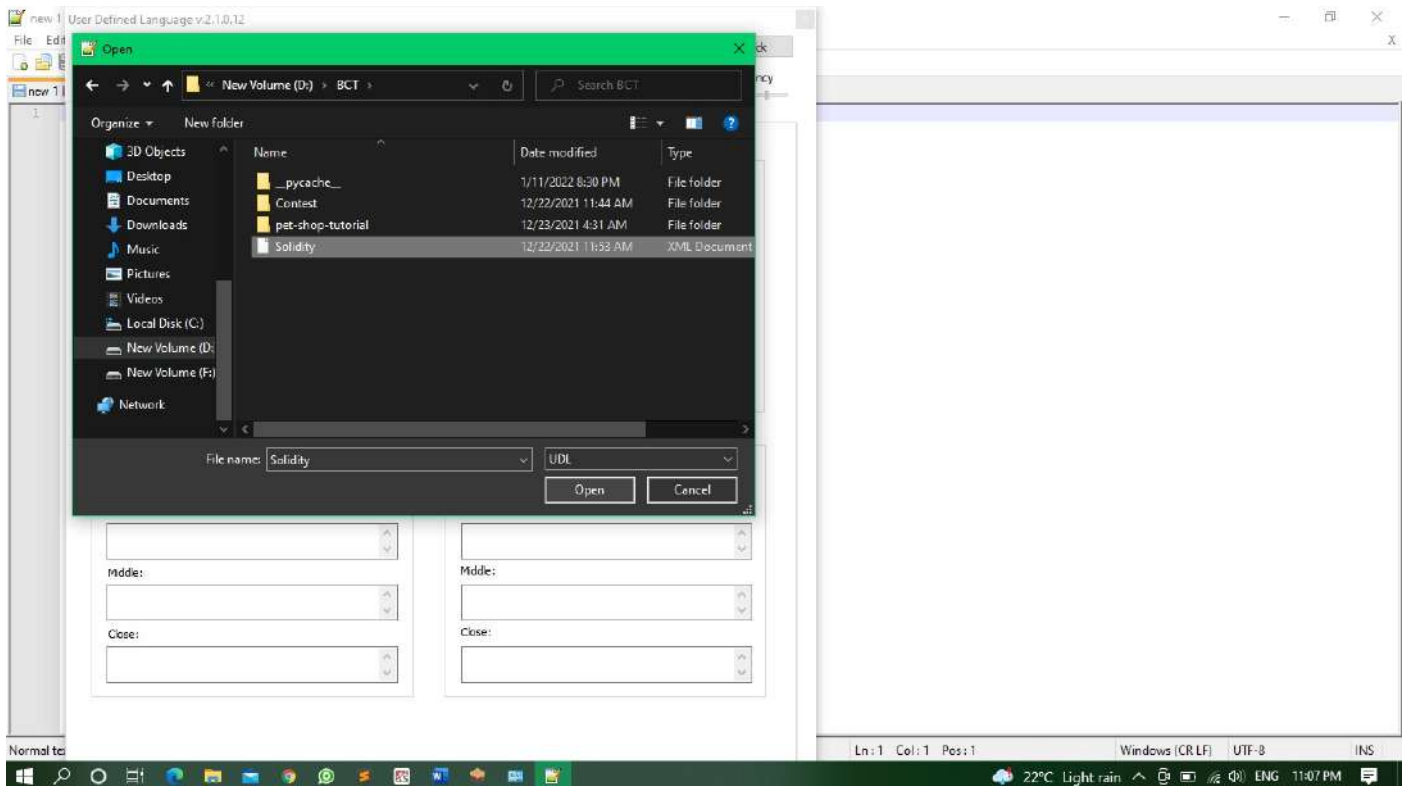
Step-3: Click on language >> User defined language >> define your language



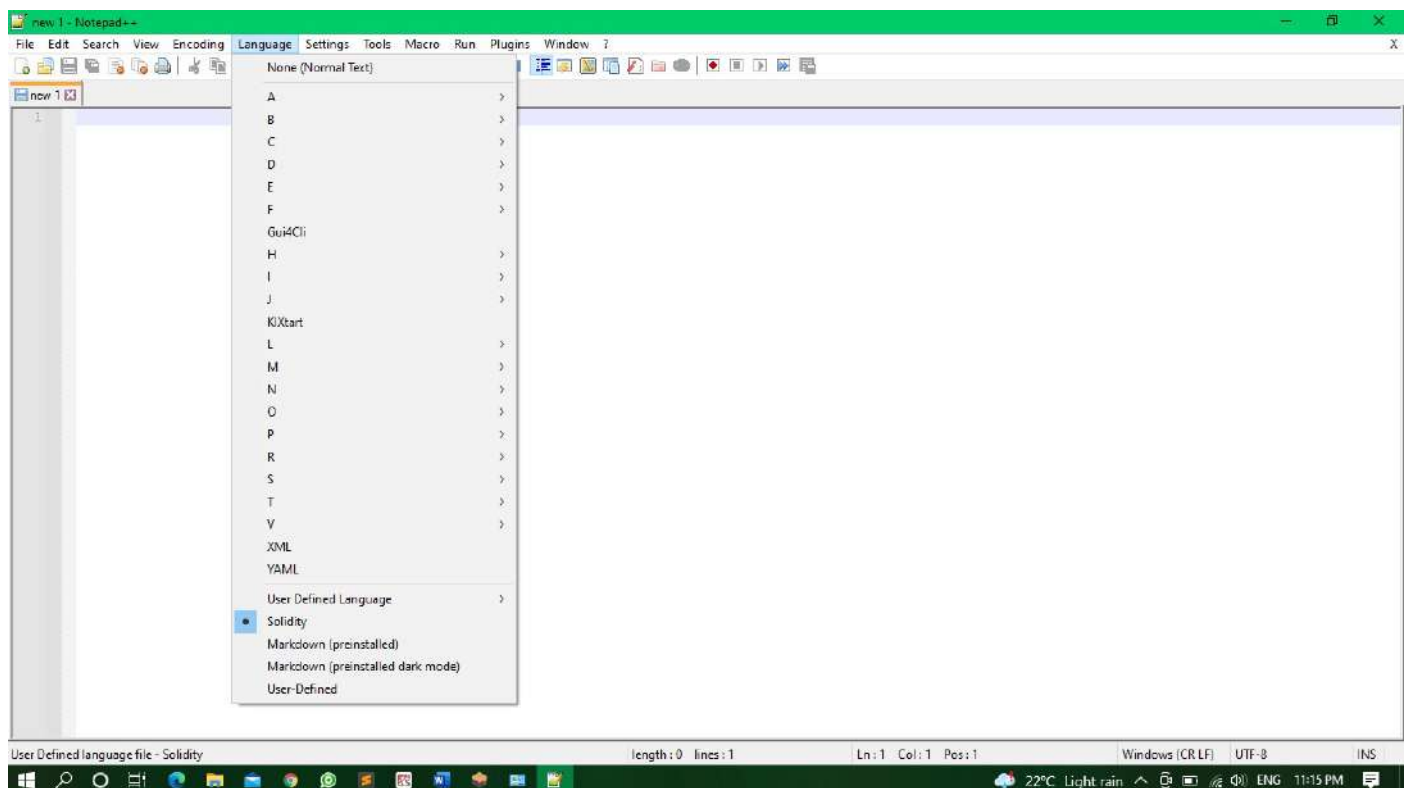
Step-4: Click on "Import"



Step-5: Click on BCT >> Solidity and open it

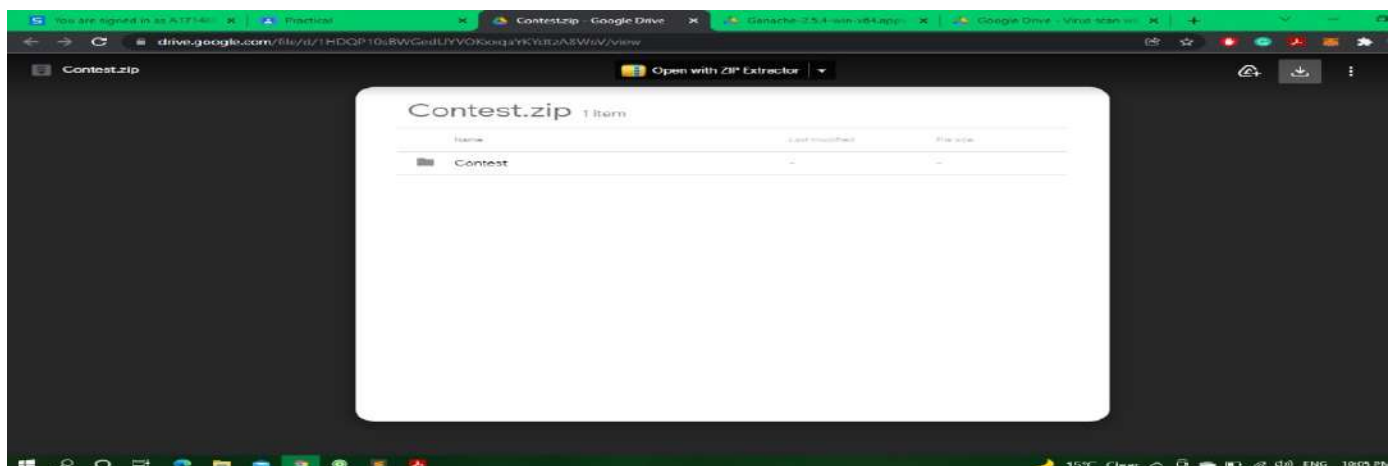
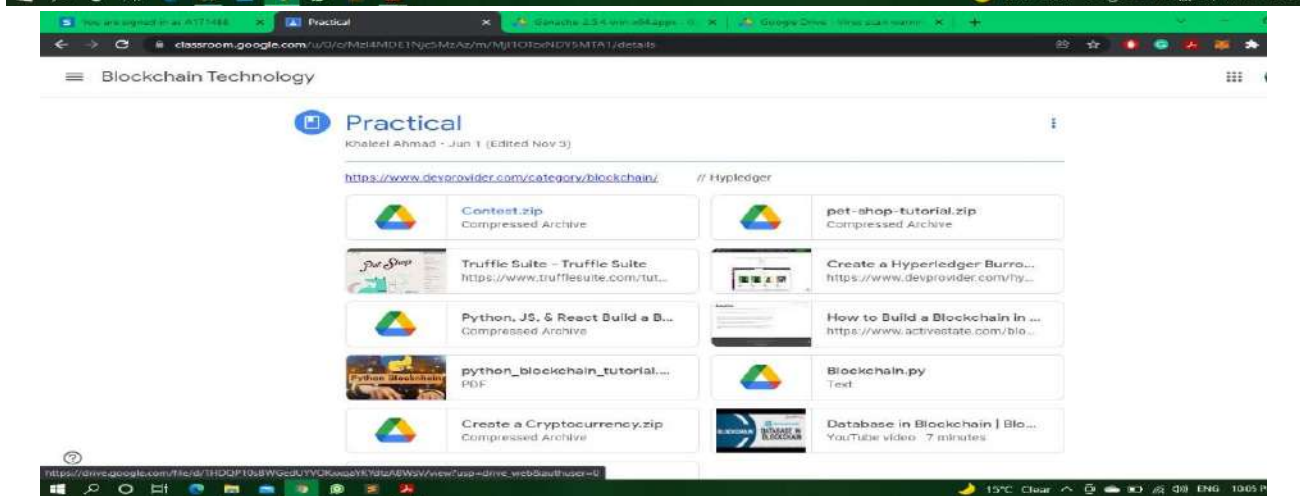
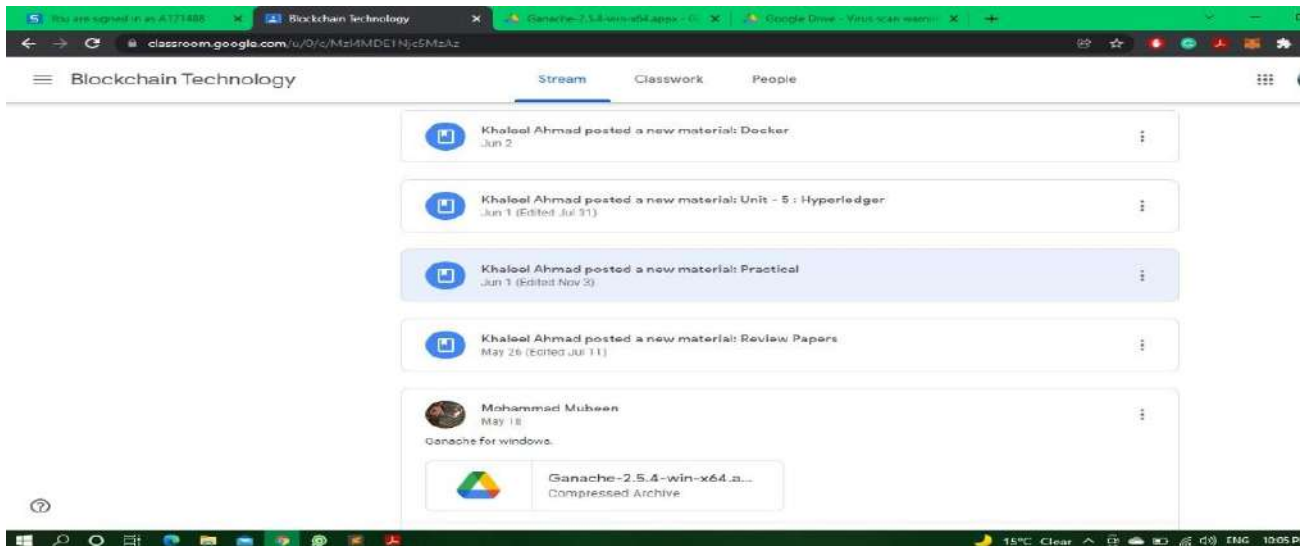


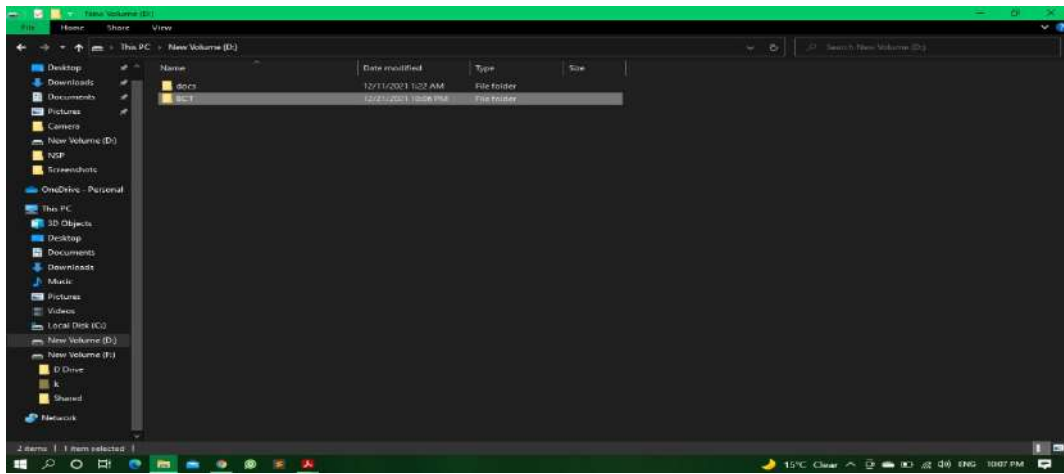
Step-6: Now you can see Solidity has been added in Notepad++.



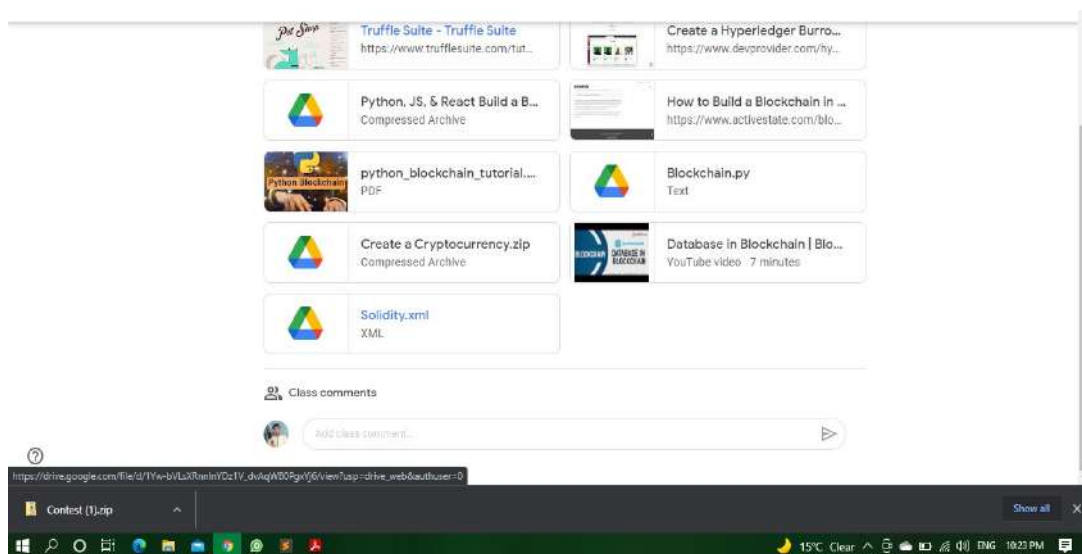
Practical No.6 E-Voting (Best Actor Contest)

Step-1: Open Google classroom of Blockchain Technology open Practical folder and download “Contest” file and extract it into a new folder called “BCT”.





Step-2: Download Notepad++, Solidity, NodeJS and Git from google classroom by clicking on truffle suite and add Solidity in Notepad++ .



Step-3: Open cmd and run “`npm install -g truffle`” command.

After installation simply copy the BCT folder path and open it in cmd and run following command one by one.

>>Truffle compile

>>Truffle migrate

>>Truffle test

>>Npm run dev

After this there will be open a Best Actor Contest web page .

Now open metamask and connect this web page to an account which you have already imported from Ganache.

And Caste your vote by using some Ethereum.

```
111 Receive updates in the future
npm WARN deprecated multicodec@2.1.3: This module has been superseded by the multiformats module
npm WARN deprecated multicodec@2.1.3: This module has been superseded by the multiformats module
npm WARN deprecated axios@0.20.0: Critical security vulnerability fixed in v0.21.1. For more information, see https://github.com/axios/axios/pull/3410
npm WARN deprecated graphql-tools@4.0.8: This package has been deprecated and now it only exports makeExecutableSchema. And it will no longer receive updates. We recommend you to migrate to scoped packages such as @graphql-tools/schema, @graphql-tools/utils and etc. Check out https://www.graphql-tools.com to learn what package you should use instead.
npm WARN deprecated graphql-tools@4.0.8: This package has been deprecated and now it only exports makeExecutableSchema. And it will no longer receive updates. We recommend you to migrate to scoped packages such as @graphql-tools/schema, @graphql-tools/utils and etc. Check out https://www.graphql-tools.com to learn what package you should use instead.
npm WARN deprecated graphql-tools@4.0.8: This package has been deprecated and now it only exports makeExecutableSchema. And it will no longer receive updates. We recommend you to migrate to scoped packages such as @graphql-tools/schema, @graphql-tools/utils and etc. Check out https://www.graphql-tools.com to learn what package you should use instead.
npm WARN deprecated core-js@2.6.12: core-js@3 is no longer maintained and not recommended for usage due to the number of issues. Because of the V8 engine whims, feature detection in old core-js versions could cause a slowdown up to 100x even if nothing is polyfilled. Please, upgrade your dependencies to the actual version of core-js.

added 26 packages, removed 112 packages, changed 946 packages, and audited 955 packages in 3m

67 packages are looking for funding
  run `npm fund` for details

56 vulnerabilities (7 low, 33 moderate, 9 high, 7 critical)
To address issues that do not require attention, run:
  npm audit fix
To address all issues possible, run:
  npm audit fix --force
Some issues need review, and may require choosing
a different dependency.

Run `npm audit` for details.
npm notice New minor version of npm available! 6.12.2 -> 6.3.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v6.3.0
npm notice Run npm install -g npm@6.3.0 to update!
npm notice

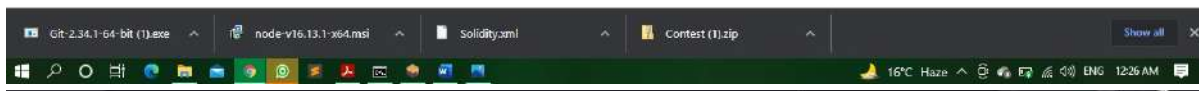
C:\Users\Qasim>
C:\Users\Qasim>
```

Best Actor Contest

Your Account: null

#	Name	Votes
1	Omy	0
2	Jammy	0

Select Contestant



Best Actor Contest

Your Account: 0xc52b307b455f94d5a2a63962611f1cda76fcada7

#	Name	Votes
1	Omy	0
2	Jammy	0

Select Contestant

Local

Account 2
0xc52b307b455f94d5a2a63962611f1cda76fcada7

99.9715 ETH

Connected sites

Account 2 is not connected to any sites.

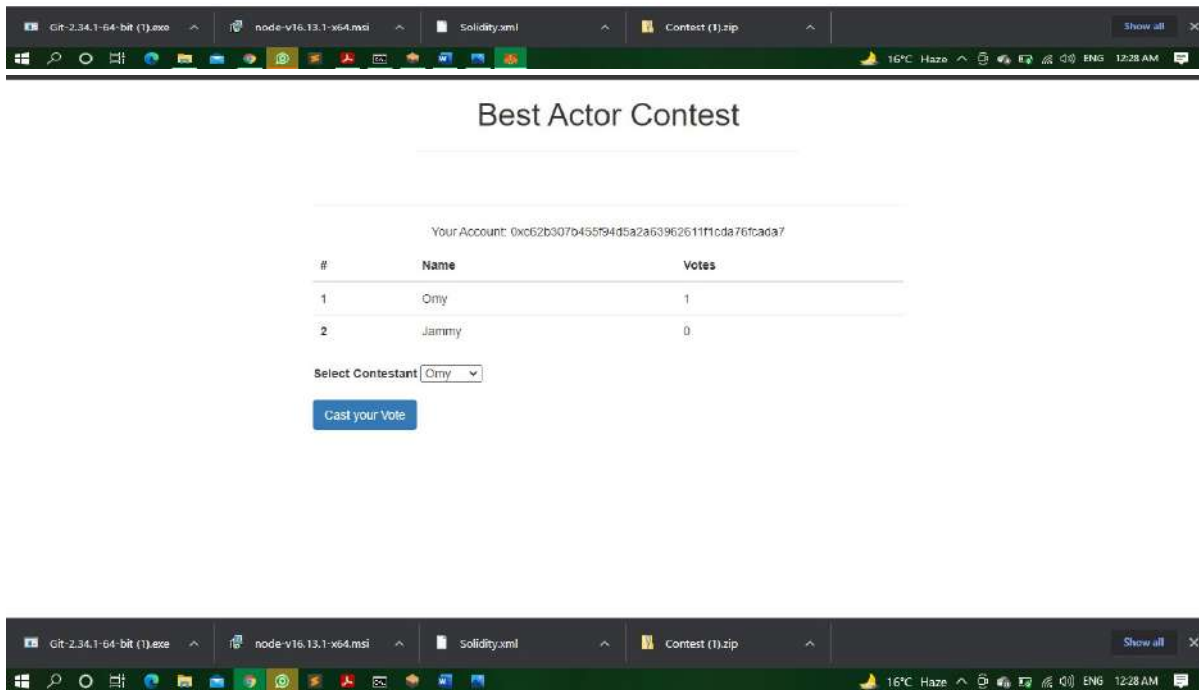
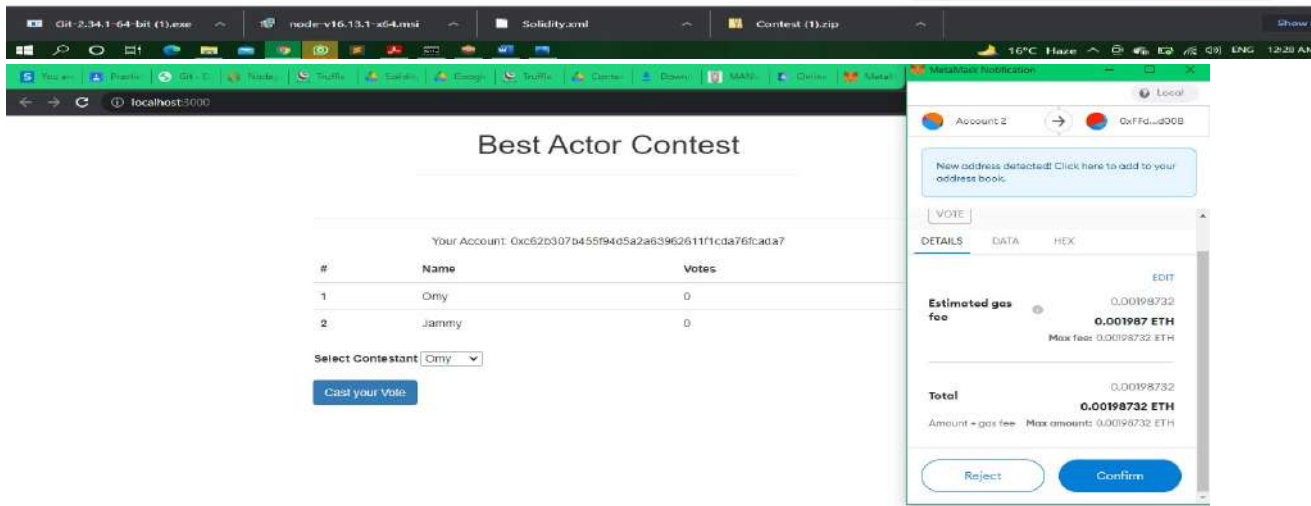
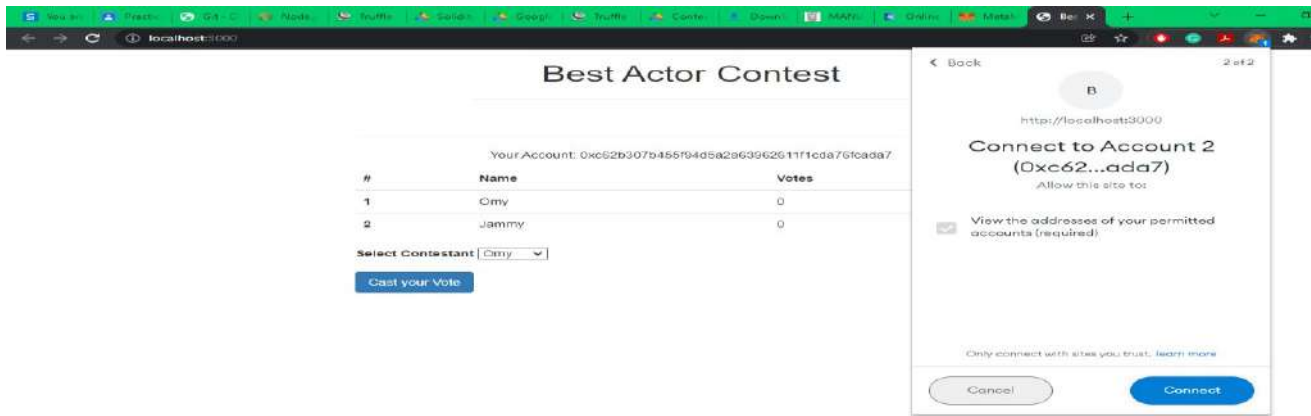
[Manually connect to current site](#)

99.9715 ETH

Don't see your token?
[Import tokens](#)

Need help? [Contact MetaMask Support](#)





Your E-voting has been done Successfully.

Practical No.7

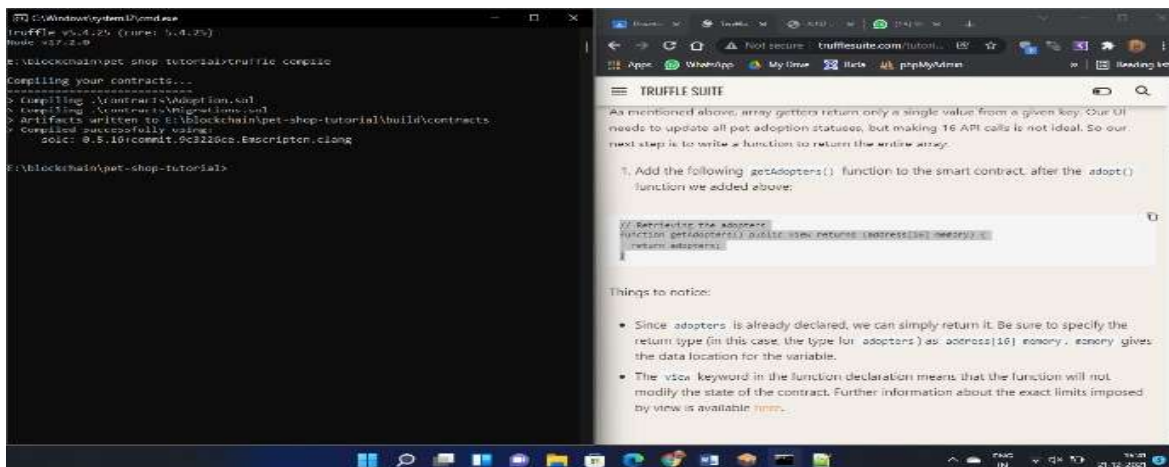
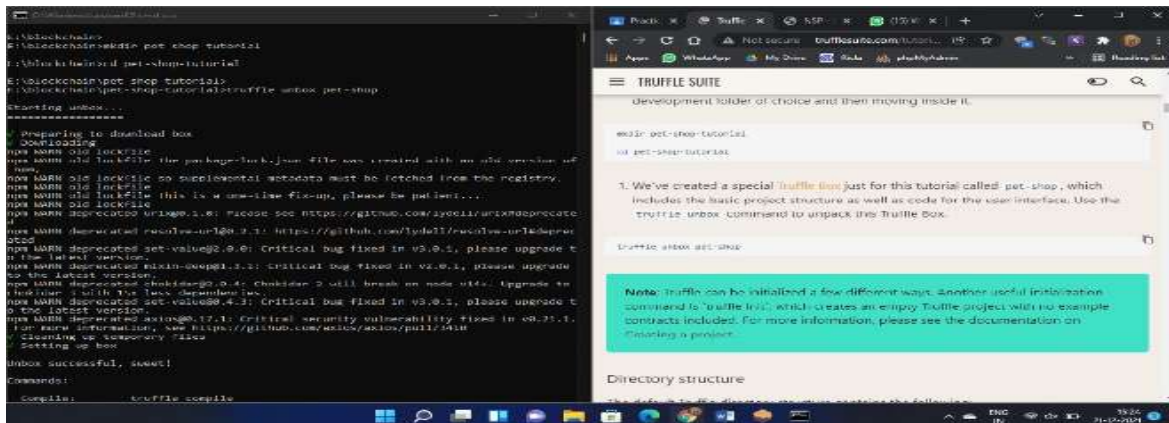
Pet-shop Tutorial

Step-1: Open cmd and run the following commands one by one,

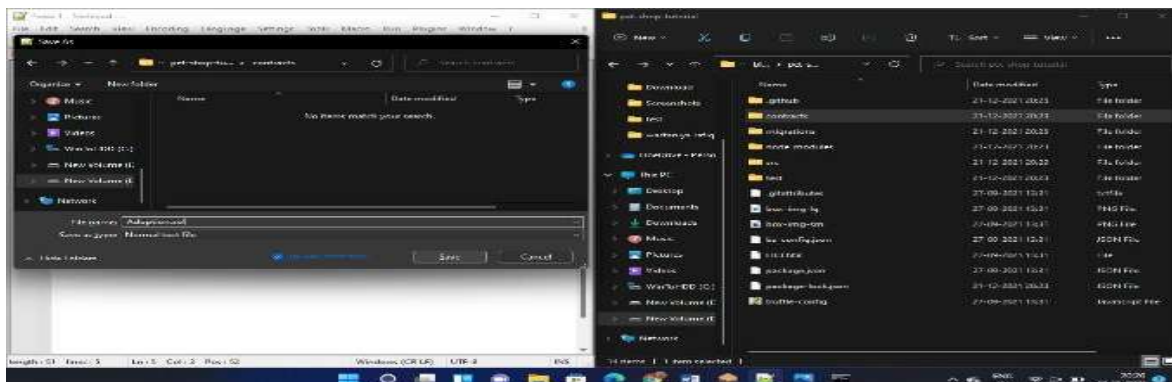
>> mkdir pet-shop-tutorial

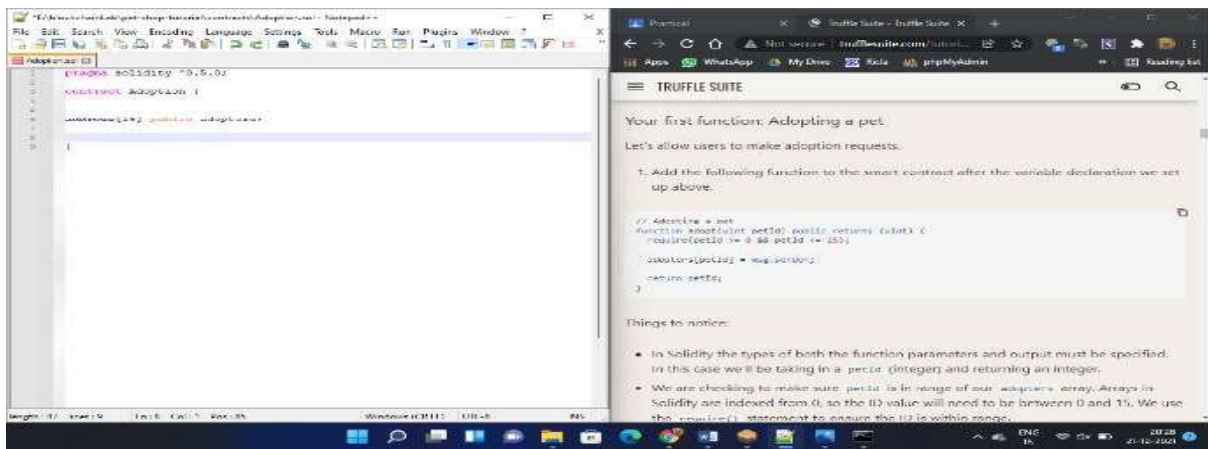
>> cd pet-shop-tutorial

>> truffle unbox pet-shop

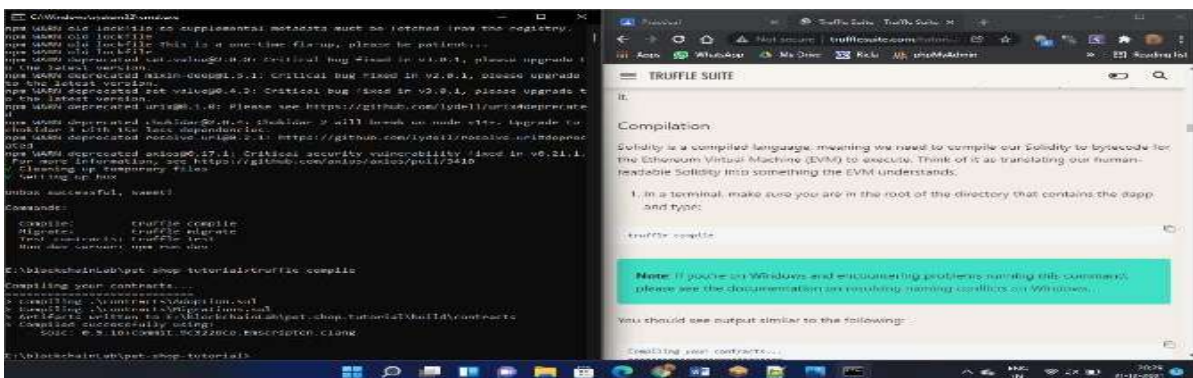
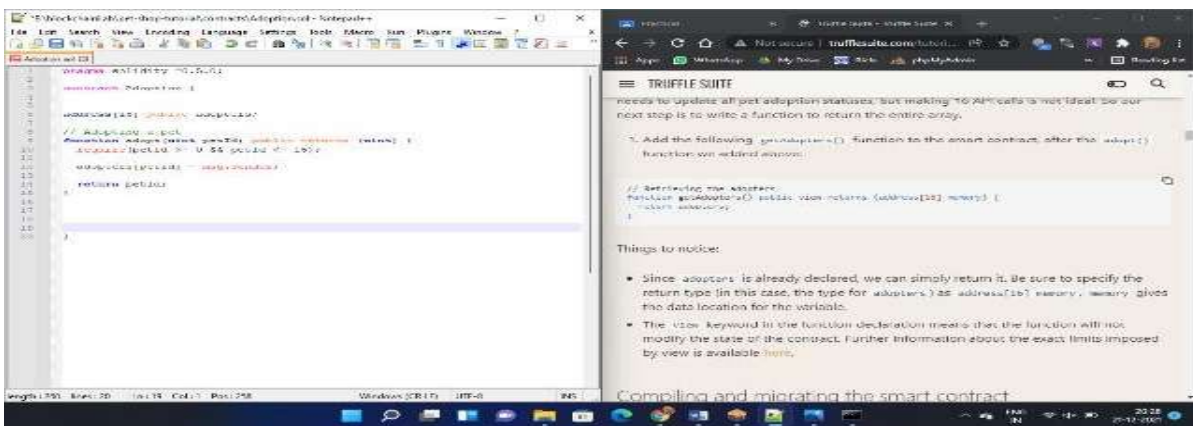


Step-2: Create a new file named Adoption.sol in the contracts/ directory, and add the following files.

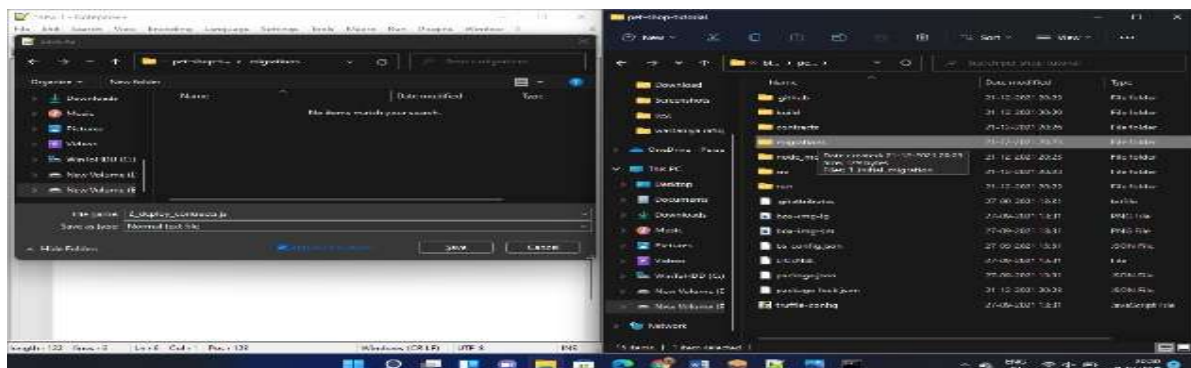


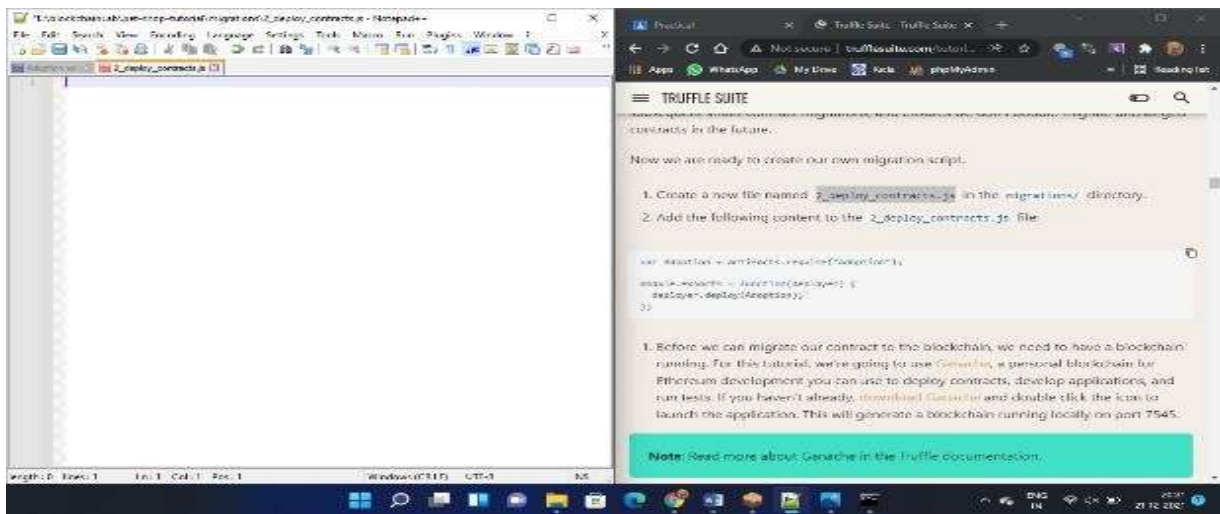


Step-3: Copy code and paste and run this command in
cmd >> truffle compile



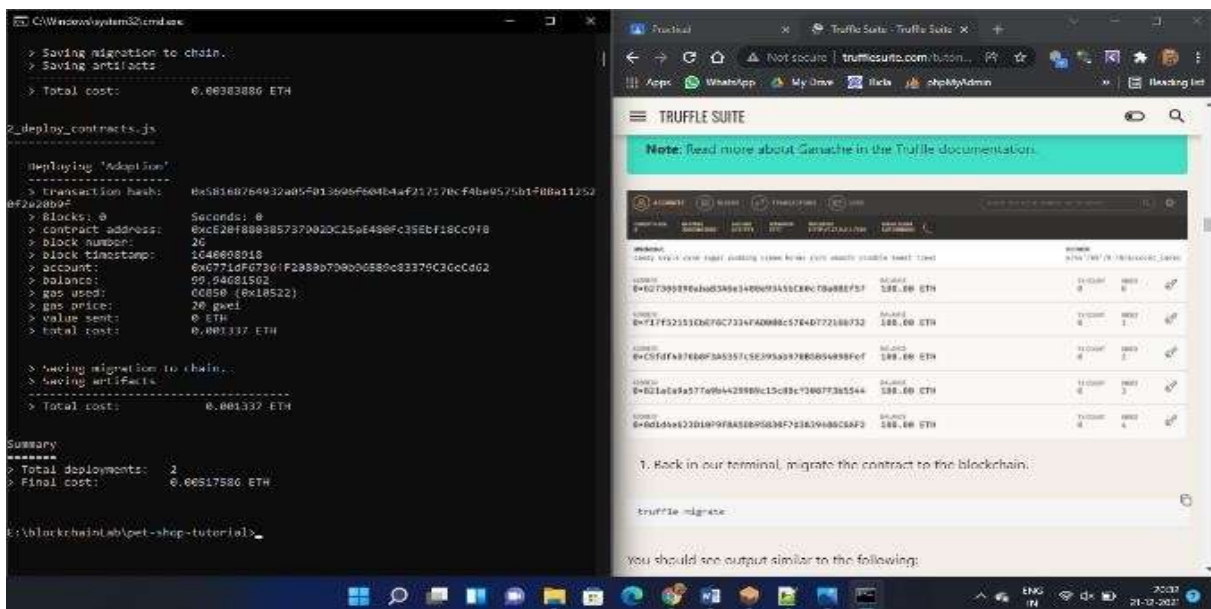
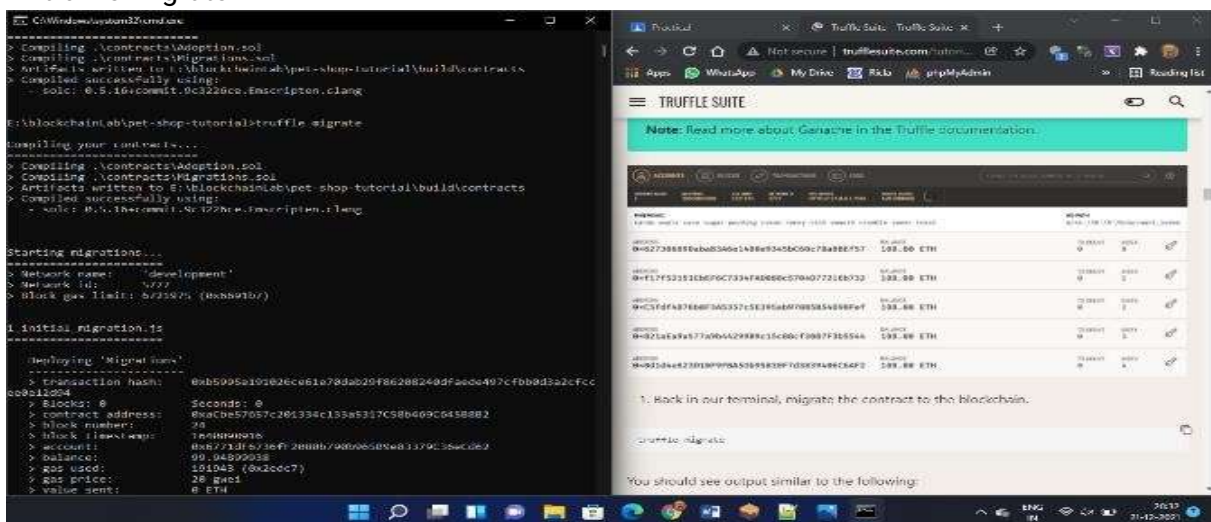
Step-4: Create a new file named `2_deploy_contracts.js` in `migrations/` directory and write the codes.



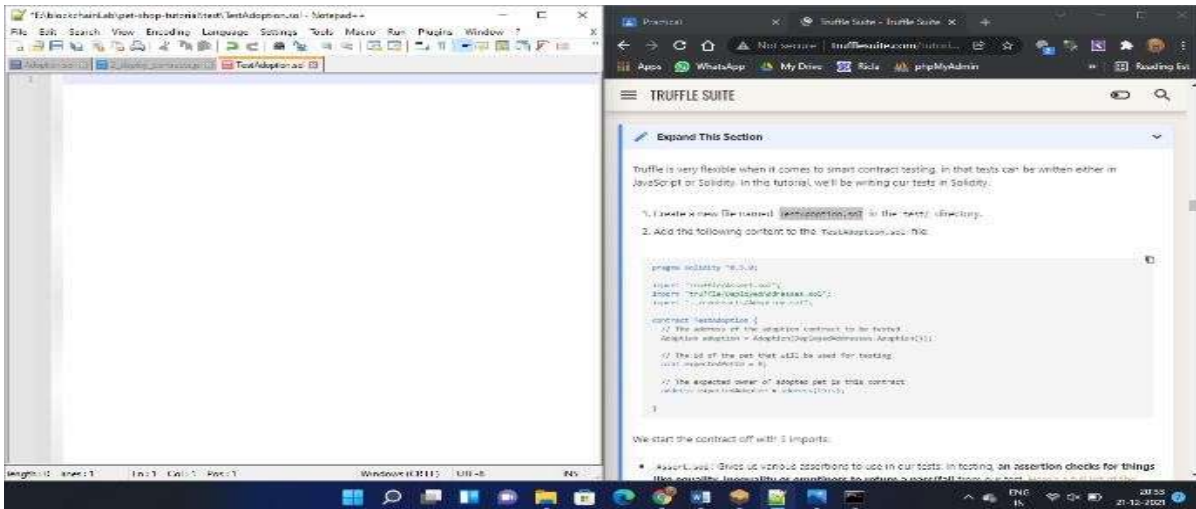
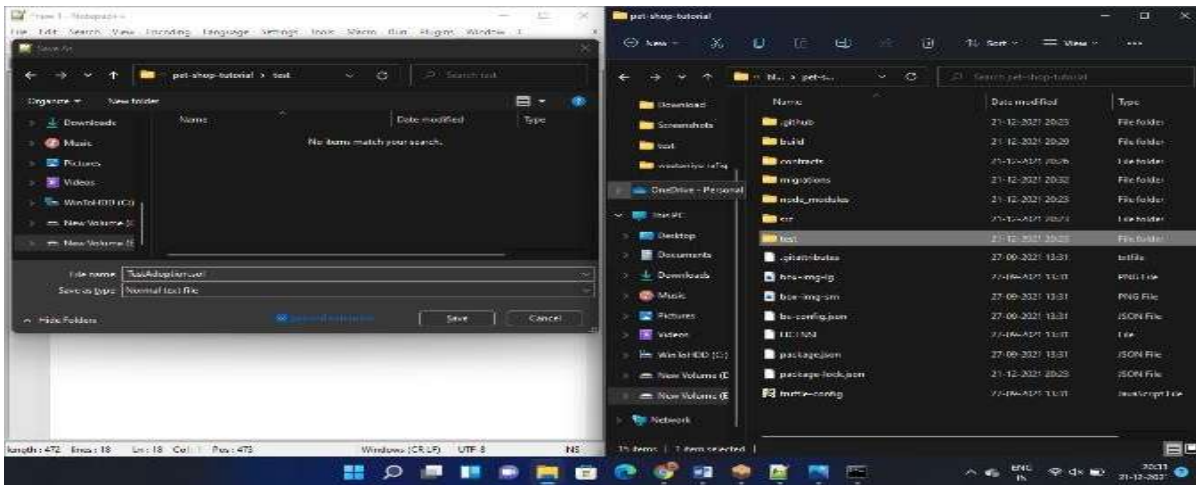


Step-5: Run this command in cmd-

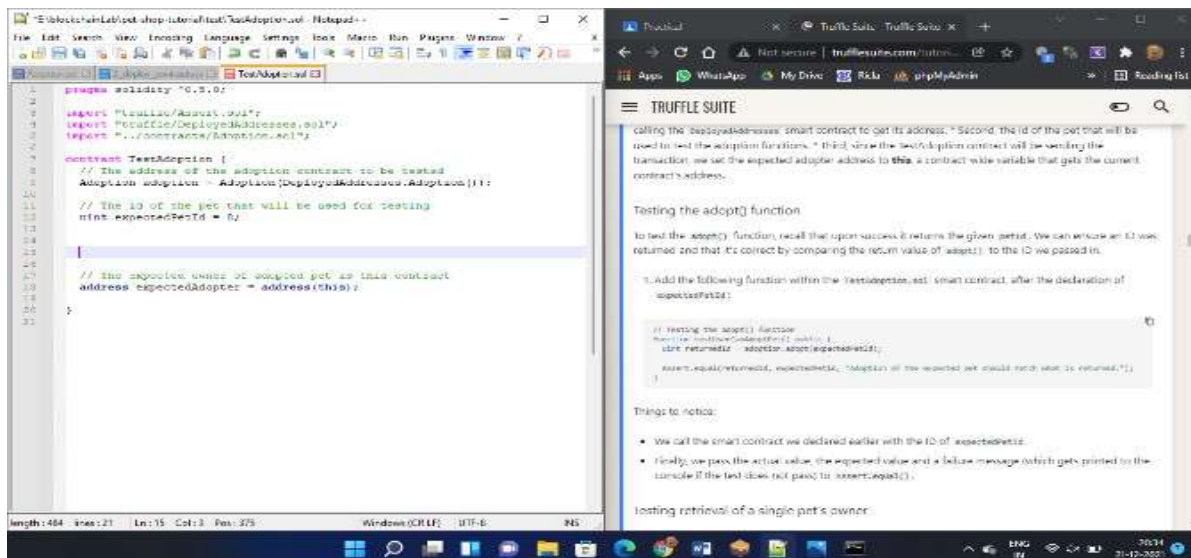
>> truffle migrate

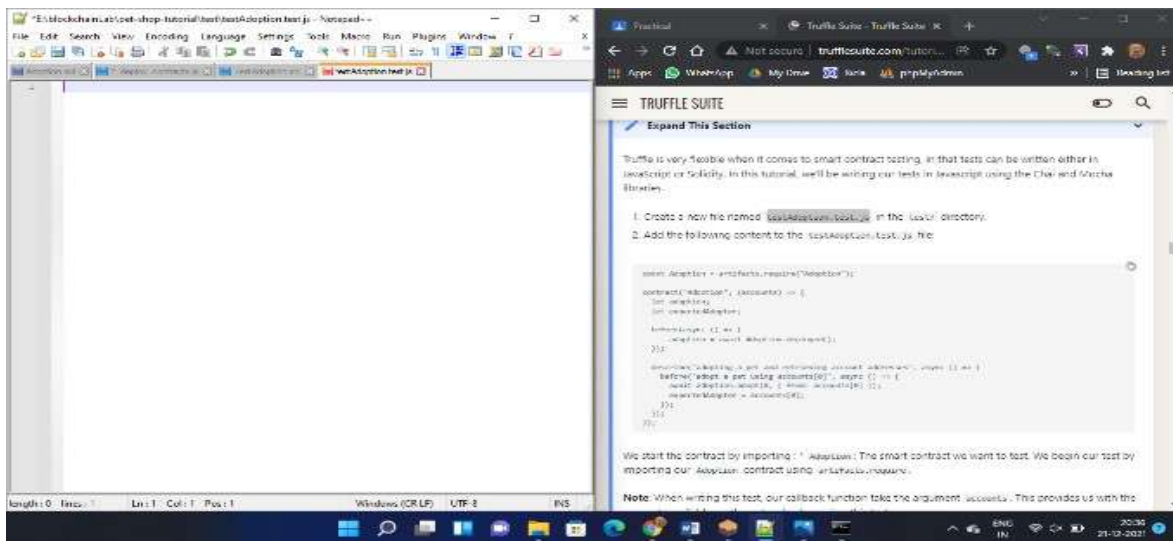


Step-6: Create a new file named TestAdoption.sol in test/ directory and write the code

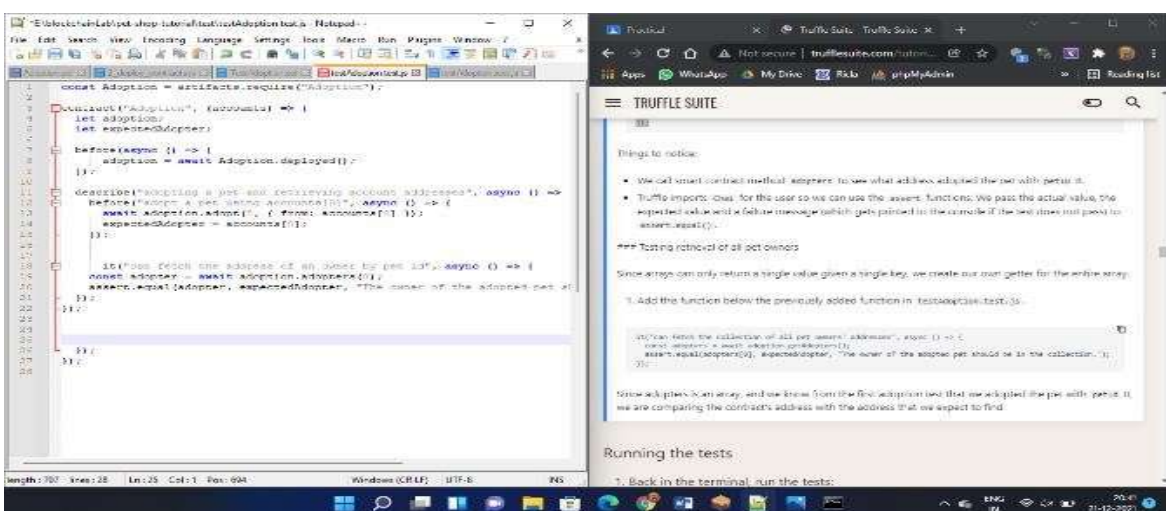
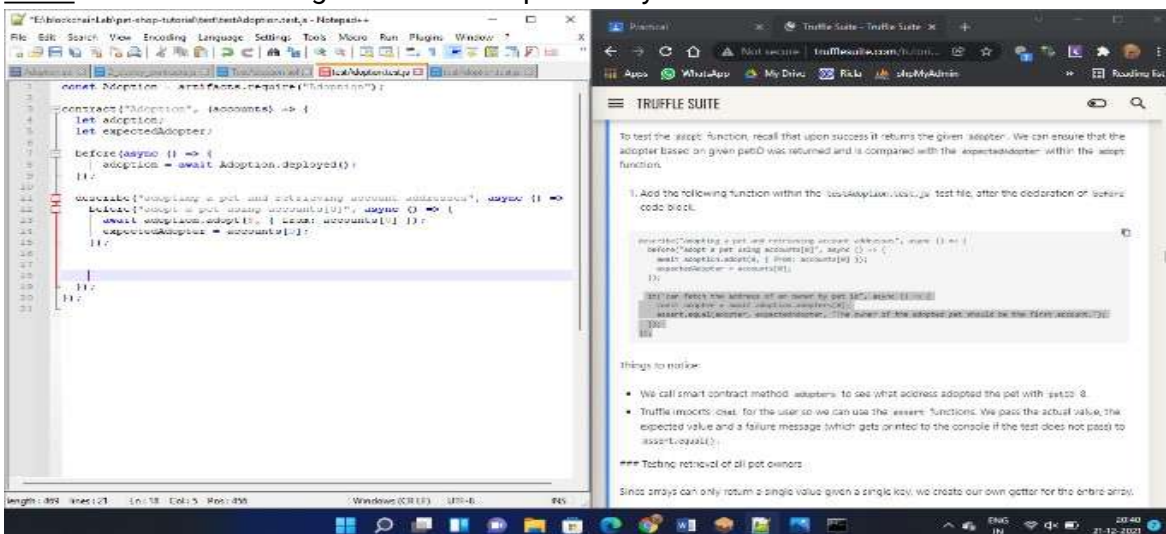


Step7: Add the following function within the TestAdoption after declaration of expectedId and following function previously added function



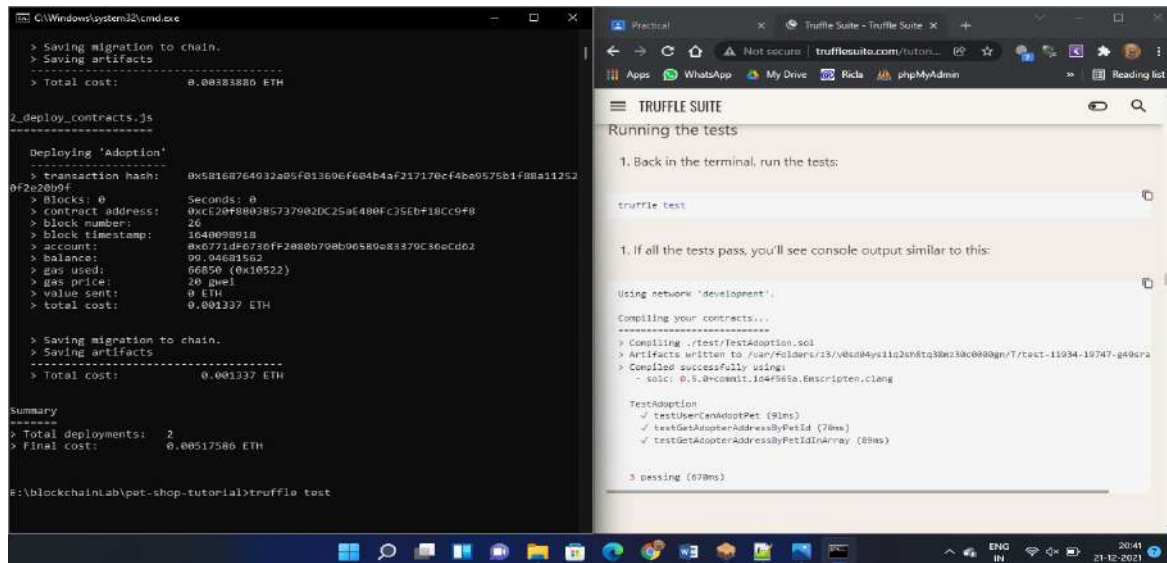


Step-9: Add the following code within the `testAdoption.test.js` test file after the declaration of before code block and add following code after the previously added code.



Step-10: Run this command in cmd-

>> truffle test



The terminal window shows the execution of 'truffle test' in a Windows command prompt. It displays the deployment of a contract named 'Adoption' and the execution of tests. The Truffle Suite web interface shows the test results, indicating that all tests passed.

```
> Saving migration to chain.
> Saving artifacts
> Total cost: 0.00383880 ETH

2_deploy_contracts.js

Deploying 'Adoption'
-----
> transaction hash: 0x58168764032a05f013696f604b4af217170cf4be9575b1f8a112520f2e20b0f
> Blocks: 0
> contract address: 0xc120f88093857379020c25a6480fc35ebf10cc9f8
> block number: 26
> block timestamp: 1640098918
> account: 0x0771d0730ff2880b70ab96580a83470c36acd02
> balance: 90.06081562
> gas used: 66850 (0x10522)
> gas price: 20 gwei
> value sent: 0 ETH
> total cost: 0.001337 ETH

> Saving migration to chain.
> Saving artifacts
> Total cost: 0.001337 ETH

Summary
-----
> Total deployments: 2
> Final cost: 0.00517580 ETH

E:\blockchain\pat-shop-tutorial>truffle test
```

TRUFFLE SUITE

Running the tests

- Back in the terminal, run the tests:

```
truffle test
```

If all the tests pass, you'll see console output similar to this:

Using network 'development'.

Compiling your contracts...

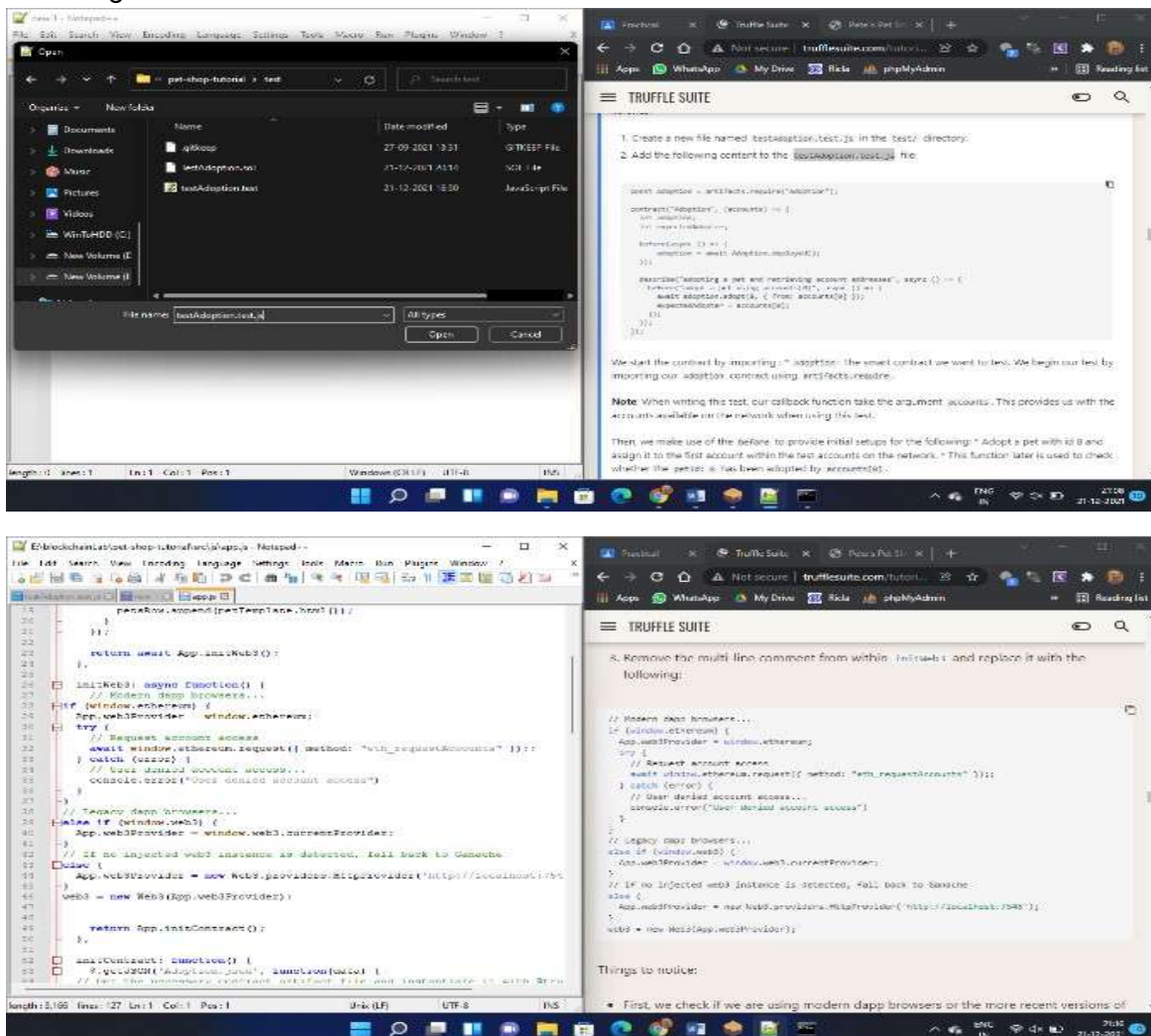
```
> Compiling .\test\testAdoption.sol
> Artifacts written to .\artifacts
> Compiled successfully using:
  - solc: 0.5.0+commit.104f555a.Emscripten.clang
```

TestAdoption

- ✓ testUserCanAdoptPet (91ms)
- ✓ testGetAdopterAddressByPetId (78ms)
- ✓ testGetAdopterAddressByPetIdInArray (89ms)

3 passing (678ms)

Step 11: Open /src/js/app.js and remove multiline comment from within initWeb3 and replace it with the following code



The Notepad++ editor shows the code in app.js. The Truffle Suite web interface shows the updated code, which includes a multi-line comment and a request for account access.

```
const { ethers } = require('ethers');
const { web3 } = require('web3');

// Modern dapp browsers...
if (window.ethereum) {
  // Request account access
  window.ethereum.request({ method: 'eth_requestAccounts' });
} else {
  // Legacy dapp browsers...
  const web3Provider = new Web3.providers.HttpProvider('http://localhost:7545');
  web3 = new Web3(web3Provider);
}

// If no injected web3 instance is detected, fall back to Ganache
const web3Provider = window.ethereum ? window.ethereum.request({ method: 'eth_requestAccounts' }) :
  new Web3.providers.HttpProvider('http://localhost:7545');

web3 = new Web3(web3Provider);

return App.initContract();
```

TRUFFLE SUITE

Remove the multi line comment from within initWeb3 and replace it with the following:

```
// Modern dapp browsers...
if (window.ethereum) {
  // Request account access
  window.ethereum.request({ method: 'eth_requestAccounts' });
} else {
  // Legacy dapp browsers...
  const web3Provider = new Web3.providers.HttpProvider('http://localhost:7545');
  web3 = new Web3(web3Provider);
}

// If no injected web3 instance is detected, fall back to Ganache
const web3Provider = window.ethereum ? window.ethereum.request({ method: 'eth_requestAccounts' }) :
  new Web3.providers.HttpProvider('http://localhost:7545');

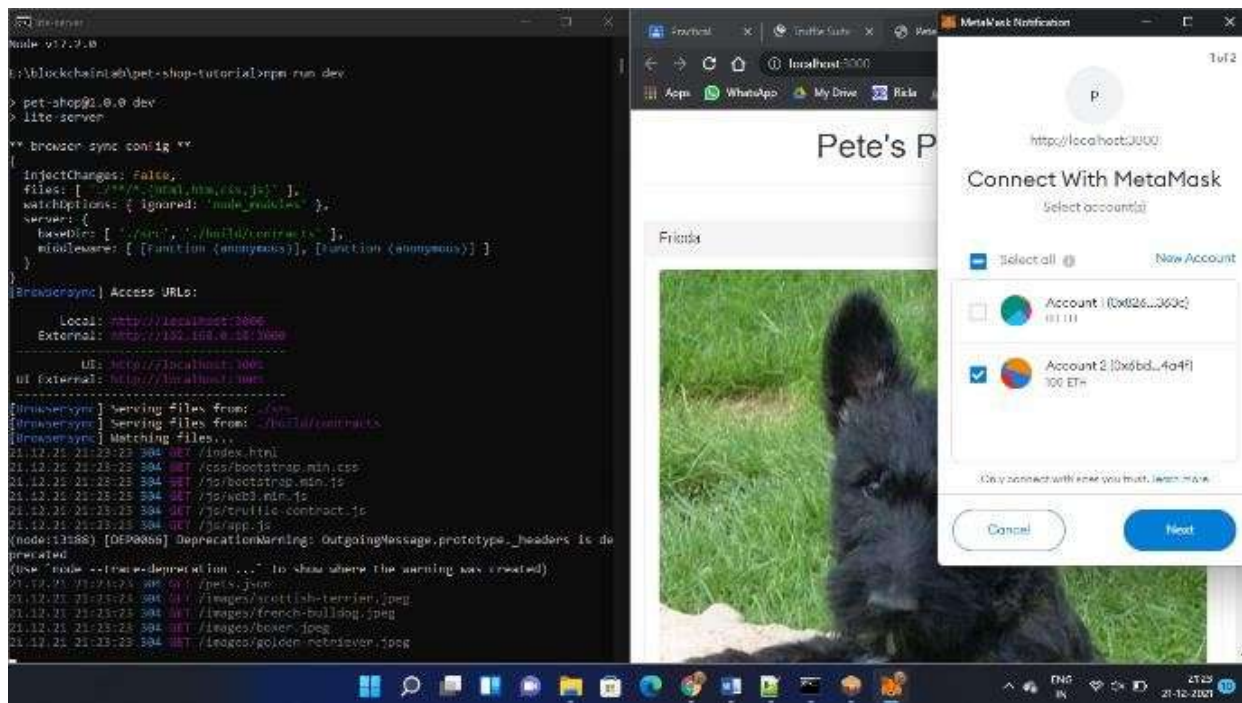
web3 = new Web3(web3Provider);

return App.initContract();
```

Things to notice:

- First, we check if we are using modern dapp browsers or the more recent versions of

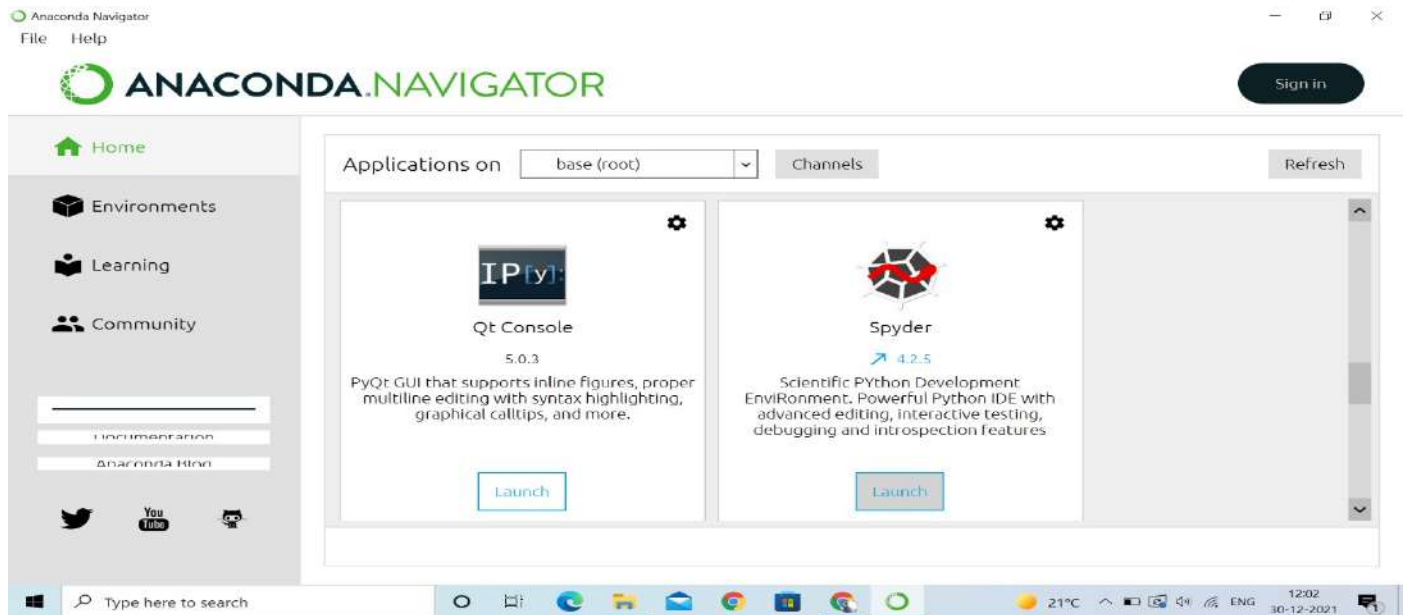
Step 12: Remove multiline comment from within initContract and replace it and markAdopted and replace



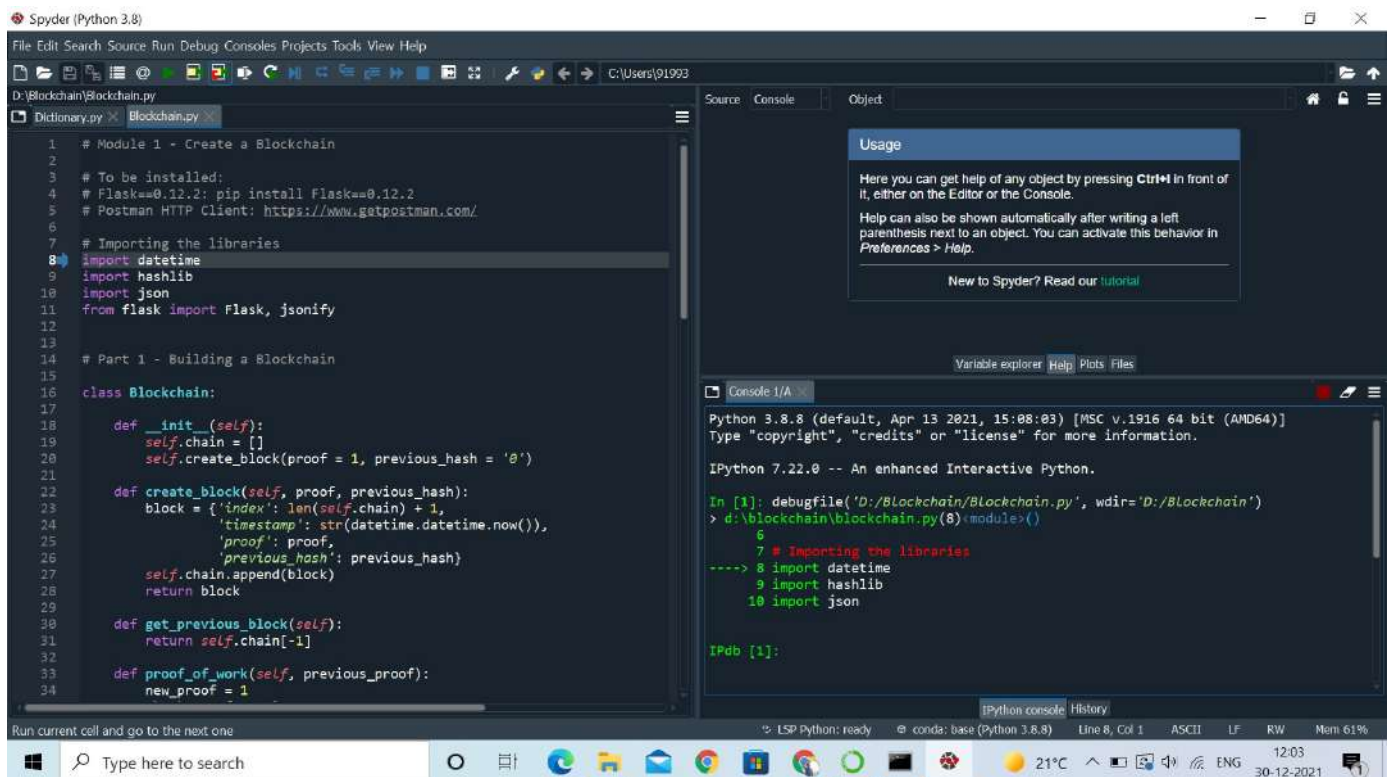
Practical No.8

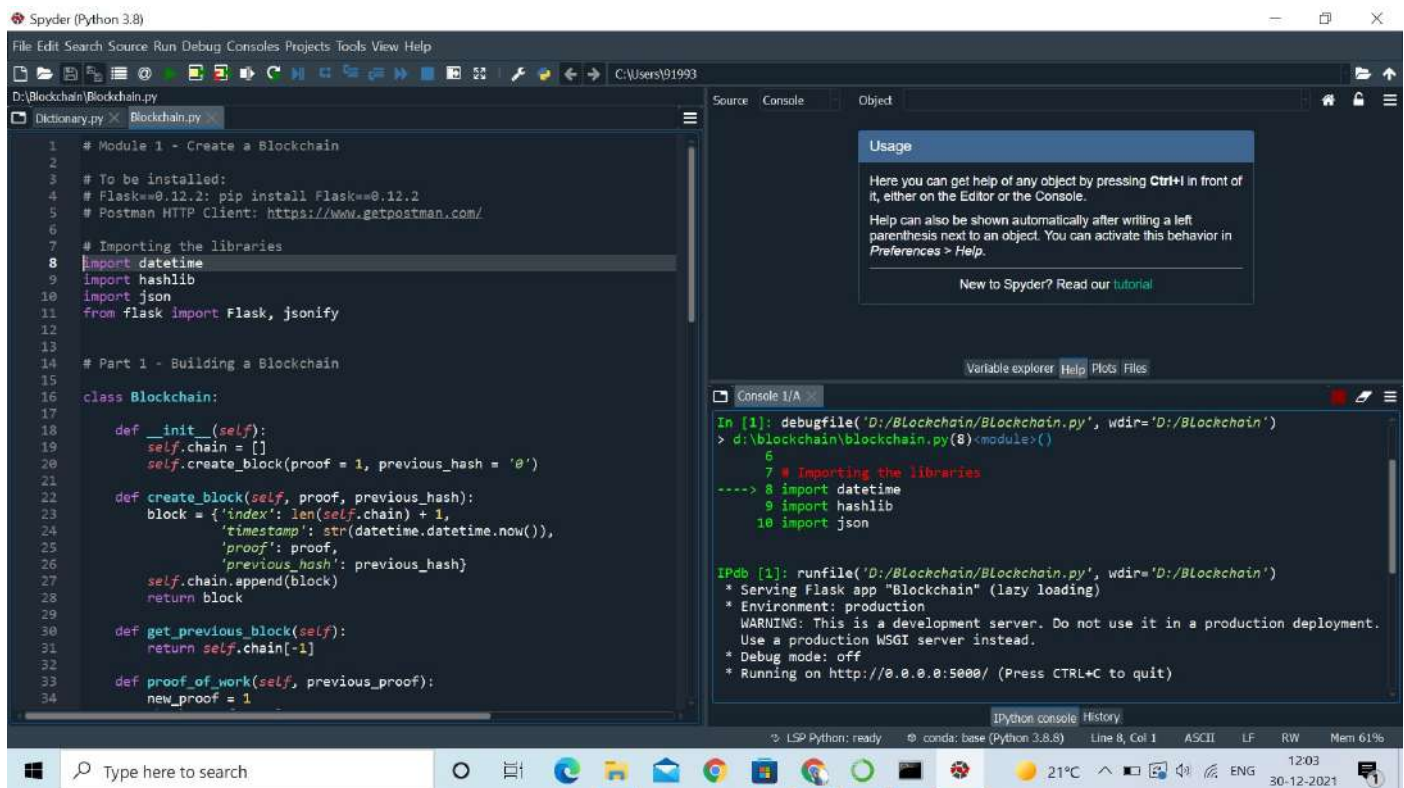
Create the Blockchain using Python, Flask micro web framework and Postman.

Step 1 : Open Anaconda Navigator and Open Spyder.

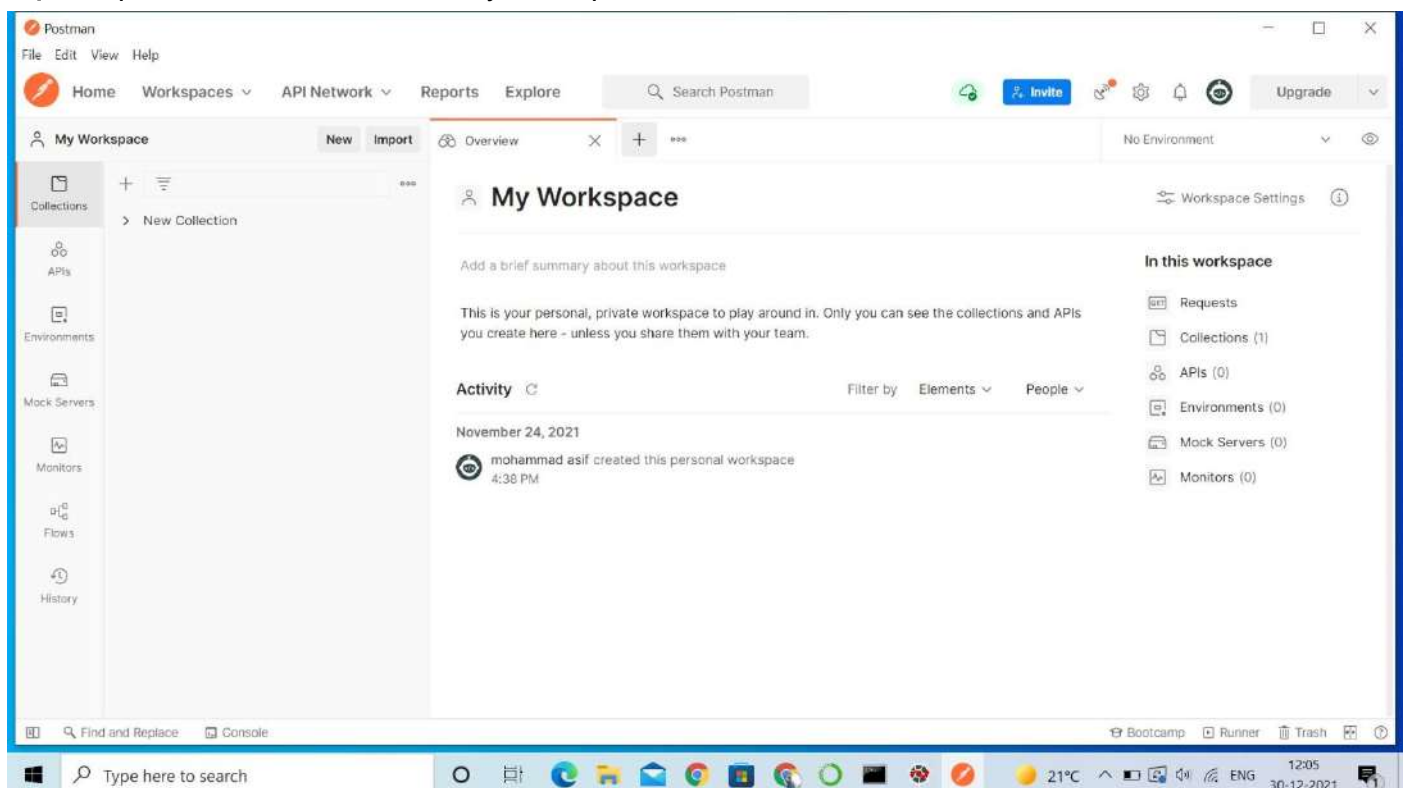


Step 2 : Now open "Blockchain.py" file in spyder >> debug it and run it

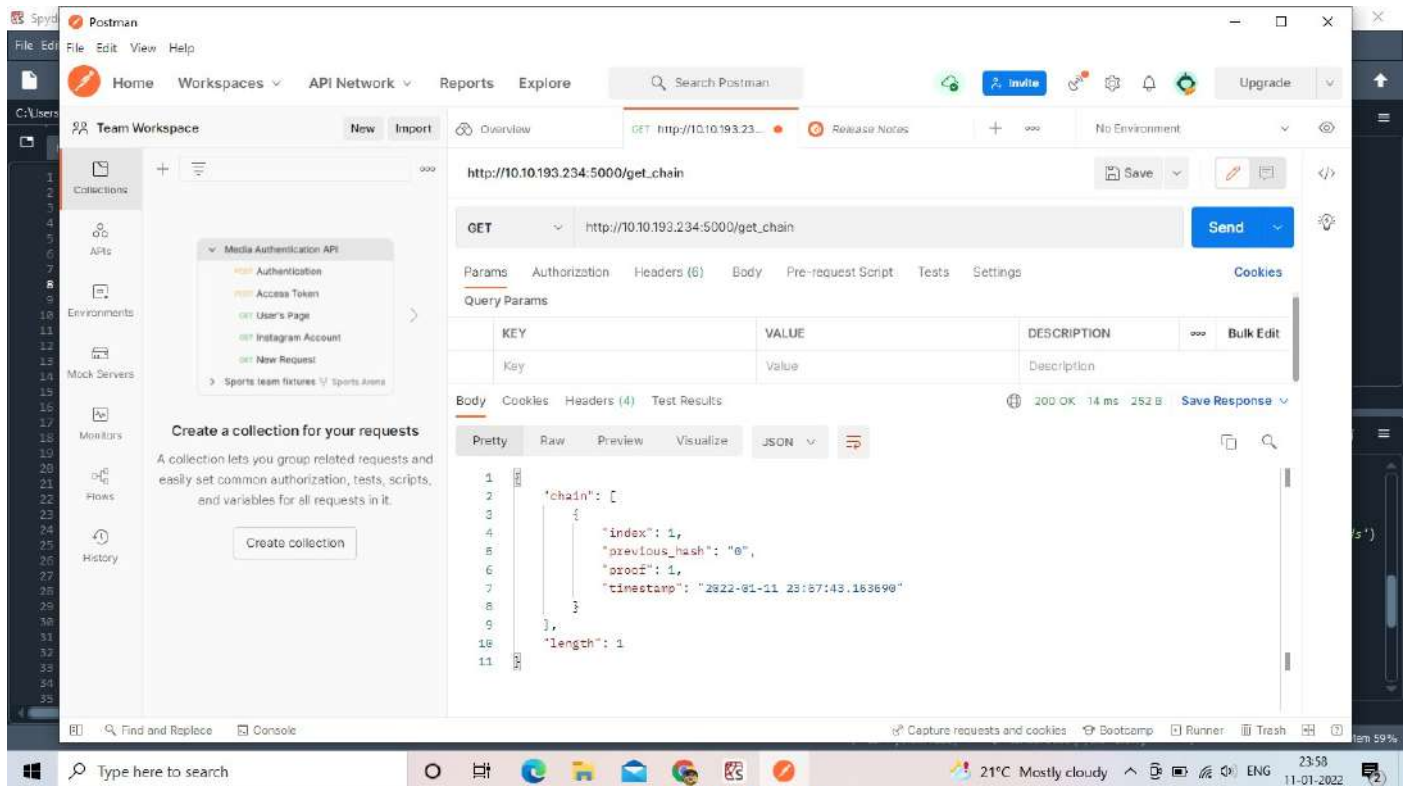




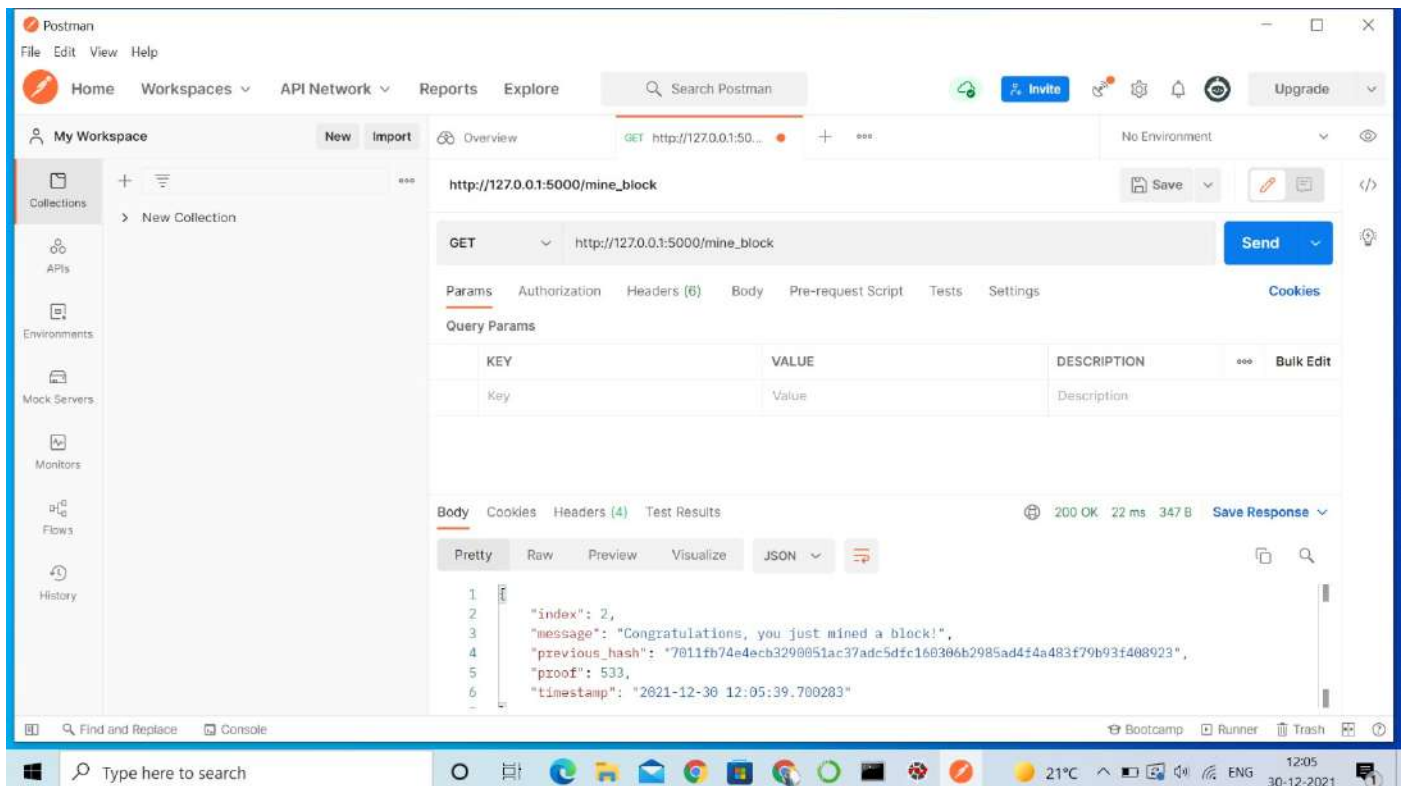
Step 3 : Open Postman and click on My Workspace.



Step 4 : Now write "http://127.0.0.1:5000/get_chain" in GET field and click on Send.



Step 5 : Now write "http://127.0.0.1:5000/mine_block" in GET field and click on Send and repeat it for three four times.



Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:5000/mine_block No Environment

Collections + New Collection

APIs

Environments

Mock Servers

Monitors

Flows

History

http://127.0.0.1:5000/mine_block

GET http://127.0.0.1:5000/mine_block Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (4) Test Results 200 OK 131 ms 349 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "index": 3,
3   "message": "Congratulations, you just mined a block!",
4   "previous_hash": "691f027f1b0aaf810ecf5242766b11bee4041e325db8974fb5aa0b6a3fc17547",
5   "proof": 45293,
6   "timestamp": "2021-12-30 12:06:02.268808"
7 }
```

Find and Replace Console Bootcamp Runner Trash

Type here to search 21°C ENG 12:06 30-12-2021

Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:5000/mine_block No Environment

Collections + New Collection

APIs

Environments

Mock Servers

Monitors

Flows

History

http://127.0.0.1:5000/mine_block

GET http://127.0.0.1:5000/mine_block Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (4) Test Results 200 OK 71 ms 349 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "index": 4,
3   "message": "Congratulations, you just mined a block!",
4   "previous_hash": "48848053556f405d9253225f95200f0997aaa4c8e8b173747d09081983397fa34",
5   "proof": 21391,
6   "timestamp": "2021-12-30 12:06:15.259601"
7 }
```

Find and Replace Console Bootcamp Runner Trash

Type here to search 21°C ENG 12:06 30-12-2021

Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:5000/get_chain No Environment

Collections + New Collection

APIs

Environments

Mock Servers

Monitors

Flows

History

http://127.0.0.1:5000/get_chain

GET http://127.0.0.1:5000/get_chain Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (4) Test Results 200 OK 14 ms 700 B Save Response

Pretty Raw Preview Visualize JSON

```
1
2
3
4   "chain": [
5     {
6       "index": 1,
7       "previous_hash": "0",
8       "proof": 1,
9       "timestamp": "2021-12-30 12:03:27.963689"
10    },
11    {
12      "index": 2,
13      "previous_hash": "7011fb74e4ecb3290651ac37adc5dfc160366b2985ad4f4a483f79b93f408923",
14      "proof": 533,
15      "timestamp": "2021-12-30 12:05:39.700283"
16    },
17    {
18      "index": 3,
```

Postman

File Edit View Help

Home Workspaces API Network Reports Explore Search Postman

My Workspace New Import Overview GET http://127.0.0.1:5000/get_chain No Environment

Collections + New Collection

APIs

Environments

Mock Servers

Monitors

Flows

History

http://127.0.0.1:5000/get_chain

GET http://127.0.0.1:5000/get_chain Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (4) Test Results 200 OK 14 ms 700 B Save Response

Pretty Raw Preview Visualize JSON

```
14
15
16   "index": 3,
17   "previous_hash": "691f027f1b0aaf810ecf5242766b11bee4041e325db8974fb5aa0b6a3fc17547",
18   "proof": 45293,
19   "timestamp": "2021-12-30 12:06:02.268888"
20 },
21 {
22   "index": 4,
23   "previous_hash": "48840653556f405d9253225f95200f0997aaa4c8e0b173747d0901983397fa34",
24   "proof": 21391,
25   "timestamp": "2021-12-30 12:06:15.259601"
26 },
27 ],
28 "length": 4
29
```