

# Ethereum using Solidity

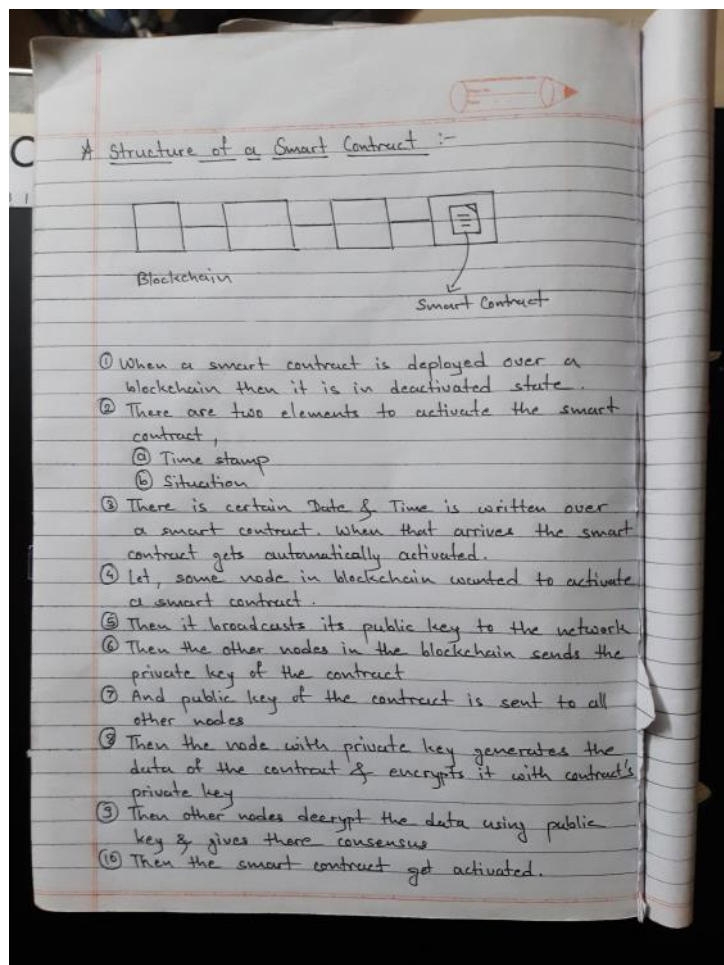
Friday, December 23, 2022 9:30 PM

## Smart Contract -



- A smart contract is a self-executing contract in which the agreements between two parties are written in the form of code.
- It is distributed over a decentralized network i.e. Blockchain and it is available to everyone involved in the agreement
- The execution of it is handled by the program written
- For ex. There is team A of developers who want to develop some software and they require funds
- They can get funds from blockchain network by contracting with multiple donors
- Then team A and donors D1, D2, D3 and D4 decides on certain agreements and these are written in the form of a code
- Then this code or program then added to blockchain and it is made public
- Now it becomes immutable that is no one can change the agreements made
- Now whenever team A completes some milestone then the desired amount written in smart contract will be automatically sent to them
- There is no need of any third party to deal with all the stakeholders
- Nick Szabo coined the term Smart Contract
- Smart contracts can be written in programming languages like Solidity or Python

## Working of a Smart Contract -



## Types of Smart Contract -

### 1. Smart Legal Contract -

- It has similar legal agreements written in the code like traditional legal contract
- After deployed on the blockchain it is applicable to the parties involved in it
- When someone violates the agreement then the smart contract automatically triggers the legal action against them
- It is the most common type of smart contract being used

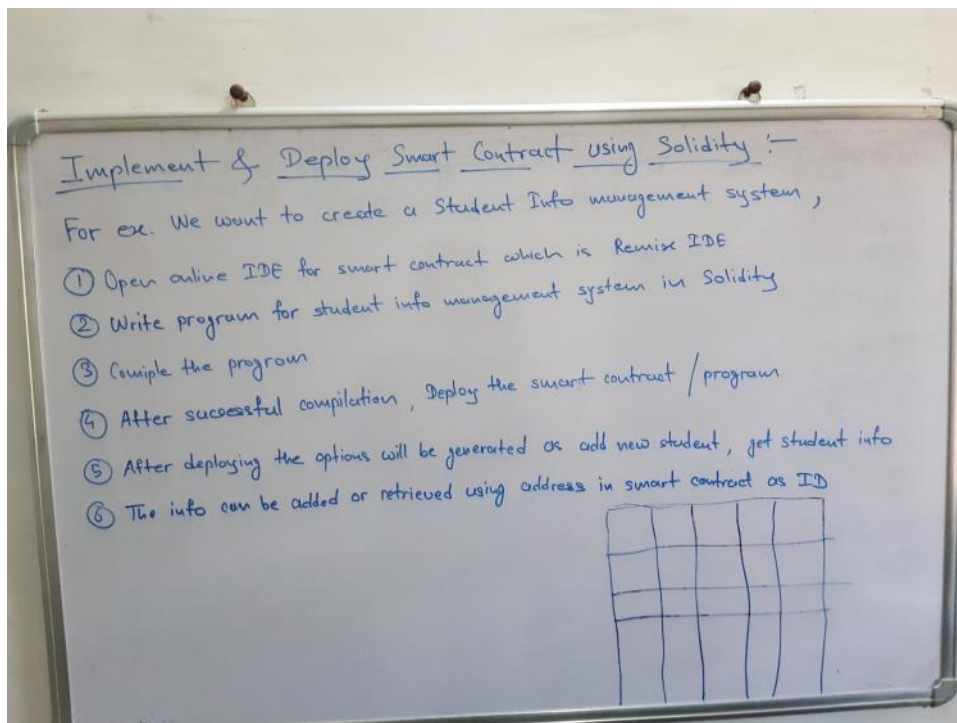
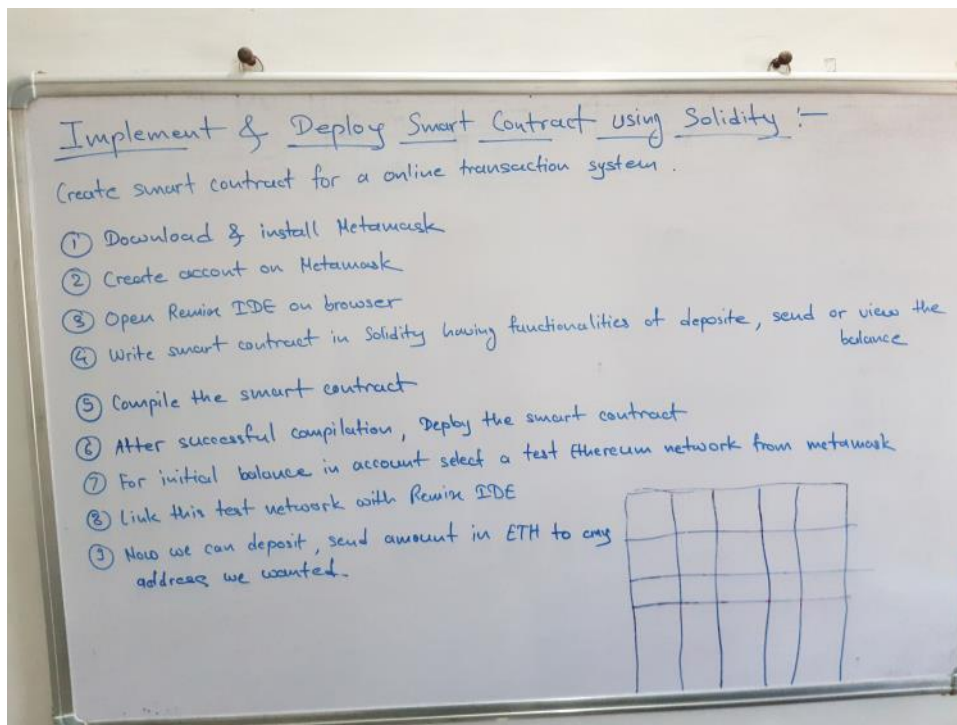
### 2. Decentralized Autonomous Organization -

- DAO are the communities that exist over blockchain
- These communities are defined over some rules and agreements written in the form of smart contract
- Every user has to abide by the rules of contract
- The decision taken are democratized.

### 3. Application Logic Contracts -

- ALC acts as an interface for communicating between smart contracts
- These can be used on various devices to use a smart contract
- They contain application based codes

## Implementing and deploying smart contract -



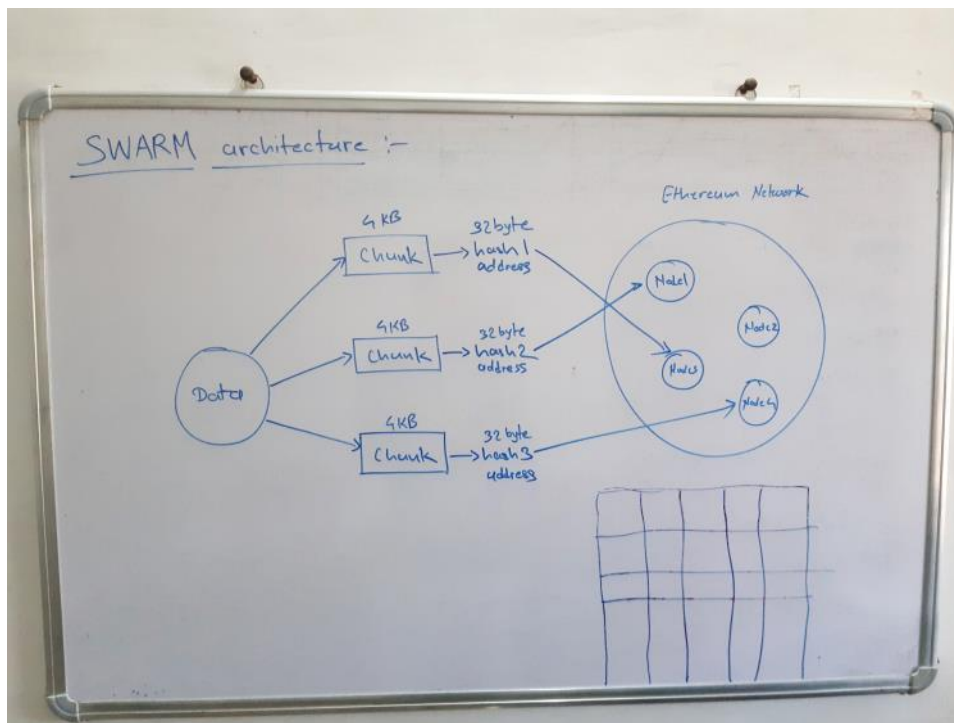
## SWARM -

- SWARM is a decentralized storage platform implemented over blockchain
- It is a data storage protocol for the Ethereum
- Decentralized storage means there the data is not stored on a single central server or in a single copy
- Data is stored on multiple nodes on a blockchain network and in the chunk format

- Chunks are the parts of the data
- SWARM provides permissionless and censorship resistant storage platform
- It also aims to provide WEB 3.0 services like audio, video streaming and database hosting, etc.
- It is similar to the P2P networks like Torrent
- On torrent the data is stored on multiple computers and the users share data in the form of seeds or parts
- The data being uploaded on SWARM is divided into chunks and each chunk is hashed using some hashing algo
- The chunk size should be greater than 4KB
- The hash value is address to some node in the network and it is 32 Byte long
- When we have to access the chunk then we can use these hash values as reference to access the data
- It is in the development phase

#### SWARM architecture -

- The hash values which are used for referencing are not user friendly and not human readable
- So an interface is introduced for it called as Ethereum Name Service or ENS
- ENS allows users to name their chunks of data as they wanted
- ENS is similar to DNS
- To use SWARM user must have an Ethereum account
- The token on SWARM is called as BZZ
- BZZ is used to give incentive to the nodes to maintain and manage the SWARM network



#### Whisper -

- Whisper is the decentralized messaging protocol used by Ethereum
- It is used to communicate between dApps
- It is only used when data being transferred is small and it is not useful in real time communication
- It uses DEVp2p wire protocol for data routing and encryption

- It is very useful in blockchain because the communication in the blockchain is hidden
- The message content, sender, receiver info are hidden from network
- The message header contains topic like contact code topic, partitioned topic and negotiation topic
- When node 1 wants to communicate with node 2 then following flow happens,
  1. node 1 sends contact code topic to node 2
  2. node 1 waits for contact code topic of node 2
  3. node 1 then sends partitioned topic to the node 2
- If we wanted to do one to many messaging then symmetric encryption of message is done
- As anyone with the key can decrypt it and read it
- For one to one messaging asymmetric encryption is done
- The node having public key can only decrypt the message

#### Types of Ethereum network -

##### 1. Public network -

- a. MainNet
- b. TestNet

##### 2. Private network -

#### Public network -

A Public blockchain is accessible to anyone in the world. One can read or push transaction on a public blockchain and validate the transactions being executed on the blockchain. The kind of blocks which can be added to the blockchain is decided by a consensus by the peers.

#### Private network -

The write permission is in command of a central authority in the network, However, the transaction is fully transparent to every peer in the network. Read privileges can also be customized.

#### MainNet -

The ETHER or ETH carry the real value of ether on the MainNet. As the size of MainNet grows, you need more compute and storage power to validate the blocks on the MainNet. Anyone can connect to the "MainNet".

The blockchain network is in development

#### TestNet -

The Ether on the TestNet does not carry any real value and is only for a collaborative testing on the network.

Used for testing purposes.

Also used for troubleshooting.

Dummy network

Mainnet

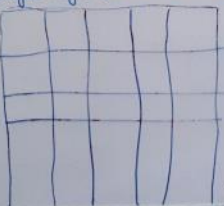
Testnet

Purpose	Functional Blockchain	Testing Environment
Transactions	Real Transactions Stored on Actual Blockchain	Fake Transactions
Coins	Posses Real Value	No Monetary Value
Transaction Cost	Paid Using Native Coins	No Cost (Free)
Transaction Frequency	High	Low
Mining	Economic Incentive to Earn Real Coins	No Economic Benefit

EVM -

Ethereum Virtual Machine :-

- ① EVM is a software used by Ethereum platform to execute smart contracts.
- ② EVM also computes state of Ethereum network after changes made to network.
- ③ Every node in Ethereum network has its own EVM running on there system
- ④ Smart contracts are written Solidity are not understandable to machines
- ⑤ After compiling a smart contract a Byte Code is generated
- ⑥ This byte code is executed by EVM
- ⑦ It is like a messenger between smart contracts
- ⑧ There can be multiple small smart contracts in a single large smart contract
- ⑨ To execute all those contracts properly which are interdependent, EVM plays vital role
- ⑩ EVM has a ROM, volatile memory RAM, storage.





## Ethereum Virtual Machine :-

Smart Contract  
written in Solidity

↓  
Compiler

↓  
Bytecode

↓  
EVM

↓  
Block added to  
Blockchain

Solidity

↓  
compiles to

Ethereum Bytecode

↓  
Executed by

EVM



## Ethereum Gas -

### Ethereum Gas :-

- ① It is a unit describing the amount of computational power required to execute some operation of a smart contract.
- ② For ex. we have to send some amount to other user then to execute the operation of sending, certain amount of Gas is required.
- ③ It acts as a fuel to the Ethereum protocol.
- ④ Standard limit of Gas per user is 21,000 units
- ⑤ Unit for the Gas is Gwei
- ⑥  $1 \text{ Gwei} = 1 \times 10^9 \text{ ETH}$
- ⑦ If a transaction req. 10,000 units of Gas & user has 21,000 units, then the transaction can be executed successfully
- ⑧ Now, other transaction requires 15,000 units but user has only 11,000 units & if user tries to execute it then the Gas will be consumed as well as the transaction will not execute.

