**INTERNSHIP REPORT**

**Blockchain Technology:Transportation of gasoline using blockchain technology.**



Under the guidance of:

**MS. VARSHA NIRBHAVNE**

Dy. General Manager (BIS)

Submitted By:

**Aditi Jain**

**CONTENTS**

1. Acknowledgement
2. Abstract
3. About GAIL(INDIA) Limited
   1. Gail Projects
      1. Gail Pankh
      2. Gail Hriday
   2. Gail’s Business Portfolio
4. Chapter-1 “What is Blockchain”?
   1. Immutability
   2. Distribution
   3. Terminology
5. Chapter-2 Working of Blockchain
   1. Big Picture
   2. Part-1 (Hash Functions)
   3. Part-2 (Public Key Cryptography)
   4. Part-3 (Adding a Digital Signature)
   5. Part-4 (Data Storage in Blocks)
   6. Part-5 (Mining)
   7. Part-6 (Applications of Blockchain Technology)
6. Chapter-3 Problem Statement
7. Chapter-4 Ethereum Blockchain
   1. Part-1 (Interfacing with Ethereum network)
   2. Part-2 (What is transaction)
   3. Part-3 (Cryptography)
8. Chapter-5 SOLIDITY, WEB 3JS, REMIX
9. Smart Contract
10. Conclusion

**AKNOWLEDGEMENT**

I am overwhelmed in all humbleness and gratefulness to acknowledge my depth to those who have helped me put these ideas well above the level of simplicity and into something concrete.

At very outside I am thankful to **GAIL (INDIA) Limited,** for giving me this opportunity to work on this project and making available all kinds of resources in order to complete this project successfully.

I would like to express my sincere gratitude to **Ms. VARSHA NIRBHAVNE – Dy. General Manager (BIS)** supervisor for her motivation, guidance and co-operation in the completion of this 4 weeks of summer internship as well as for providing her invaluable guidance, comments and suggestions throughout the course of the project and for constantly motivating me to work harder, as without her guidance it would have been difficult to overcome the challenges faced during the execution of this project.

**ABSTRACT**

Given geographic, political and operational realities, along with the scale of investment required within the industry, companies in chemicals and petroleum are connected through a diverse network to other supply chains. Blockchain here can provide secure transactions, reduce compliance costs, and speed up data transfer processing. Blockchain is more than just a disruptive technology. It represents an opportunity for chemicals and petroleum companies to increase their profitability in today’s economic environment and well into the future

In the future, as blockchain-enabled business transactions become more sophisticated, business and industry networks could evolve into self-governing cognitive business networks and reduce the cost of payment transactions by an estimated 30 percent. Much more than a technology to automate business transactions, blockchains create a new model for trust by establishing transactional relationships between businesses via smart contracts, certifications and digital compliance.

This project implements transportation of Gasoline between a Producer and a Distributor where an order is initiated by a Distributor and accepted by the producer if sufficient stock is available. During transportation, quality and quantity check is maintained with the help of sensors. If specifications are not met and quality deteriorates, payment amount decreases accordingly

**ABOUT GAIL (INDIA): Gas Authority Of India Limited.**

***“Enhancing quality of life through clean energy and beyond****”.*

GAIL (India) Limited (formerly known as Gas Authority of India Ltd.) is a government owned natural gas cooperation currently chaired and directed by Mr. Manoj Jain responsible for natural gas processing and distribution in India headquartered in New Delhi, India. It is under the ownership of Ministry of Petroleum and Natural Gas , Government of India. It has the following business segments: natural gas, liquid hydrocarbon, liquified petroleum gas transmission, petrochemical, city gas distribution, exploration and production, GAILTEL and electricity generation. GAIL was conferred with the Maharatna status on 1 Feb 2013, by the Government of India. Only eight other Public Sector Enterprises (PSEs) enjoy this coveted status amongst all central CPSEs. GAIL was listed in the 131st position among India’s most trusted brands according to the Brand Trust Report 2014, a study conducted by the Trust Research Advisory.

GAIL (India) Limited was incorporated in August 1984 as a Central Public Sector Undertaking (PSU) under the Ministry of Petroleum & Natural Gas (MoP&NG). This Company was formerly known as Gas Authority of India Limited (GAIL). GAIL is India’s principal gas transmission and marketing company.

**August 10, 2021**

Gail awarded for making highest procurement among CPSEs through GeM portal.

**GAIL PROJECTS:**

**GAIL PANKH:**

In a bid to support the Start-Ups operating in identified Focus Areas, GAIL (India) Limited announced its plans to invest in such companies through its Start-Up initiative ‘Pankh’.  
  
GAIL has opened a fresh Round for Solicitation of Investment Proposals from the Start-Ups operating in the Focus Areas, which mainly includes natural gas, petrochemicals, energy, project management, bio-manure marketing, nano materials, IoT, data mining, environment, health and social. The further details of Focus Areas are available on GAIL website under GAIL PANKH Section. The Start-Ups that are interested in equity investment from GAIL can apply through link ‘GAIL PANKH’ on GAIL’s website <https://gailonline.com>.

**GAIL HRIDAY:**

GAIL remains committed to the principles of Corporate Social Responsibility (CSR) which it strongly believes play a defining role in the development of the country. GAIL has implemented CSR programmes primarily in the rural areas which are in close proximity to the major work centres/installations of the Company. As identified under Schedule VII of the Companies Act, 2013, GAIL has identified seven broad areas of CSR intervention, each of which is titled by the objective they seek to achieve viz. Arogya (Wellness) – Nutrition, Health and Sanitation and Drinking Water projects etc.

***VISION:*** *“Be the leader in natural gas value-chain and beyond, with global*

*presence, creating value for stakeholders with environmental*

*responsibility”.*

**GAILS’s Business Portfolio includes:**

1. 7,700 km of Natural Gas high pressure trunk pipeline with a capacity to carry 157 MMSCMD of natural gas across the country
2. 7 LPG Gas Processing Units to produce 1.2 MMTPA of LPG and other liquid hydrocarbons
3. North India’s only gas based integrated Petrochemical complex at Pata with a capacity of producing 4,10,000 TPA of Polymers
4. 1,922 km of LPG Transmission pipeline network with a capacity to transport 3.8 MMTPA of LPG
5. 27 oil and gas Exploration blocks and 3 Coal Bed Methane Blocks
6. 13,000 km of OFC network offering highly dependable band with for telecom service providers
7. Joint venture companies in Delhi, Mumbai, Hyderabad, Kanpur, Agra, Lucknow, Bhopal, Agartala and Pune, for supplying Piped Natural Gas (PNG) to households and commercial users, and Compressed Natural Gas (CNG) to the transport sector
8. Established presence in the CNG and City Gas sectors in Egypt through equity participation in three Egyptian companies: Fayum Gas Company SAE, Shell CNG SAE and National Gas Company SAE
9. A wholly-owned subsidiary company GAIL Global (Singapore) Pte Ltd in Singapore

**CHAPTER-1**

**WHAT IS “BLOCKCHAIN”?**

Blockchain technology is most simply defined as a decentralized, distributed ledger that records the provenance of a digital asset. By inherent design, the data on a blockchain is unable to be modified, which makes it a legitimate disruptor for industries like Transportation and oil, payments, cybersecurity and healthcare. 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.

**HISTORY**

Blockchain has the potential to grow to be a bedrock of the worldwide record-keeping systems, but was launched just 10 years ago. It was created by the unknown persons behind the online cash currency bitcoin, under the pseudonym of Satoshi Nakamoto.

Posting their seminal whitepaper in 2008 and launching the initial code in 2009, Nakamoto created bitcoin to be a form of cash that could be sent peer-to-peer without the need for a central bank or other authority to operate and maintain the ledger, much as how physical cash can be.

While it wasn’t the first online currency to be proposed, the bitcoin proposal solved several problems in the field and has been by far the most successful version. The engine that runs the bitcoin ledger that Nakamoto designed is called the blockchain; the original and largest blockchain is the one that still orchestrates bitcoin transactions today.

Blockchain is a term used to describe **DLT**, or **Distributed Ledger Technology**. It is used to create a storage system for data in a distributed and immutable manner.

**#IMMUTABILITY**

This means that once data is written to a blockchain data store or **ledger**, it cannot be changed – it’s there forever. In contrast, in a standard Relational Database, no matter how much security you implement, the data can be accessed and modified on the file system on which the data is persisted. This could be done by corrupt admin or a hacker.

A blockchain system ensures that even if a bit of data is changed at any level on the ledger, the entire system will report an invalid state. And since the data is distributed on multiple systems, the actual data with a valid state can be recovered from one of the systems.

#**DISTRIBUTION**

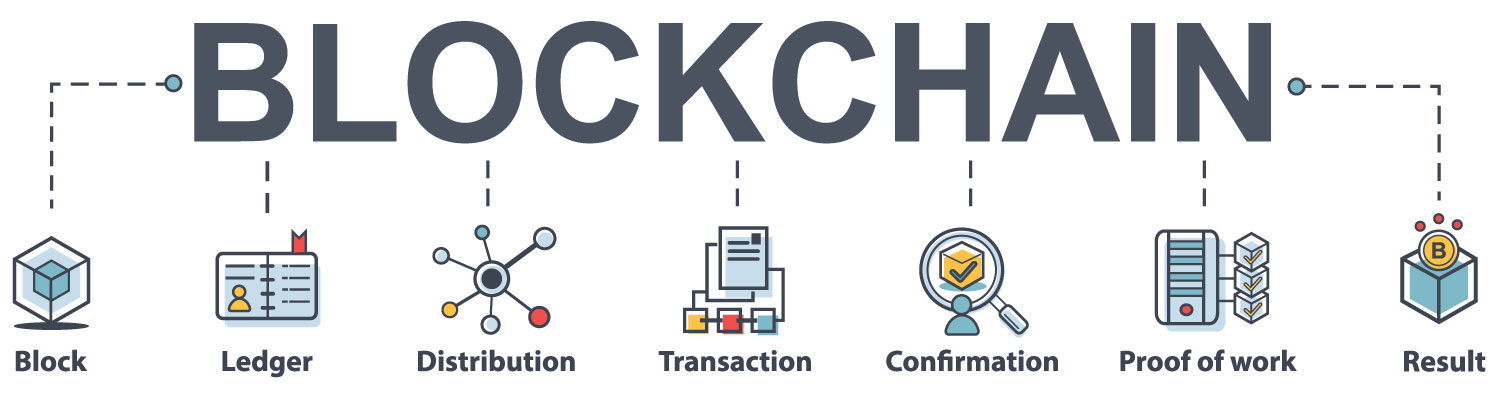
If the data stored of the network of blockchain is legitimate or is in valid state on majority of distributed nodes, the data can be perceived as legitimate. This trust is the key achieved by replicating the datastore on a number of peers(hosts) on the internet. If one of misbehaving peers goes in an invalid state, the other peers can filter it out. As long as there is a majority of peers agreeing to a common valid state, you can completely trust the data that is stored on that system. This replication also guarantees high availability.

**TERMINOLOGY:**

The data store that keeps all blockchain data is called a **ledger**. Each entry in the ledger is called a **transaction**. The transactions are timestamped and stored in groups of **blocks**.

On the ledger you can only add a new transaction or read the old ones. You can not update or delete anything.

The **ledger** is designed to be immutable using cryptographic algorithms and its distributed design. The **ledger** is replicated and maintained by multiple **hosts** or **peers** or **nodes**. This is why its called **DLT** or **Distributed Ledger Technology.**



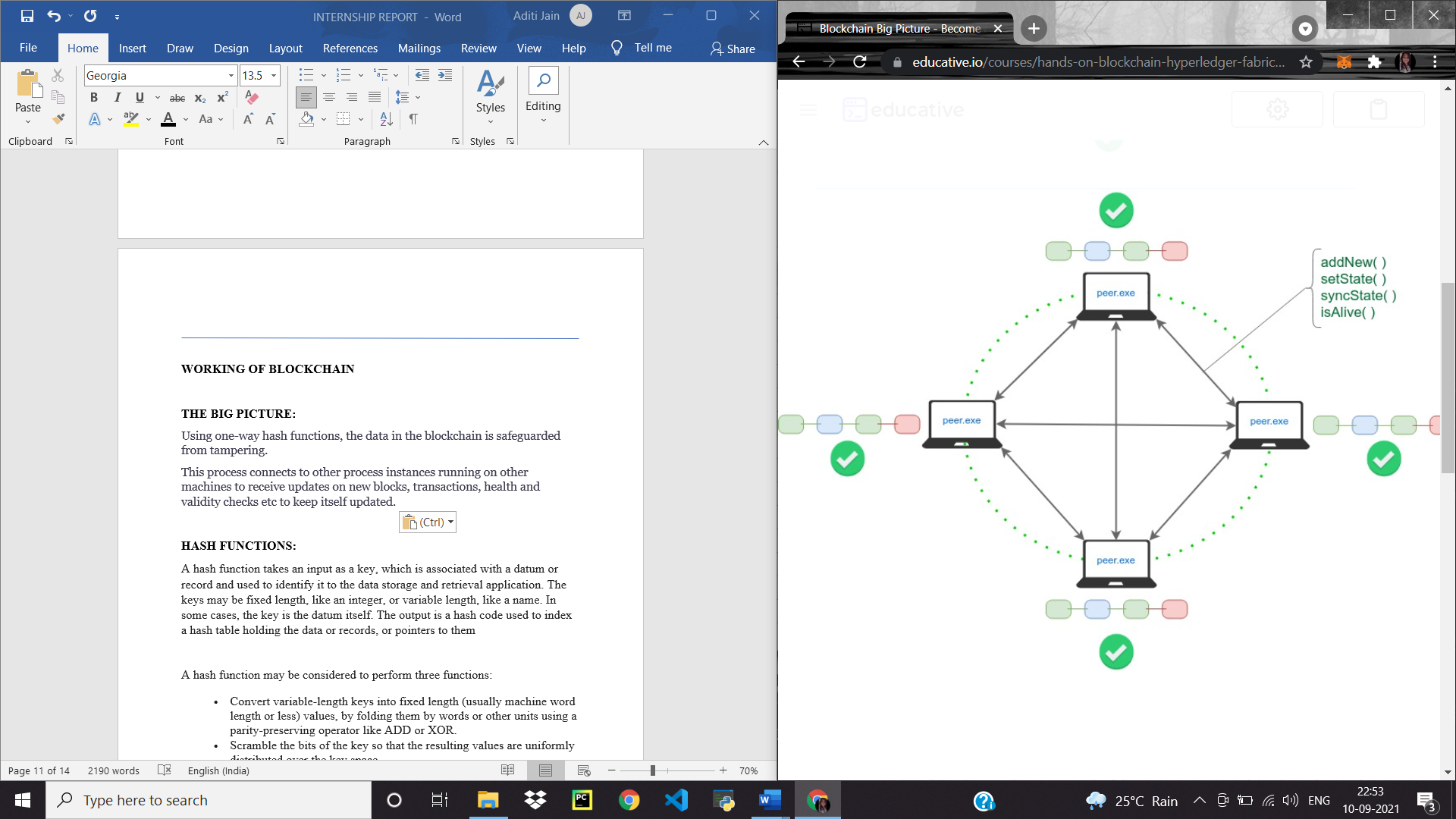
**CHAPTER-2**

**WORKING OF BLOCKCHAIN**

**THE BIG PICTURE:**

Using one-way hash functions, the data in the blockchain is safeguarded from tampering.

This process connects to other process instances running on other machines to receive updates on new blocks, transactions, health and validity checks etc to keep itself updated.



A transaction/block can be appended by any peer and is then broadcast-ed to all peers

Since multiple peers are adding transactions/blocks simultaneously, the consensus protocol along with the ledger implementation ensures “validity” and ordering of transactions in blocks forming the blockchain

**PART-1**

**HASH FUNCTIONS:**

A hash function takes an input as a key, which is associated with a datum or record and used to identify it to the data storage and retrieval application. The keys may be fixed length, like an integer, or variable length, like a name. In some cases, the key is the datum itself. The output is a hash code used to index a hash table holding the data or records, or pointers to them

A hash function may be considered to perform three functions:

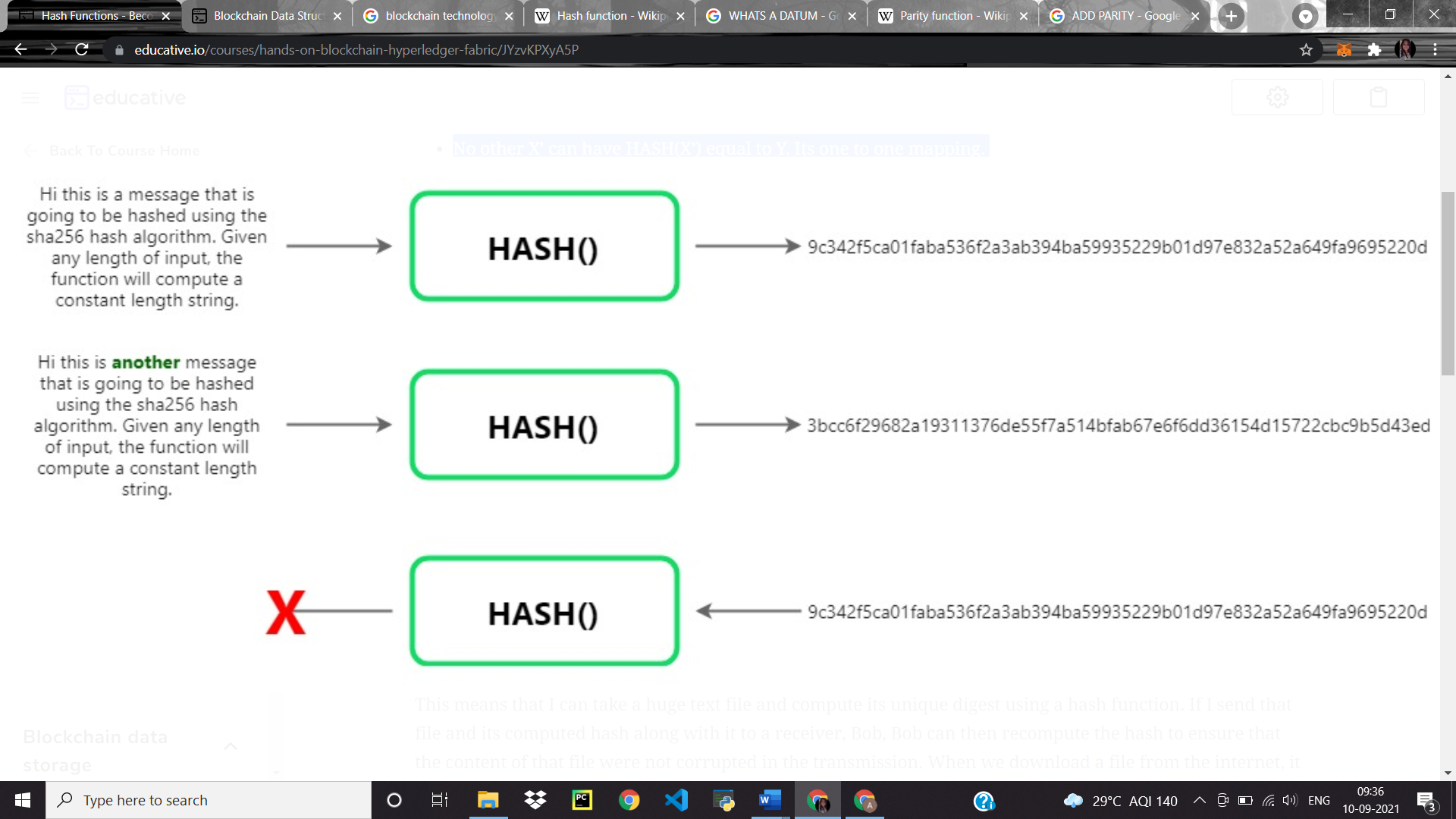
* Convert variable-length keys into fixed length (usually machine word length or less) values, by folding them by words or other units using a parity-preserving operator like ADD or XOR.
* Scramble the bits of the key so that the resulting values are uniformly distributed over the key space.
* Map the key values into ones less than or equal to the size of the table

A hash value for a data is X is a function:

HASH(X) = Y

Such that:

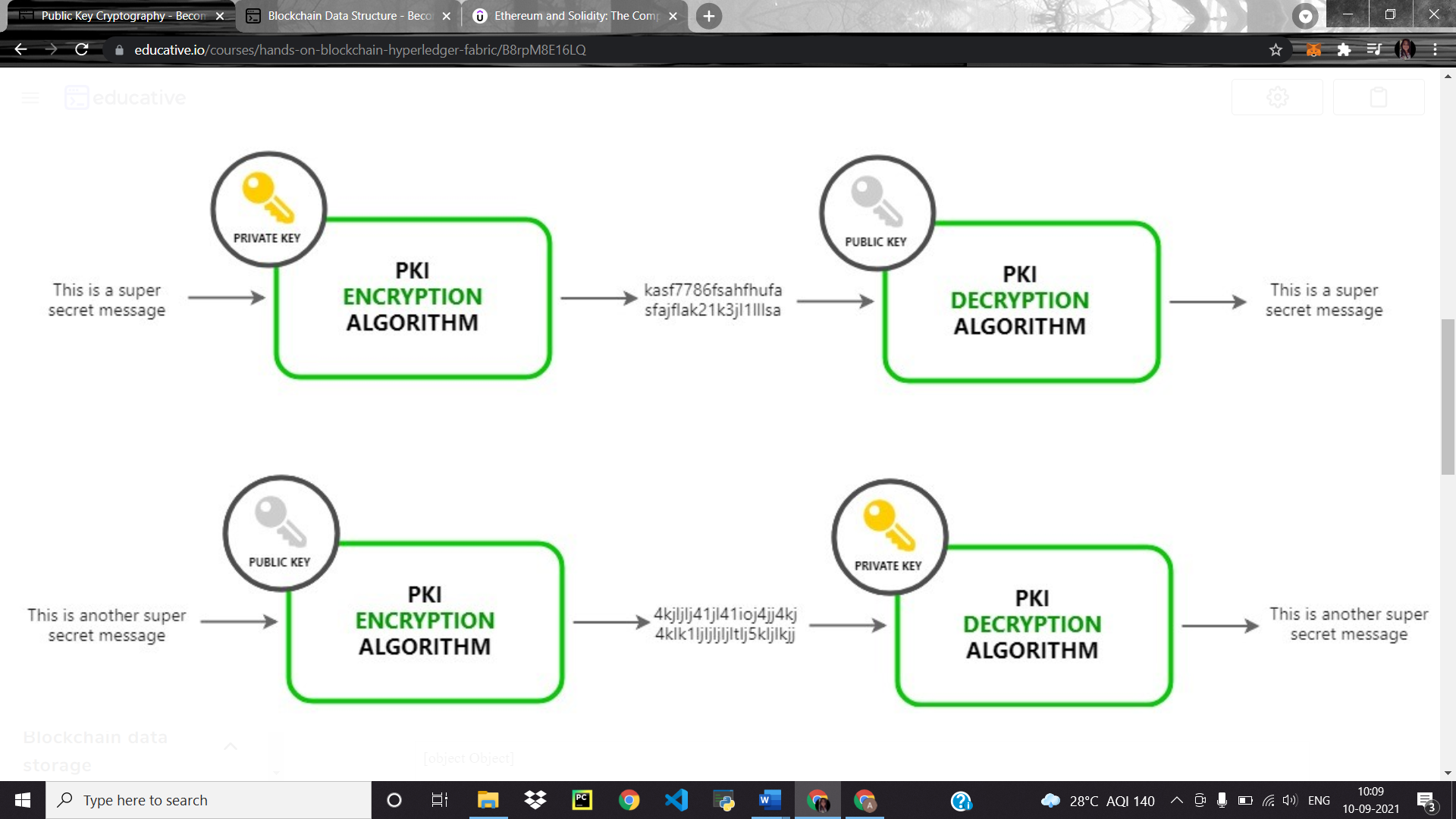
* No other X’ can have HASH(X’) equal to Y. It’s a one to one mapping.
* The size of Y is fixed and the size of X can be arbitrary.
* Given Y you cannot calculate X. It’s a one-way function



**PART-2**

**PUBLIC KEY CRYPTOGRAPHY:**

Public key cryptography is a cryptographic system used for the encryption/decryption of data. You start by generating a special, related pair of keys. These keys can be generated in a pair only. They are output from single execution of a key generation algorithm.



**TRANSMITTING MESSAGE USING SECRET KEY:**

1. Each participant generates their own key pairs.
2. Each party **guards their private key** and **broadcasts the public key** on the network
3. For a private message the sender will encrypt the message with receiver’s public key and share through the network.
4. Since the corresponding private key is only known to receiver therefore only she can decrypt it. Chris cannot make any sense of the encrypted message.

**PART-3**

**ADDING A DIGITAL SIGNATURE:**

MESSAGE

ENCRYPTION

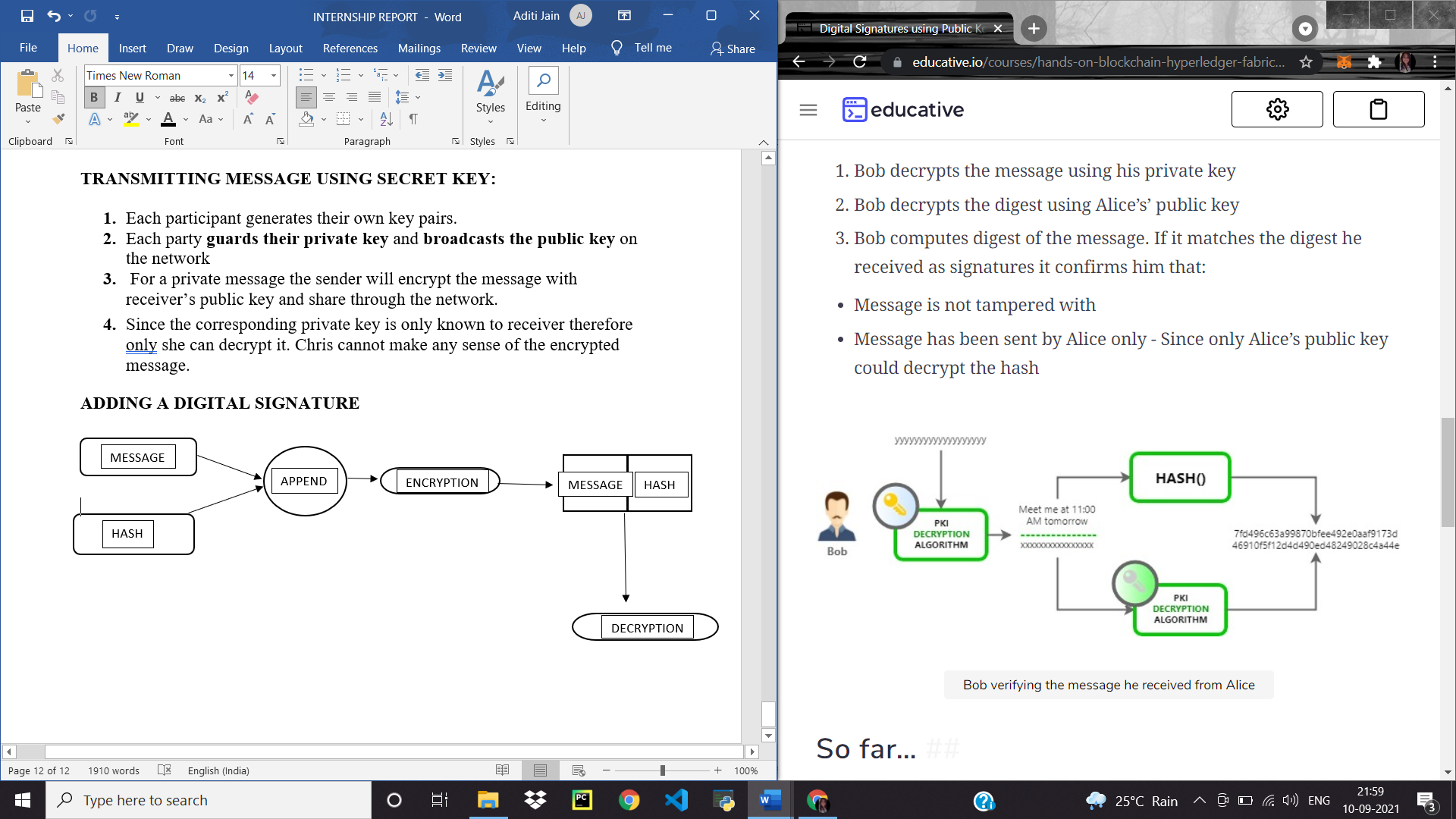
APPEND

HASH

MESSAGE

HASH

HASHED



HASH

COMPARE

DECRYPTION

The obtained hash value should match the value of message hashed signifies that the message received is legitimate and has not been tampered with.

1. Sender computes a message digest by hashing the message she is about to send.
2. Sender calculates the message digest = HASH(message);
3. Sender signs the message digest by encrypting it with her private key.
4. Sender appends the signed digest with message and encrypts with Bob’s public key Alice sends it over to Bob

VERIFYING

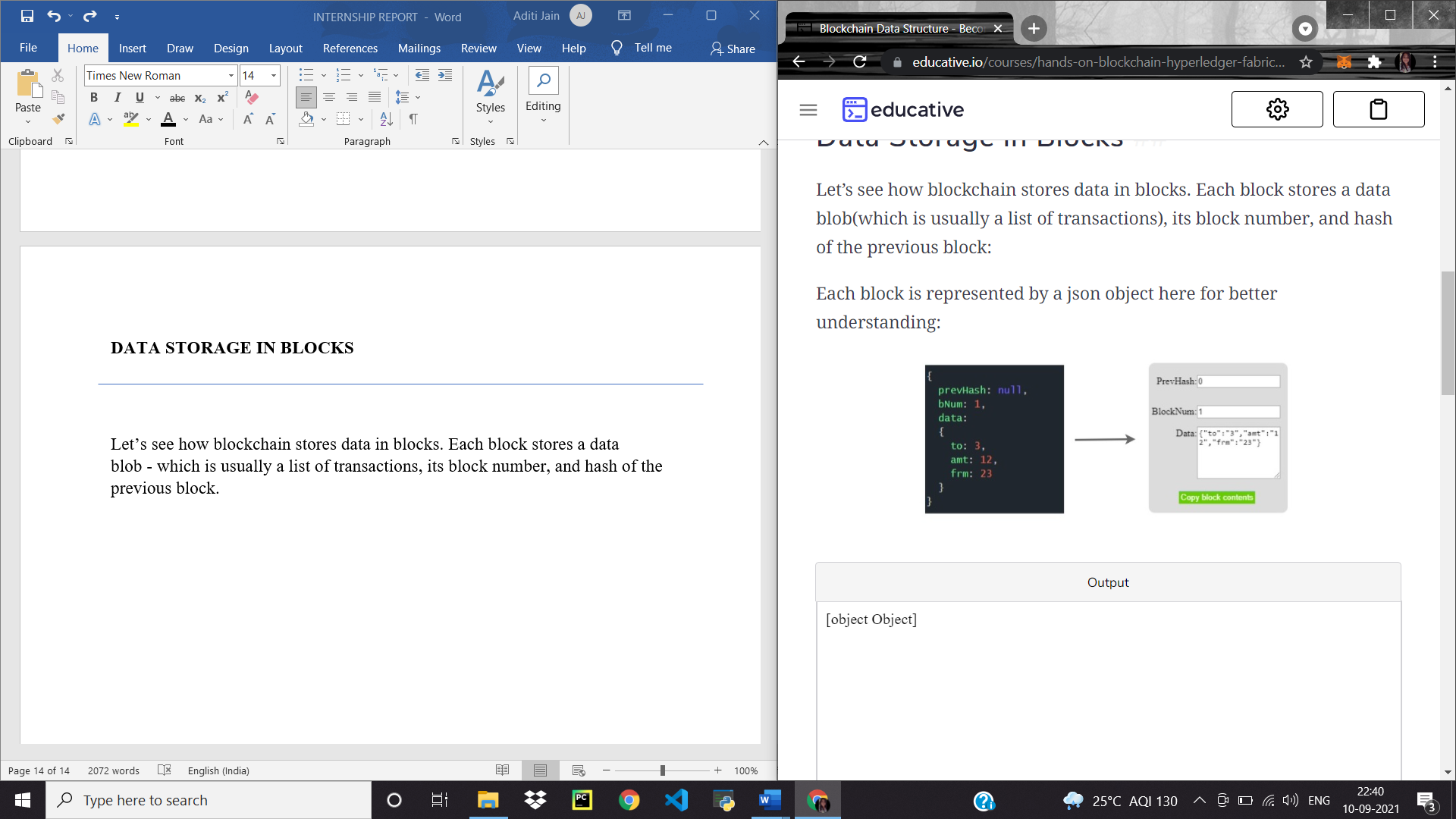
1. Receiver decrypts the message using his private key
2. Receiver decrypts the digest using sender’s public key
3. Receiver computes digest of the message. If it matches the digest he received as signatures it confirms him that:

This message is digitally signed!

**PART-4**

**DATA STORAGE IN BLOCKS**

Let’s see how blockchain stores data in blocks. Each block stores a data blob – which is usually a list of transactions, its block number, and hash of the previous block.



Each block is represented by a json object here for better understanding

When creating a new block, the hash of the previous block is calculated and added to the next block. Now if the previous block is altered later in time, the next block will be invalid as the prevHash stored in it will not match the actual hash of previous block.

Using one-way hash functions, the data in the blockchain is safeguarded from tampering.

**PART-5**

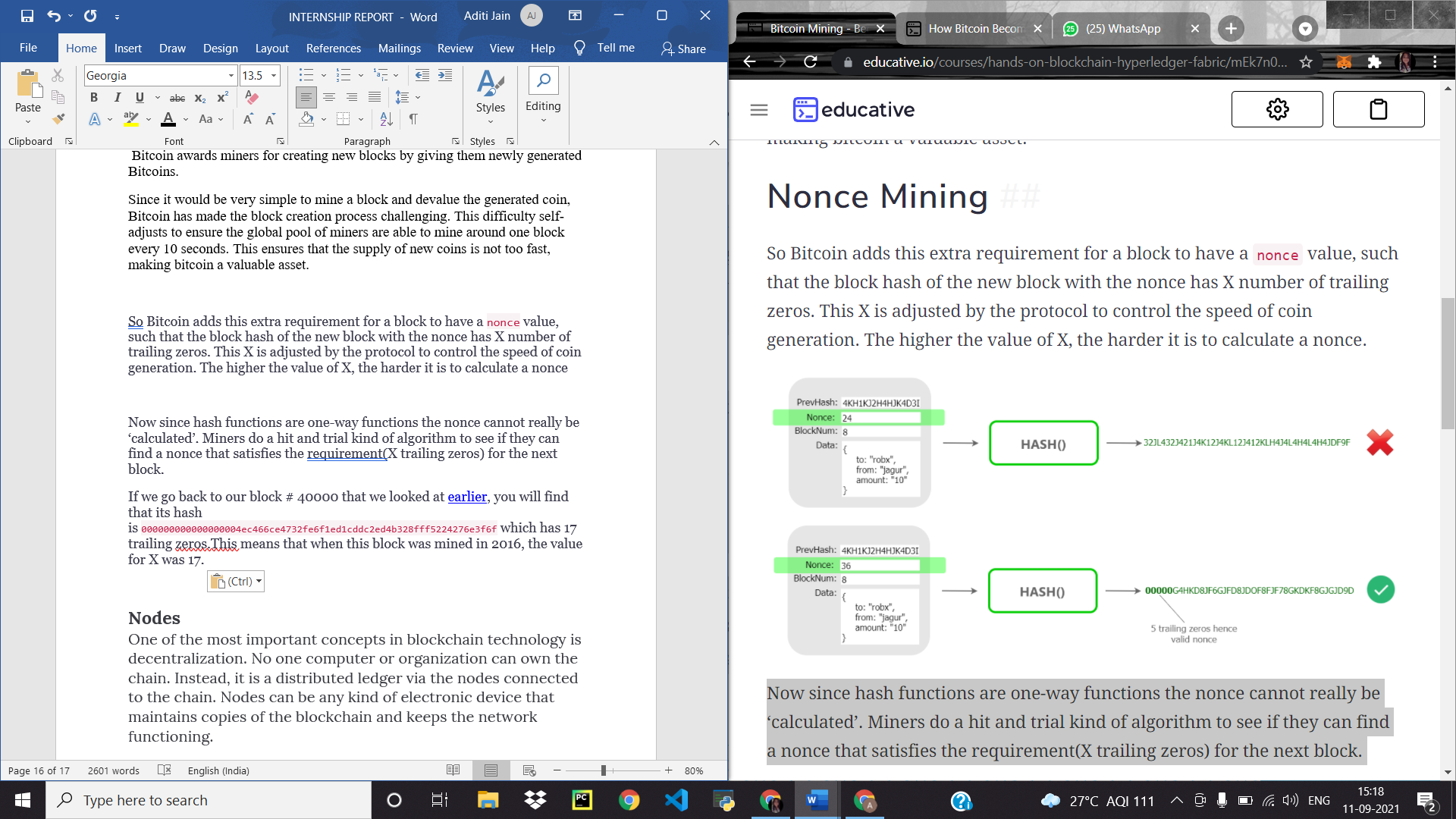
**MINING**

In Bitcoin implementation there is an extra field added in each block, called nonce.

 Bitcoin awards miners for creating new blocks by giving them newly generated Bitcoins.

Since it would be very simple to mine a block and devalue the generated coin, Bitcoin has made the block creation process challenging. This difficulty self-adjusts to ensure the global pool of miners are able to mine around one block every 10 seconds. This ensures that the supply of new coins is not too fast, making bitcoin a valuable asset.

So Bitcoin adds this extra requirement for a block to have a nonce value, such that the block hash of the new block with the nonce has X number of trailing zeros. This X is adjusted by the protocol to control the speed of coin generation. The higher the value of X, the harder it is to calculate a nonce



Now since hash functions are one-way functions the nonce cannot really be ‘calculated’. Miners do a hit and trial kind of algorithm to see if they can find a nonce that satisfies the requirement(X trailing zeros) for the next block.

If we go back to our block # 40000 that we looked at [earlier](https://www.blockchain.com/btc/block/000000000000000004ec466ce4732fe6f1ed1cddc2ed4b328fff5224276e3f6f), you will find that its hash is 000000000000000004ec466ce4732fe6f1ed1cddc2ed4b328fff5224276e3f6f

 which has 17 trailing zeros. This means that when this block was mined in 2016, the value for X was 17

### Nodes

One of the most important concepts in blockchain technology is decentralization. No one computer or organization can own the chain. Instead, it is a distributed ledger via the nodes connected to the chain. Nodes can be any kind of electronic device that maintains copies of the blockchain and keeps the network functioning.

Every node has its own copy of the blockchain and the network must algorithmically approve any newly mined block for the chain to be updated, trusted and verified. Since blockchains are transparent, every action in the ledger can be easily checked and viewed. Each participant is given a [unique alphanumeric identification number](https://hbr.org/2017/01/the-truth-about-blockchain) that shows their transactions.

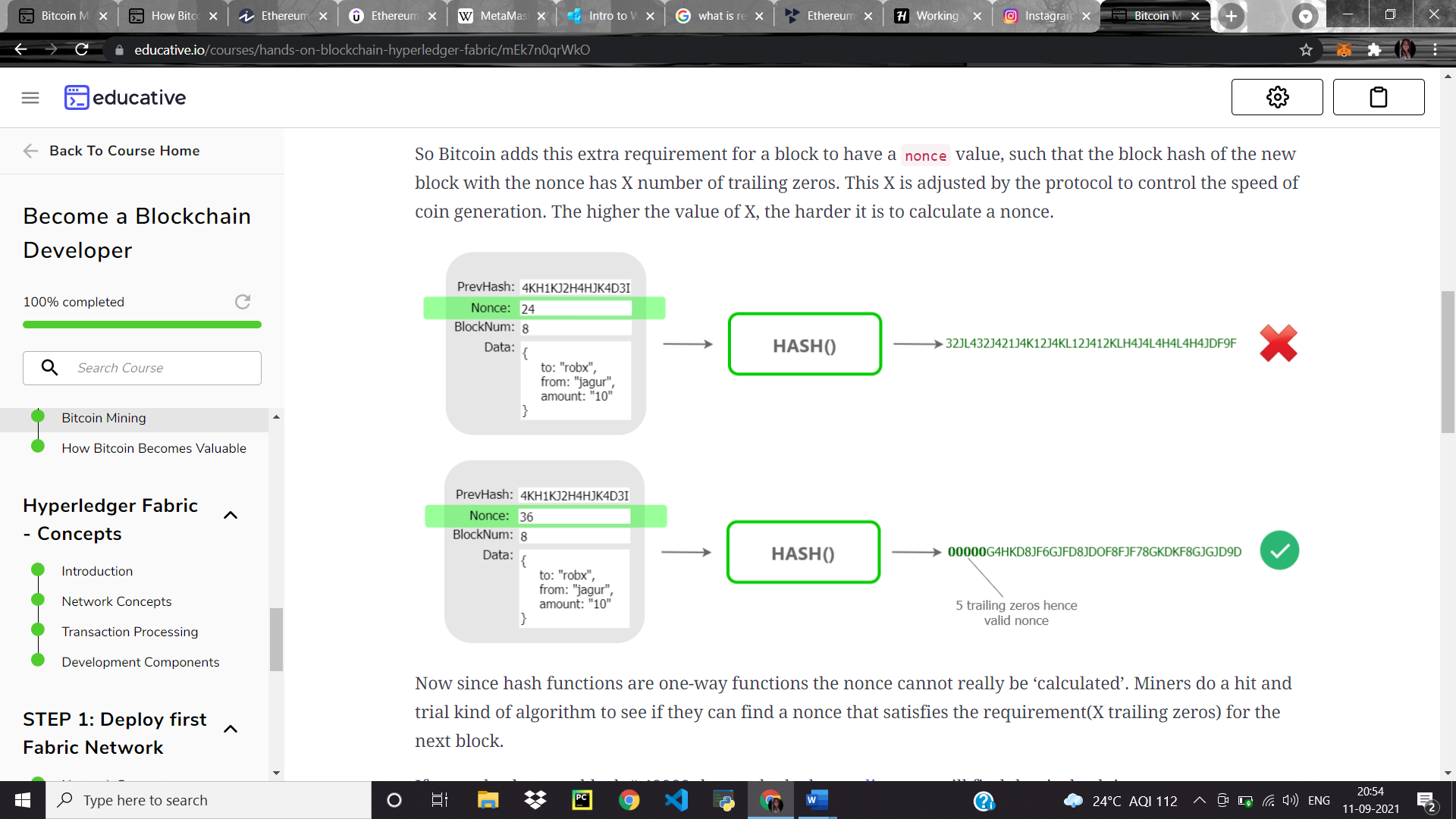
Combining public information with a system of checks-and-balances helps the blockchain maintain integrity and creates trust among users. Essentially, blockchains can be thought of as the scalability of trust via technology.

**Types of nodes**

1. 1. FULL node: They have a complete copy of all the transactions since their genesis. As they possess the complete history of the transactions it has the ability to verify transactions without relying on the other nodes.
2. 2. LIGHTWEIGHT node: Unlike full node they only store the headers of the blocks as they lack storage. These nodes request from the several full nodes for proof if inclusion and can-do simple verifications.
3. 3. MINING node: These nodes compile newly created transactions into blocks. The mining nodes dedicate a part of their computational power in exchange for an economic incentive by executing mining protocol. These protocols involve finding a solution to a mathematical puzzle.

**NONCE MINING**

So Bitcoin adds this extra requirement for a block to have a nonce value, such that the block hash of the new block with the nonce has X number of trailing zeros. This X is adjusted by the protocol to control the speed of coin generation. The higher the value of X, the harder it is to calculate a nonce.



Now since hash functions are one-way functions the nonce cannot really be ‘calculated’. Miners do a hit and trial kind of algorithm to see if they can find a nonce that satisfies the requirement (X trailing zeros) for the next block.

If we go back to our block # 40000 that we looked at [earlier](https://www.blockchain.com/btc/block/000000000000000004ec466ce4732fe6f1ed1cddc2ed4b328fff5224276e3f6f), you will find that its hash is 000000000000000004ec466ce4732fe6f1ed1cddc2ed4b328fff5224276e3f6f which has 17 trailing zeros. This means that when this block was mined in 2016, the value for X was 17.

**PART-6**

**Applications of Blockchain Technology:**

Blockchain finds a wide range of applications in many sectors of which some are listed below:

* 1. 1. Financial Services
  2. 2. Healthcare
  3. 3. -Asset Management: Trade Processing and Settlement
  4. 4. Internet of Things (IOT)
  5. 5. Insurance

**CHAPTER-3**

**PROBLEM STATEMENT:**

This project implements transportation of Gasoline between a Producer and a Distributor where an order is initiated by a Distributor and accepted by the producer if sufficient stock is available. During transportation, quality and quantity check is maintained with the help of sensors. If specifications are not met and the quality deteriorates, then payment amount decreases accordingly

**CHAPTER-4**

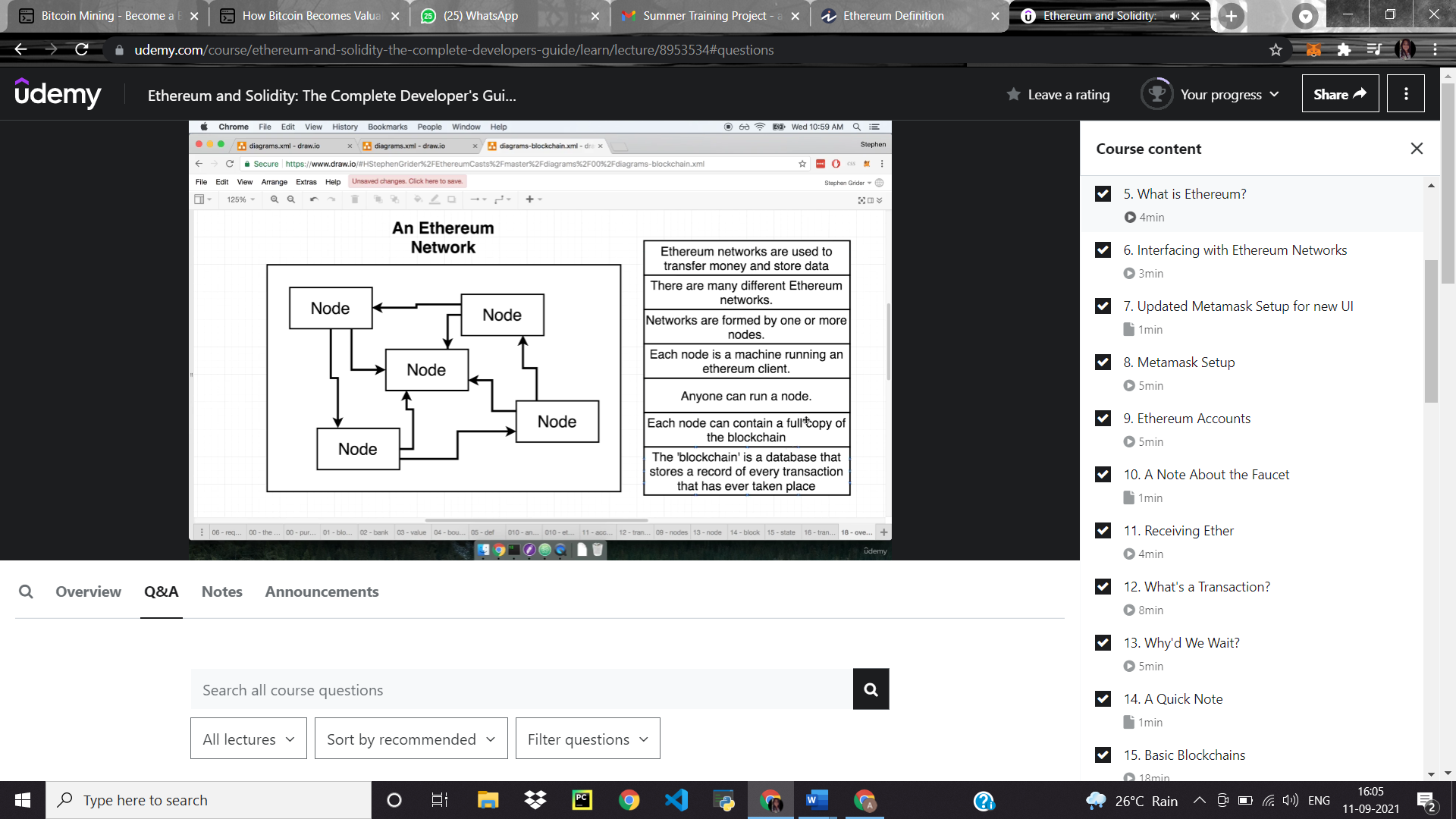
**ETHEREUM BLOCKCHAIN**

As a blockchain network, Ethereum is a decentralized public ledger for verifying and recording transactions. The network's users can create, publish, monetize, and use applications on the platform, and use its Ether cryptocurrency as payment. Insiders call the decentralized applications on the network "dApps."

As a cryptocurrency, Ethereum is second in market value only to Bitcoin, as of May 2021.

Ethereum was launched in July 2015 by a small group of blockchain enthusiasts. They included Joe Lubin, founder of ConsenSys, a blockchain applications developer that uses the Ethereum network. Another co-founder, Vitalik Buterin, is credited with originating the Ethereum concept and now serves as its CEO and public face. Buterin is sometimes described as the world's youngest crypto billionaire. (He was born in 1994.)

* Ethereum is an open-source blockchain-based platform that creates and shares business, financial services, and entertainment applications.
* Ethereum users pay fees to use dApps. The fees are called "gas" because they vary depending on the amount of computational power required.
* Ethereum has its own associated cryptocurrency, Ether or ETH.
* Its cryptocurrency is now second only to Bitcoin in market value.



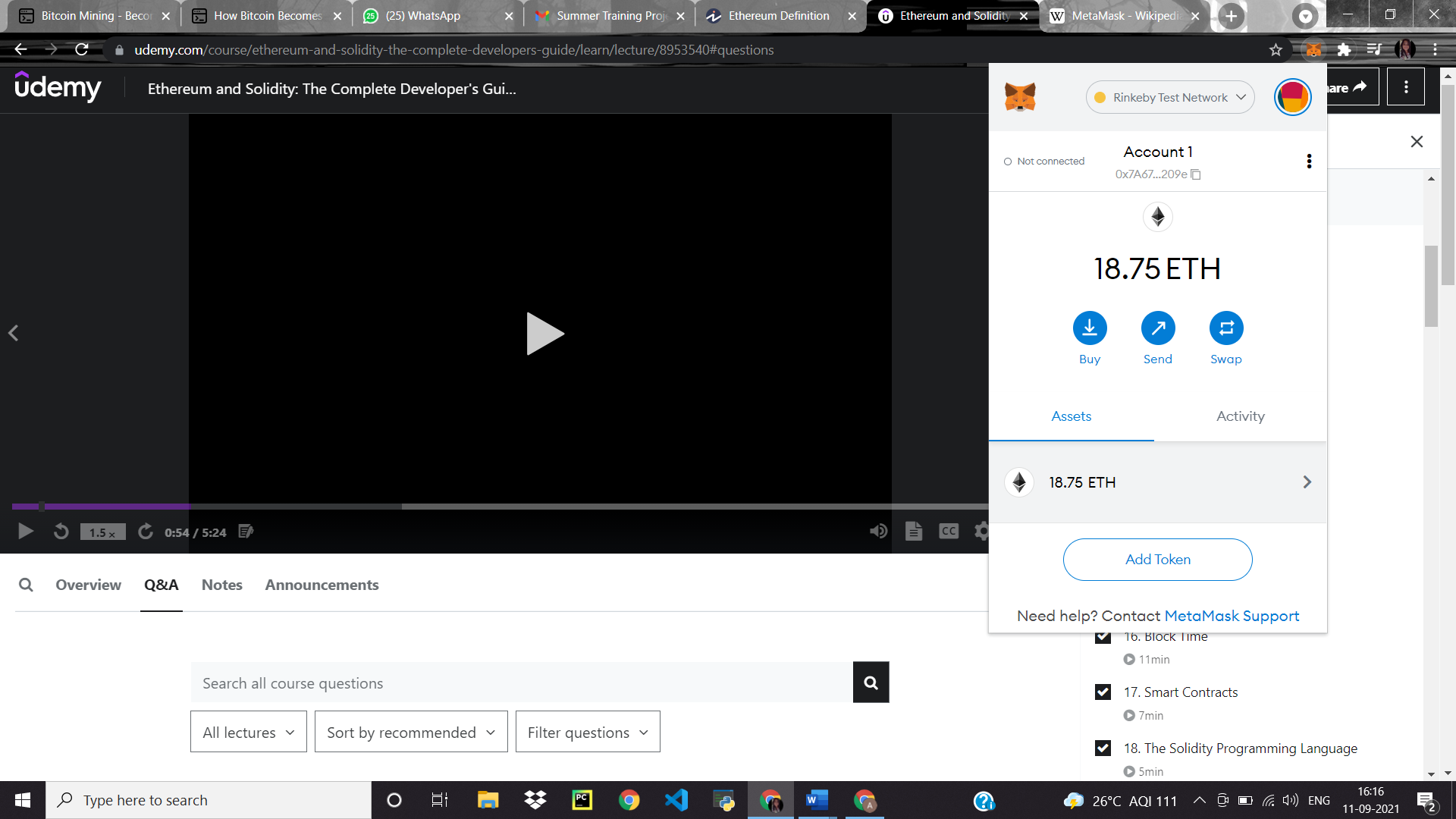
**PART-1**

**INTERFACING WITH ETHEREUM NETWORKS**

**MetaMask** is a [software](https://en.wikipedia.org/wiki/Software) [cryptocurrency wallet](https://en.wikipedia.org/wiki/Cryptocurrency_wallet) used to interact with the [Ethereum](https://en.wikipedia.org/wiki/Ethereum) [blockchain](https://en.wikipedia.org/wiki/Blockchain). It allows users to access their Ethereum wallet through a [browser extension](https://en.wikipedia.org/wiki/Browser_extension) or [mobile app](https://en.wikipedia.org/wiki/Mobile_app), which can then be used to interact with [decentralized applications](https://en.wikipedia.org/wiki/Decentralized_application).

MetaMask was created in 2016 by [ConsenSys](https://en.wikipedia.org/wiki/ConsenSys), a [blockchain](https://en.wikipedia.org/wiki/Blockchain) [software company](https://en.wikipedia.org/wiki/Software_company) focusing on [Ethereum](https://en.wikipedia.org/wiki/Ethereum)-based tools and infrastructure.[[7]](https://en.wikipedia.org/wiki/MetaMask#cite_note-Newsweek_2018-7)[[8]](https://en.wikipedia.org/wiki/MetaMask#cite_note-8)

Up until 2019, MetaMask had only been available as a desktop [browser extension](https://en.wikipedia.org/wiki/Browser_extension) for [Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome) and [Firefox](https://en.wikipedia.org/wiki/Firefox) browsers.



The overview of the blockchain is analogous to a conventional machine which takes input, processes it and then produces output. Ethereum starts from a blank state (also called a genesis state), processes transactions and reaches a final state. If we need to make a transition from one state to the other, the transaction must be valid. Transactions are grouped into blocks such that each block contains many transactions and is chained to the previous block. Every computations that occurs due to a transaction incurs a fee . This fee is paid in a denomination called “gas”.

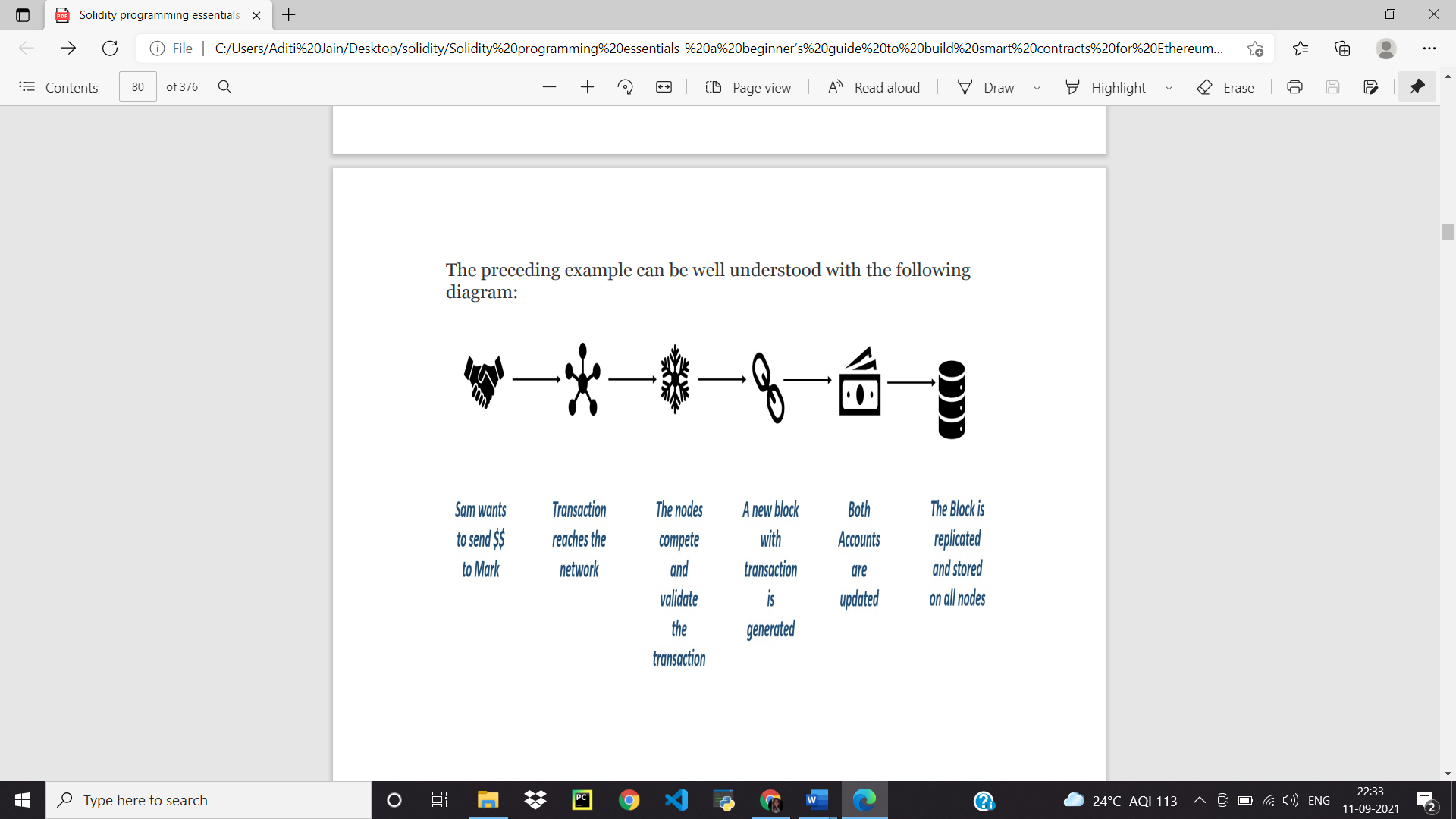
**PART-2**

**WHAT IS A TRANSACTION?**

A transaction is an agreement between a buyer and a seller, a supplier and a consumer, or a provider and a consumer that there will be an exchange of assets, products, or services for currency, cryptocurrency, or some other asset, either in the present or in the future. Ethereum helps in executing the transaction. Following are the three types of transactions that can be executed in Ethereum:

**A transaction consists of following properties.**

1. **NONCE:** How many times a sender has sent a transaction.
2. **TO:** Address of account this money is going to,
3. **VALUE:** Amount of ether to send to the target address
4. **GASPRICE:** Amount of ether a sender is willing to pay per unit gas to get this transaction processed.
5. **START GAS/ GAS LIMIT:** Units of gas that this transaction can consume.
6. **V, R, S:** Cryptographic pieces of data that can be used to generate the sender’s account address. Generated from sender’s private key.



**PART-3**

**CRYPTOGRAPHY**

Cryptography is the science of converting plain simple text intosecret, hidden, meaningful text, and vice-versa. It also helps in transmitting and storing data that cannot be easily deciphered using owned keys. There are the following two types of cryptography in computing: Symmetric Asymmetric

**SYMMETRIC ENCRYPTION AND DECRYPTION**

Symmetric cryptography refers to the process of using a single key for both encryption and decryption. It means the same key should be available to multiple people if they want to exchange messages using this form of cryptography.

**ASYMMETRIC ENCRYPTION AND DECRYPTION**

Asymmetric cryptography refers to the process of using two keys for encryption and decryption. Any key can be used for encryption and decryption. Message encryption with a public key can be decrypted using a private key and messages encrypted by a private key can be decrypted using a public key. Let's understand this with the help of an example.

**CHAPTER-5**

**SOLIDITY, WEB 3JS, REMIX**

**SOLIDITY**

Solidity is known as a contract-based, high-level programming language. This platform has similar syntax to the scripting language of JavaScript. Solidity as a programming language is made to enhance the Ethereum Virtual Machine. Solidity is statically typed scripting language which does the process of verifying and enforcing the constraints at compile-time as opposed to run-time.

Compared to the other languages which compile on the EVM, Solidity contains a number of important advantages:

* 1. 1. [**Complex member variables**](https://en.wikipedia.org/wiki/Member_variable) for contracts are supported. For example, hierarchical mappings and structures.
  2. 2. Contracts support [inheritance](https://en.wikipedia.org/wiki/Inheritance_(object-oriented_programming)), including multiple inheritance.
  3. 3. An [**Application Binary Interface (ABI)**](https://docs.google.com/document/d/1EKyByeoiEUuFq91elswUzV720WjcSIqD_Q_EWTXwddE/edit#heading%3Dh.a5rjosbdrj1b) facilitating multiple type-safe functions within a single contract is supported in Solidity. Type safety means that the compiler will validate datatypes while compiling, and throw an error if you try to assign the wrong datatype to a variable. This feature is now supported by Serpent as well.

**WEB3 JS**

[Web3.js](https://web3js.readthedocs.io/en/1.0/) enables you to fulfill the responsibility of developing clients that interact with The Etherem Blockchain. It is a collection of libraries that allow you to perform actions like send Ether from one account to another, read and write data from smart contracts, create smart contracts, and so much more!

The web3 library is an open-source JavaScript library that can be used to connect to Ethereum nodes from the same or a remote computer. It allows IPC as well as RPC to connect to Ethereum nodes. web3 is a client-side library and can be used alongside a web page and query and can submit transactions to Ethereum nodes. It can be installed using the node package manager as a node module like the Solidity compiler.

**REMIX**

Remix IDE is an open source web and desktop application. It fosters a fast development cycle and has a rich set of plugins with intuitive GUIs. Remix is used for the entire journey of contract development as well as being a playground for learning and teaching Ethereum.

**CHAPTER-6**

**SMART CONTRACT**

A smart contract is custom logic and code deployed and executed within an Ethereum virtual environment. Smart contracts are digitized and codified rules of transaction between accounts. Smart contracts help in transferring digital assets between accounts as an atomic transaction. Smart contracts can store data. The data stored can be used to record information, facts, associations, balances, and any other information needed to implement logic for real-world contracts. Smart contracts are very similar to object-oriented classes. A smart contract can call another smart contract just like an object-oriented object can create and use objects of another class. Think of smart contracts as a small program consisting of functions. You can create an instance of the contract and invoke functions to view and update contract data along with the execution of some logic.

**HOW ARE CONTRACTS DEPLOYED?**

Remix makes deployment of contracts a breeze; however, it is performing a lot of steps behind the scenes. It is always useful to understand the process of deploying contracts to have finer control over the deployment process. The first step is the compilation of contracts. The compilation is done using the Solidity compiler.

The compiler generates the following two major artifacts:

* ABI definition
* Contracts bytecode

Think of the Application Binary Interface (ABI) as an interface consisting of all external and public function declarations along with their parameters and return types. The ABI defines the contract and any caller wanting to invoke any contract function can use the ABI to do so.

The bytecode is what represents the contract and it is deployed in the Ethereum ecosystem. The bytecode is required during deployment and ABI is needed for invoking functions in a contract.

A new instance of a contract is created using the ABI definition.

Deploying a contract itself is a transaction. A transaction is created for deploying the contract on Ethereum. The bytecode and ABI are necessary inputs for deploying a contract.

As any transaction execution costs gas in Ethereum, appropriate quantity of gas should be supplied while deploying the contract. As and when the transaction is mined, the contract is would be available for interaction through contract address.

Using the newly generated address, callers can invoke functions within the contract.

**CHAPTER-7**

**CONCLUSION**

Solidity contracts can be debugged using the Remix editor. Remix provides a convenient way to author and debug contracts by verifying variables and code execution at every step. It helps us move forward and back in code execution. It provides breakpoints to break the execution of code. There are other ways to debug contracts as well. These include using Block Explorers and Solidity events. Although events and Block Explorers provide limited capabilities for debugging, they are very helpful and facilitate production.

**REFRENCES**

<https://www.ibm.com/blockchain/industries/oil-gas>

educative.io

<https://www.ibm.com/downloads/cas/B4OYMO5Q>

<https://en.wikipedia.org/wiki/Blockchain>

**INTERNSHIP DETAILS:**

Interned at GAIL (India) limited from 2nd August Sunday to 31st August Tuesday under guidance of Ms. Varsha Nirbahvne Dy. General Manager(BIS).