

ASSIGNMENT NO.: 1

AIM: Installation of MetaMask and study spending Ether per transaction.

INTRODUCTION:

❖ METAMASK

- MetaMask is available as an in-browser applications for desktop or laptop computers, as well as a smartphone app available on both major app stores what makes it so. Popular is its seamless integration with many different major crypto. Websites, including NFT marketplace OpenSea, and decentralized. Exchanges including 1inch, Uniswap, and Quickswap.
- MetaMask also works with hardware cryptocurrency wallets. Ledger and Trezor, allowing users to transfer crypto and NFTS from the software-based hot wallet to the hardware based cold wallets for secure storage
- It is easy to use Meta Mask on a connected website to send, receive, or trade to kens before exchanging any tokens. MetaMask will pop-up in the browser to confirm the details, including the contract price. Before confirming the transaction,, users will be able to adjust their gas limits to either pay more to speed it up, or reduce their max price. While slowing down the confirmation.

❖ REMIX IDE

Platform to create and deploy smart contract, supports solidity.

❖ SOLIDITY

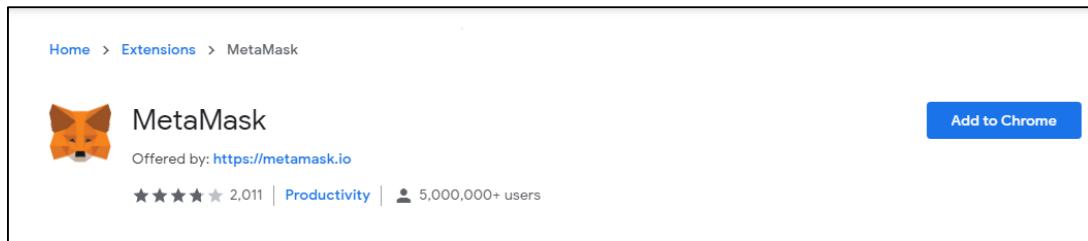
A Language to create smart contracts, similar to JavaScript. reating a De-centralized platform for testing a smart contract.

❖ STEPS TO INSTALL METAMASK ON GOOGLE CHROME.

Step 1: Go to Chrome Web Store Extensions Section.

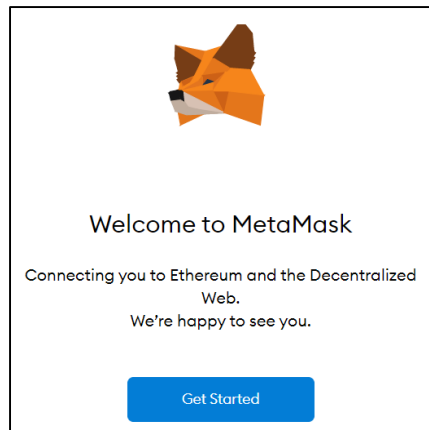
Step 2: Search *MetaMask*.

Step 3: Check the number of downloads to make sure that the legitimate MetaMask is being installed, as hackers might try to make clones of it.

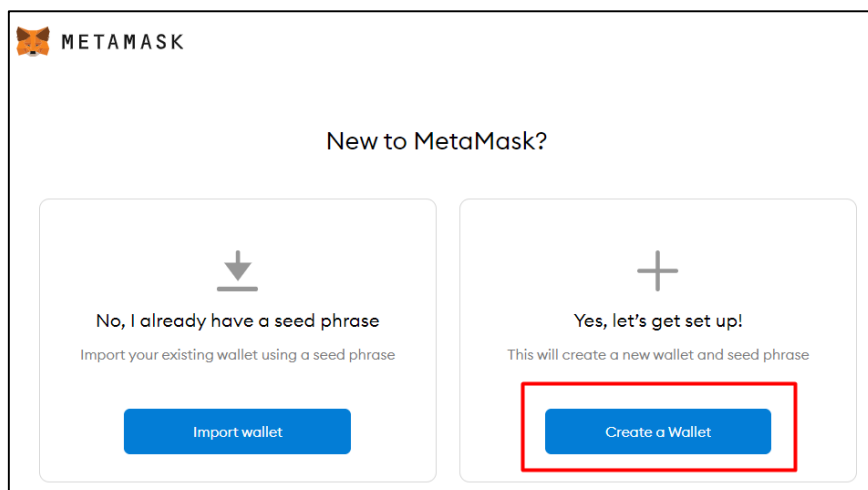


Step 4: Click the *Add to Chrome* button.

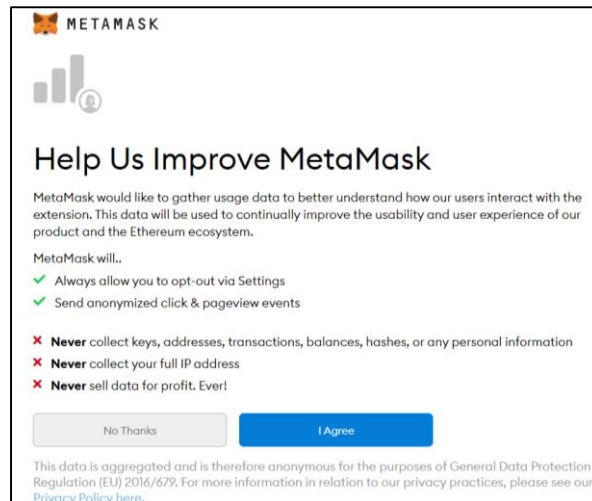
Step 5: Once installation is complete this page will be displayed. Click on the *Get Started* button.



Step 6: This is the first time creating a wallet, so click the *Create a Wallet* button. If there is already a wallet then import the already created using the *Import Wallet* button.



Step 7: Click *I Agree* button to allow data to be collected to help improve MetaMask or else click the *No Thanks* button. The wallet can still be created even if the user will click on the *No Thanks* button.




Step 8: Create a password for your wallet. This password is to be entered every time the browser is launched and wants to use MetaMask. A new password needs to be created if chrome is uninstalled or if there is a switching of browsers. In that case, go through the *Import Wallet* button. This is because MetaMask stores the keys in the browser. Agree to *Terms of Use*.

A screenshot of the MetaMask 'Create Password' screen. At the top is the MetaMask logo. Below it is a '< Back' link. The title 'Create Password' is prominently displayed. Below the title, there is a text input field for 'New password (min 8 chars)'. Below that is another text input field for 'Confirm password'. At the bottom, there is a checkbox followed by the text 'I have read and agree to the Terms of Use'. At the very bottom is a blue 'Create' button.

Step 9: Click on the dark area which says *Click here to reveal secret words* to get your secret phrase.


Step 10: This is the most important step. Back up your secret phrase properly. Do not store your secret phrase on your computer. Please read everything on this screen until you understand it completely before proceeding. The secret phrase is the only way to access your wallet if you forget your password. Once done click the *Next* button.

 METAMASK

Secret Backup Phrase

Your secret backup phrase makes it easy to back up and restore your account.

WARNING: Never disclose your backup phrase. Anyone with this phrase can take your Ether forever.


CLICK HERE TO REVEAL SECRET WORDS

[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 Backup Phrase and keep it stored safely on an external encrypted hard drive or storage medium.](#)

Step 11: Click the buttons respective to the order of the words in your seed phrase. In other words, type the seed phrase using the button on the screen. If done correctly the *Confirm* button should turn blue.

 METAMASK

< Back

Confirm your Secret Backup Phrase

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

burger

buyer

detail

fire

fossil

hold

rain

search

slight

spray

tube

wire

Confirm

Step 12: Click the *Confirm* button. Please follow the tips mentioned.



Congratulations

You passed the test - keep your seedphrase 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 seed phrase.
- If you need to back up your seed 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 seedphrase. [Learn more](#).

All Done

CONCLUSION: We have successfully studied the Installation of Metamask.

ASSIGNMENT NO.: 2

AIM: Create your own wallet using MetaMask for crypto transactions.

INTRODUCTION:

❖ A BRIEF INTRODUCTION TO METAMASK

MetaMask is an open-source, straightforward, and easy-to-use cryptocurrency wallet. It functions as a web browser extension available for Chrome, Firefox, Brave, or a mobile application for iOS or Android. Initially, this wallet supported only Ether and ERC-20 tokens, and now it is compatible with ERC-721 and ERC-1155 token standards. Furthermore, MetaMask benefits include interaction with websites; hence, it can function as a connection node for various DApps on Ethereum.

Adrian Devis and Dan Finlay are the MetaMask developers. Their idea was revolutionary and straightforward; they intended to create a web browser extension that would allow managing cryptocurrency and using the browser for fast and secure access with DApps. ConsenSys Software Inc. — a development company, focusing on applications that use Ethereum 's blockchain, implemented the idea in 2016. The solution used Ethereum 's interface and a web API called web3.js. This Ethereum library is the fundament of MetaMask since it allows the browser to interact with the local or remote blockchain nodes via HTTP, IPC, and WebSocket; also, it gained the ability to record and read data from smart contracts, transfer tokens, etc. In another way, web3.js allowed the blockchain developers to create proxy and communication bridges between MetaMask, DApps, and the user.

Adrian Devis and Dan Finlay admit that their idea was great. Yet, the technical implementation was super complicated, especially in providing security for the users (web wallets are considered the most vulnerable to hacker attacks). Nonetheless, ConsenSys succeeded, and on the 14th of July in 2016, they offered the first version of MetaMask web browser cryptocurrency wallet for Chrome. Later, they presented a version for Firefox, Brave, and other popular browsers. In 2019 they also launched the mobile version of the MetaMask cryptocurrency wallet.

❖ HOW DOES THE METAMASK WALLET FUNCTION?

As we mentioned above, the MetaMask cryptocurrency wallet employs the web3.js library to function. This library is a part of the official Ethereum product. The library was developed focusing on the requirements of web applications that could interact with the Ethereum blockchain and take advantage of all blockchain's benefits and functions. MetaMask is a cryptocurrency wallet for Ethereum and an instrument that helps to interact with DApps. MetaMask connects the extension to the DApp so that to fulfill both tasks. When the application identifies the MetaMask, it creates a connection, and the user can start using all the features of a specific application.

For instance, it can assets trading, access to resources or services, or any other task within the capability of a DApp. Each action has its cost (transaction fee) that must be paid in Ethereum or any specified token. MetaMask wallet has all instruments and protocols for this purpose.

Hence, we can state that Metamask also controls the interaction of the user and DApp, and processes the operations required for specific actions, besides the function of a wallet. Reliable and secure cryptography and safe internet connection are the environments for these operations.

Furthermore, MetaMask can generate asymmetric keys, store them on a local device, and manage access to the keys. To sum up, MetaMask is a super-safe extension.

❖ EXTENDED FUNCTIONS SET FOR METAMASK CLONE

To help your MetaMask wallet clone become famous, you should add some advantages that highlight it from the competitors and improve the user experience.

These can be the following:

Linking an account.

Your users will find it useful to be able to buy a cryptocurrency and exchange it for fiat within the wallet. This will be possible if you develop a wallet like MetaMask and add the feature of linking bank accounts, credit/debit cards, PayPal, or other online payment systems.

eCommerce integrations.

We mean integrating the wallet with exchanges, NFT marketplaces, decentralized applications, shops, and other services that the users might find useful.

Multilingual interface.

If you focus on a market where all people speak the same language, you might neglect this aspect. However, your intentions are global, and you should add as many languages as possible to increase the target audience.

Push notifications.

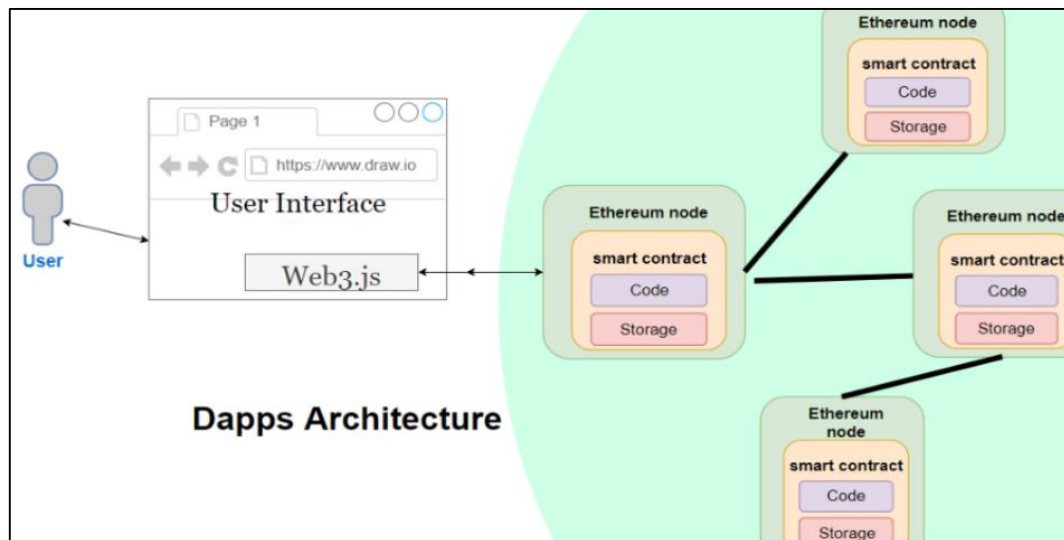
The notifications will inform the users of receiving payments, ending transactions, rapid exchange rate changes in the investment account, system updates, suspicious activity, etc.

VIP support.

Numerous cryptocurrency trading platforms offer support for an additional fee. This may include 24/7 support, communication with a personal specialist, etc.

QR scanner.

This is another useful feature that allows your users to make payments even faster. Moreover, it will decrease the number of transfers done by mistake.



CONCLUSION: We have successfully created our own wallet using Metamask for crypto transactions.

ASSIGNMENT NO.: 3

AIM: Write a smart contract on a test network, for Bank account of a customer for following operations:

- Deposit money
- Withdraw Money
- Show balance

INTRODUCTION:

The contract will allow deposits from any account, and can be trusted to allow withdrawals only by accounts that have sufficient funds to cover the requested withdrawal.

This post assumes that you are comfortable with the ether-handling concepts introduced in our post, writing a Contract That Handles Ether.

That post demonstrated how to restrict ether withdrawals to an “owner’s” account. It did this by persistently storing the owner account’s address, and then comparing it to the msg.sender value for any withdrawal attempt.

❖ CODE:

```
//SPDX-License-Identifier: MIT
pragma solidity ^0.6;

contract banking(
mapping(address=>uint) public user_account;
mapping(address=>bool) public user_exists;

function create ccount() public payable returns(string memory){
    require(user_exists[msg.sender]==false,'Account already created');
    if(msg.value==0){
user account [msg.sender]=0;
user_exists[msg.sender]=true;
    return "Account created";
    }
    require(user_exists[msg.sender] == false,"Account already created");
user account [msg.sender] = msg.value;
user_exists[msg.sender] = true;
return "Account created";
}

function deposit() public payable returns(string memory){
    require(user_exists[msg.sender]==true, "Account not created");
require(msg.value> 0, "Value for deposit is Zero");
```

```

user_account[msg.sender]=user_account(msg.sender)+msg.value;
    return "Deposited Successfully";
}

function withdraw(uint amount) public payable returns(string memory){
    require(user_account[msg.sender]> amount, "Insufficeint balance");
    require(user_exists[msg.sender]==true, "Account not created");
require(amount>0," Amount should be more than zero");
user_account[msg.sender]=user_account[msg.sender].amount;
msg.sender.transfer(amount);
    return "withdrawal Succesful"
}

function transfer(address payable userAddress, uint amount) public returns(string
memory){
    require(user account[msg.sender]>amount, "Insufficeint balance in Bank account");
    require(user_exists[msg.sender]==true, "Account is not created");
    require(user exists[userAddress]==true, "Tranfer account does not exist");
require(amount> 0, "Amount should be more than zero");
user account[msg.sender]=user_account[msg.sender]-amount;
user account[userAddress]=user_account[userAddress]+amount;
    return "Transfer succesful":
}

function send amt(address payable toAddress, uint256 amount) public payable
returns(string memory){
    require(user_account[msg.sender]>amount,"Insufficient balance in Bank account");
    require(user_exists[msg.sender]==true,"Account is not created");
require(amount>0,"Amount should be more than zero");
user_account[msg.sender]=user_account(msg.sender)-amount;
toAddress.transfer(amount);
    return 'transfer success';
}

function user_balance() public view returns(uint){
    return user_account[msg.sender];
}

function account_exist() public view returns(bool){
    return user_exists[msg.sender];
}
}

```

CONCLUSION: We have successfully created a smart contract on a test network for a bank account of a customer.

ASSIGNMENT NO.: 4

AIM: Write a program in solidity to create Student data. Use the following constructs:

- Structures
- Arrays
- Fallback

Deploy this as smart contract on Ethereum and Observe the transaction fee and Gas values.

INTRODUCTION:

❖ SOLIDITY:

Solidity is a high-level language. The structure of smart contracts in solidity is very similar to the structure of classes in object-oriented languages. The solidity file has an extension .sol.

❖ WHAT ARE SMART CONTRACTS?

Solidity's code is encapsulated in contracts which means a contract in Solidity is a collection of code (its functions) and data (its state) that resides at a specific address on the Ethereum blockchain. A contract is a fundamental block of building an application on Ethereum.

Example: In the below example, the aim is to deploy a Smart Contract for Marks Management System by using Solidity. In this contract, the details of every student like student ID, Name, department, etc can be added. After building the contract all the details of every student can be retrieved.

❖ CODE:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.6;

contract Student_management{

    struct Student{
        int stud_id;
        string Name;
        string department;
    }

    Student[] Students;
```

```

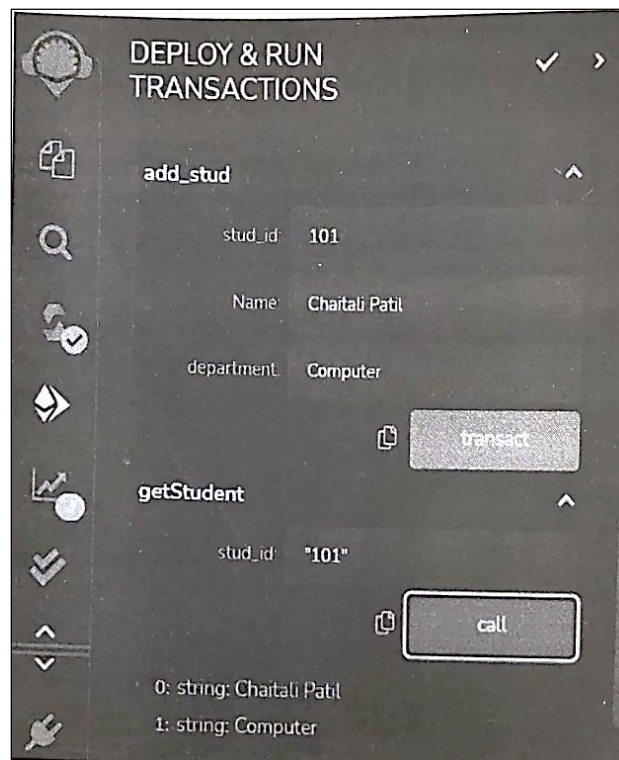
function add_stud(int stud_id, string memory Name, string memory department) public {
    Student memory stud = Student(stud_id, Name, department);
    Students.push(stud);
}

function getStudent(int stud_id) public view returns(string memory, string memory){
    for(uint i = 0; i < Students.length; i++){
        Student memory stud = Students[i];
        if(stud.stud_id == stud_id){
            return(stud.Name, stud.department);
        }
    }
    return("Not Found", "Not Found");
}
}

```

❖ OUTPUT:

After deploying the contract successfully, you can observe two buttons add_stud and getStudents. Give the input stud_id, name, dept and click on getStudents button, enter the stud_id which you have given as an input and get the information of students name and department.



CONCLUSION: We have successfully created a solidity to create student data.

ASSIGNMENT NO.: 5

AIM: Write a survey report on types of Blockchains and its real time use cases.

INTRODUCTION:

❖ BLOCKCHAIN OVERVIEW

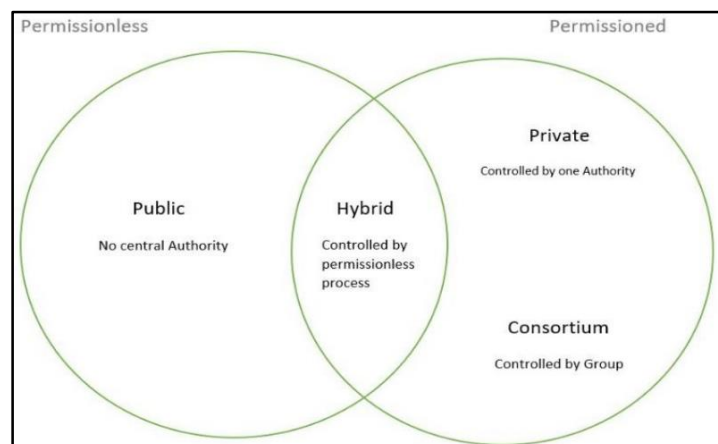
Blockchain is a shared, immutable ledger that facilitates the process of recording transactions and tracking assets in a business network. An *asset* can be tangible (a house, car, cash, land) or intangible (intellectual property, patents, copyrights, branding). Virtually anything of value can be tracked and traded on a blockchain network, reducing risk and cutting costs for all involved.

❖ WHY BLOCKCHAIN IS IMPORTANT?

Business runs on information. The faster it's received and the more accurate it is, the better. Blockchain is ideal for delivering that information because it provides immediate, shared and completely transparent information stored on an immutable ledger that can be accessed only by permissioned network members. A blockchain network can track orders, payments, accounts, production and much more. And because members share a single view of the truth, you can see all details of a transaction end to end, giving you greater confidence, as well as new efficiencies and opportunities.

❖ THERE ARE 4 TYPES OF BLOCKCHAIN:

1. Public Blockchain
2. Private Blockchain
3. Hybrid Blockchain
4. Consortium Blockchain



1. PUBLIC BLOCKCHAIN

- These blockchains are completely open to following the idea of decentralization. They don't have any restrictions, anyone having a computer and internet can participate in the network.
- As the name is public this blockchain is open to the public, which means it is not owned by anyone.
- Anyone having internet and a computer with good hardware can participate in this public blockchain.
- All the computer in the network hold the copy of other nodes or block present in the network.
- In this public blockchain, we can also perform verification of transactions or records.

Advantages:

1. Trustable: There are algorithms to detect no fraud. Participants need not worry about the other nodes in the network.
2. Secure: This blockchain is large in size as it is open to the public. In a large size, there is greater distribution of records
3. Anonymous Nature: It is a secure platform to make your transaction properly at the same time, you are not required to reveal your name and identity in order to participate.
4. Decentralized: There is no single platform that maintains the network, instead every user has a copy of the ledger.

Disadvantages:

1. Processing: The rate of the transaction process is very slow, due to its large size. Verification of each node is a very time-consuming process.
2. Energy Consumption: Proof of work is high energy-consuming. It requires good computer hardware to participate in the network
3. Acceptance: No central authority is there so governments are facing the issue to implement the technology faster.
4. Use Cases: Public Blockchain is secured with proof of work or proof of stake they can be used to displace traditional financial systems. The more advanced side of this blockchain is the smart contract that enabled this blockchain to support decentralization.

- Examples of public blockchain are Bitcoin, Ethereum.

2. PRIVATE BLOCKCHAIN

- These blockchains are not as decentralized as the public blockchain only selected nodes can participate in the process, making it more secure than the others.
- These are not as open as a public blockchain.

- They are open to some authorized users only.
- These blockchains are operated in a closed network.
- In this few people are allowed to participate in a network within a company/organization.

Advantages:

1. Speed: The rate of the transaction is high, due to its small size. Verification of each node is less time consuming.
2. Scalability: We can modify the scalability. The size of the network can be decided manually.
3. Privacy: It has increased the level of privacy for confidentiality reasons as the businesses required.
4. Balanced: It is more balanced as only some user has the access to the transaction which improves the performance of the network.

Disadvantages:

1. Security- The number of nodes in this type is limited so chances of manipulation are there. These blockchains are more vulnerable.
2. Centralized- Trust building is one of the main disadvantages due to its central nature. Organizations can use this for malpractices.
3. Count- Since there are few nodes if nodes go offline the entire system of blockchain can be endangered.
4. Use Cases: With proper security and maintenance, this blockchain is a great asset to secure information without exposing it to the public eye. Therefore companies use them for internal auditing, voting, and asset management.

- An example of private blockchains is Hyperledger, Corda.

3. HYBRID BLOCKCHAIN

- It is the mixed content of the private and public blockchain, where some part is controlled by some organization and other makes are made visible as a public blockchain.
- It is a combination of both public and private blockchain.
- Permission-based and permissionless systems are used.
- User access information via smart contracts
- Even a primary entity owns a hybrid blockchain it cannot alter the transaction

Advantages:

1. Ecosystem: Most advantageous thing about this blockchain is its hybrid nature. It cannot be hacked as 51% of users don't have access to the network
2. Cost: Transactions are cheap as only a few nodes verify the transaction. All the nodes don't carry the verification hence less computational cost.

3. Architecture: It is highly customizable and still maintains integrity, security, and transparency.
4. Operations: It can choose the participants in the blockchain and decide which transaction can be made public.

Disadvantages:

1. Efficiency: Not everyone is in the position to implement a hybrid Blockchain. The organization also faces some difficulty in terms of efficiency in maintenance.
 2. Transparency: There is a possibility that someone can hide information from the user. If someone wants to get access through a hybrid blockchain it depends on the organization whether they will give or not.
 3. Ecosystem: Due to its closed ecosystem this blockchain lacks the incentives for network participation.
 4. Use Case: It provides a greater solution to the health care industry, government, real estate, and financial companies. It provides a remedy where data is to be accessed publicly but needs to be shielded privately.
- Examples of Hybrid Blockchain are Ripple network and XRP token.

4. CONSORTIUM BLOCKCHAIN

- It is a creative approach that solves the needs of the organization. This blockchain validates the transaction and also initiates or receives transactions.
- Also known as Federated Blockchain.
- This is an innovative method to solve the organization's needs.
- Some part is public and some part is private.
- In this type, more than one organization manages the blockchain.

Advantages:

1. Speed: A limited number of users make verification fast. The high speed makes this more usable for organizations.
2. Authority: Multiple organizations can take part and make it decentralized at every level. Decentralized authority, makes it more secure.
3. Privacy: The information of the checked blocks is unknown to the public view. but any member belonging to the blockchain can access it.
4. Flexible: There is much divergence in the flexibility of the blockchain. Since it is not a very large decision can be taken faster.

Disadvantages:

1. Approval: All the members approve the protocol making it less flexible. Since one or more organizations are involved there can be differences in the vision of interest.
2. Transparency: It can be hacked if the organization becomes corrupt. Organizations may hide information from the users.

3. **Vulnerability:** If few nodes are getting compromised there is a greater chance of vulnerability in this blockchain
 4. **Use Cases:** It has high potential in businesses, banks, and other payment processors. Food tracking of the organizations frequently collaborates with their sectors making it a federated solution ideal for their use.
- Examples of consortium Blockchain are Tendermint and Multichain.

❖ TOP BLOCKCHAIN USE CASES

Blockchain is "a general-purpose technology, which means it is applicable across sectors," said Christos Makridis, a research professor at Arizona State University, senior adviser at Gallup, digital fellow at Stanford University's Digital Economy Lab and CTO at arts and education technology start-up Living Opera.

1. **Smart contracts.** The primary function of computer programs called "smart contracts" is to automate the execution of contract terms when conditions warrant them. The computer code follows a relatively simple command of "when/if _then_" to ensure that all parties receive the benefits or penalties as the contract stipulates and actions require.
2. **Cybersecurity.** Blockchains are highly secure because of their permanency, transparency and distributed nature. With blockchain storage, there's no central entity to attack and no centralized database to breach. Because blockchains are decentralized, including those privately owned, and the data stored in each block is unchangeable, criminals can't access the information.
3. **IoT.** Two primary IoT uses of blockchains are in the supply chain sector and for asset tracking and inventory management. A third use is in recording measurements made by machines whether those sensors are in the Artic, the Amazon jungle, a manufacturing plant or on a NASA drone surveying Mars.
4. **Cryptocurrencies.** The blockchain concept was originally developed to manage digital currencies such as bitcoin. While the two technologies still compete against each other in alternative transactions, they've also been separated so blockchains could serve other purposes. Given the anonymity of crypto coins, blockchain is the only way to document transactions with accuracy and privacy for the parties involved.
5. **NFTs.** Nonfungible tokens are units of data certified to be unique and not interchangeable. In short, they are digital assets. According to Rafferty, NFTs are revolutionizing the digital art and collectibles world.

CONCLUSION: We have successfully studied types of blockchain and its applications in real time.