

---

# “Y-Coin”

- A custom created cryptocurrency build and delivered using Solarity and MyEtherWallet

Made by [Yashraj Shukla](#) ([Linkedin](#)) ([Github](#))

Duration- October 15th to December 30th, 2021

Y-Coin is a custom private cryptocurrency implemented using its own blockchain protocol, which uses Postman API calls to interact with requests.

This project involves challenges like creating your own blockchain, distributing the blockchain over the network of nodes, *mining blocks*, transactions of cryptocurrency with additional security, creating smart contracts for your crypto distribution—

## Working of software [ Theory ]

My Notes on Blockchain and Ledger are available here-

 [BlockChain myNotes](#)

1. BLOCKCHAIN: The blockchain is created using python3 and flask framework. Y\_coin\_blockchain is the class which has all the functions our blockchain will need. Creating a genesis block, mine a block, add a transaction and validating a block, check if its a legal block or not, update the blockchain.
  2. TRANSACTIONS and DISTRIBUTING THE BLOCKCHAIN NETWORK:  
In the flask app we use the functions to connect nodes together, we decentralize the blockchain network. We add transactions in a block.
-

3. SMART CONTRACT FOR ICO OF THE COIN: For the ICO of the coin we write a smart contract with the help of remix solidity. We can also push other type of code and software in our blockchain as smart contracts.

## How to compile and run the software

### Pre-Requisites

1. Flask version 0.12.2  
Install command "pip install Flask==0.12.2"
2. Postman API integrated software. Installation:  
[PostmanAPI](#)
3. For running Smart Contracts on the blockchain: you need  
[Ganache](#)
4. For a user friendly UI based wallet for transactions [MyEtherWallet](#) is used.
5. Install request module in python for json file request and conversions pip install requests==2.18.4

### Running the software

1. Run the blockchain program written in python3, allot it to a port
2. Run postman API and use respective commands to get chain, mine block, add transactions, update chain,
3. For interacting with smart contracts you need to use myEtherWallet and Ganache suit for temporary virtual ether and prewritten demo private and public keys. get key from dictionary

## Commands

1. **chain\_status** *"display the updated chain"*
2. **mine\_block** *inserts a new block in the chain*
3. **update\_chain** *checks if a longer chain exist in the network*

4. **is\_valid** *checks if the mined block is valid or not*
5. **add\_transactions** *adds a transaction from the sender to the receiver with the given amount*
6. **connect\_node** *connects the given node with the other nodes on the network*
7. **update\_chain** *fetches the latest longest blockchain and updates the blockchain on the current node with it.*