Assignment No : 01

- Title : Installation of MetaMask and study spending Ether per transaction.

- Objective : Students should be able to learn new tech. such as metamask. Its app. and implementations.

- Prerequisite : 1. Basic knowledge of cryptocurrency.
  2. Basic knowledge of distributed computing concept.
  3. Working of blockchain.

- Theory :

1. Introduction to Blockchain :

- Blockchain can be described as a data structure that holds transactional records and while ensuring security, transparency and decentralization.
- A blockchain is a distributed ledger that is completely open to any everyone on the network.
- Each transaction on a blockchain is secured with a digital signature that proves its authencity.
- Due to the use of encryption & digital signatures, the data stored on the blockchain is tamper-proof and cannot be changed.

- Features :

1] Decentralized : No single person or group holds the authority of the overall network.

2] Peer-to-Peer Network : It allows all the network participants to hold an identical copy of transactions, enabling approval through a machine consensus.

3] Immutable : It refers to the fact that any data once written on the blockchain cannot be changed.

4] Tamper-Proof : They are considered tamper-proof as any change in even one block can be detected and addressed smoothly.

- Benefits :

1] Time saving
2] Cost - saving
3] Tighter security

- How to use MetaMask : Step-by-step

Step 1 : Install MetaMask on your browser.
Step 2 : Create an account.
Step 3 : Depositing funds

- What advantages does MetaMask have ?

1. Popular : It is commonly used, so users only need one plugin to access a wide range of apps.
2. Simple : Instead of managing private keys, users just need to remember a list of words and transactions are signed on their behalf.
3. Saves space : User don't have to download Ethereum blockchain, as MetaMask sends requests to nodes outside of the user's computer.
4. Integrated : Dapps are designed to work with MetaMask, so it becomes much easier to send Ether in & out.

- Conclusion : In this way we have explored concept blockchain and metamat wallet for transaction of digital currency.

Assignment No : 02

Page No :
Date : / /

- Title : Create your own wallet using MetaMask for crypto transactions.

- Objective : Students should be able to learn about cryptocurrencies and learn how transaction done by using different digital currency.

- Prerequisite : 1. Basic knowledge of cryptocurrency
2. Working of blockchain.

- Theory :

Introduction to Cryptocurrency :

- Cryptocurrency is a digital payment system that doesn't rely on banks to verify transactions.
- It's a peer-to-peer system that can enable anyone anywhere to send and receive payments.
- Cryptocurrency is stored in a digital wallet.
- The first cryptocurrency was Bitcoin, which was founded in 2009 and remains the best known today.

- How does cryptocurrency works ?

- It works/run on a distributed public ledger called blockchain, a record of all transactions updated and held by currency holders.
- Units of cryptocurrency are created through a process called mining, which involves using computer power to solve complicated mathematical problems that generate coins.
- Users can also buy the currencies from brokers, then store and spend them using cryptographic wallets.

- Cryptocurrency example :

1] Bitcoin : Founded in 2009, Bitcoin was first cryptocurrency and is still the most commonly traded. The currency was developed by Satoshi Nakamoto- widely believed to be a pseudonym for an individual or group of people whose precise identify remains unknown.

2] Ethereum : Developed in 2015, Ethereum is a blockchain platform with its own cryptocurrency, called Ether (ETH) It is the most popular cryptocurrency after bitcoin.

3] Litecoin : This currency is most similar to bitcoin but has moved more quickly to develop new innovations, including faster payments.

4] Ripple : It is a distributed ledger system that was founded in 2012. It can be used to track different kinds of transactions, not just crypto.

- How to store cryptocurrency ?

- Once you have purchased cryptocurrency, you need to store it safely to protect it from hacks or theft.
- Usually, cryptocurrency is stored in crypto wallets, which are physical devices or online softwares used to store the private keys to your cryptocurrencies securely.
- There are different wallet providers to choose from. The terms "hot wallet" and "cold wallet" are used:

1] Hot wallet storage : It refers to crypto storage that uses online software to protect the private keys to your assets.

2] Cold wallet storage : Unlike hot wallets, cold wallets rely on offline electronic devices to securely store your private keys.

- Conclusion : In this way we have explored concept of cryptocurrency and learn how transactions are done using digital currency.

Assignment No : 03

Page No :
Date : / /

- Title : Write a smart contract on a test network, for bank account of a customer for following operations :
  - Deposit money
  - Withdraw money
  - Show balance

- Objective : Students should be able to learn new tech. such as metamask. Its app. and implementations.

- Prerequisite : 1. Basic knowledge of cryptocurrency
  2. Basic knowledge of distributed computing concept
  3. Working of blockchain.

- Contents of theory :
  The contract will allow deposits from any account and can be trusted to allow withdrawls only by accounts that have sufficient funds to cover the requested withdrawl.

```
contract TipJar {
  address owner;   //current owner of the contract
  function TipJar() public {
  owner = msg.sender;
 }
 function withdraw() public {
 require ( owner == msg.sender);
 msg.sender. transfer ( address (this). balance);
 }
 function deposit (uint256 amount) public payable {
   require (msg.value == amount);
 }
```

```
  function getBalance () public view returns (uint 256) {
  return address (this). balance;
  }
}
```

I am going to generalize this contract to keep track of ether deposits based on the account address of the depositor, and then only allow that same account to make withdrawals of that ether.

```
contract Bank {
  mapping (address => uint256) public balanceOf;
  function deposit (uint256) public payable {
    require (msg. value == amount);
    balanceOf [msg. sender] += amount;
  }
}
```

Here are the new concepts in the code above:

- mapping (address => uint 256) public balanceOf; declares a persistent public variable, balanceOf, that is a mapping from account addresses to 256-bit unsigned integers.
- Mappings can be indexed just like arrays / lists / tables in most modern prog. languages.
- The value of a missing mapping value is 0.

```
contract Bank {
  mapping (address => uint256) public balanceOf;
  Function deposit (uint256amount) public payable {
  require (msg. value == amount);
} balanceOf [msg. sender] += amount;
```

```
 function withdraw (uint256 amount) public {
  require (amount <= balanceOf [msg.sender]);
  balanceOf [msg.sender] -= amount;
  msg.sender.transfer (amount);
 }
}
```

The code above demonstrates the following :

- The require (amount <= balances [msg.sender]) checks to make sure the sender has sufficient funds to cover the requested withdrawl. If not then the transaction aborts without making any state changes or ether transfers.
- The balanceOf mapping must be updated to reflect the lowered residual amount after the withdrawl.
- The funds must be sent to the sender requesting the withdrawl.

Conclusion : Hence we have written a smart contract on a test network, for bank account of a customer.