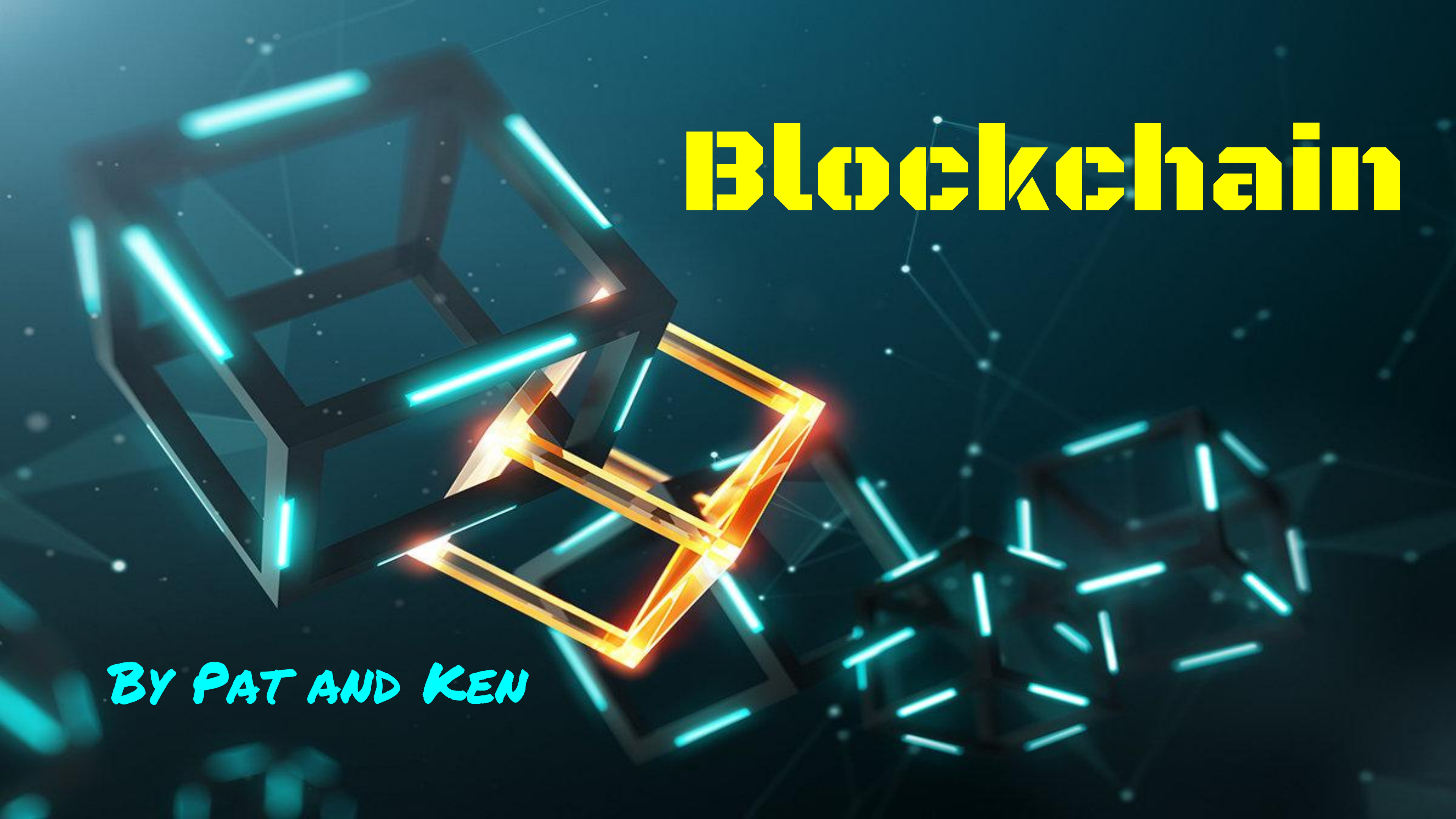
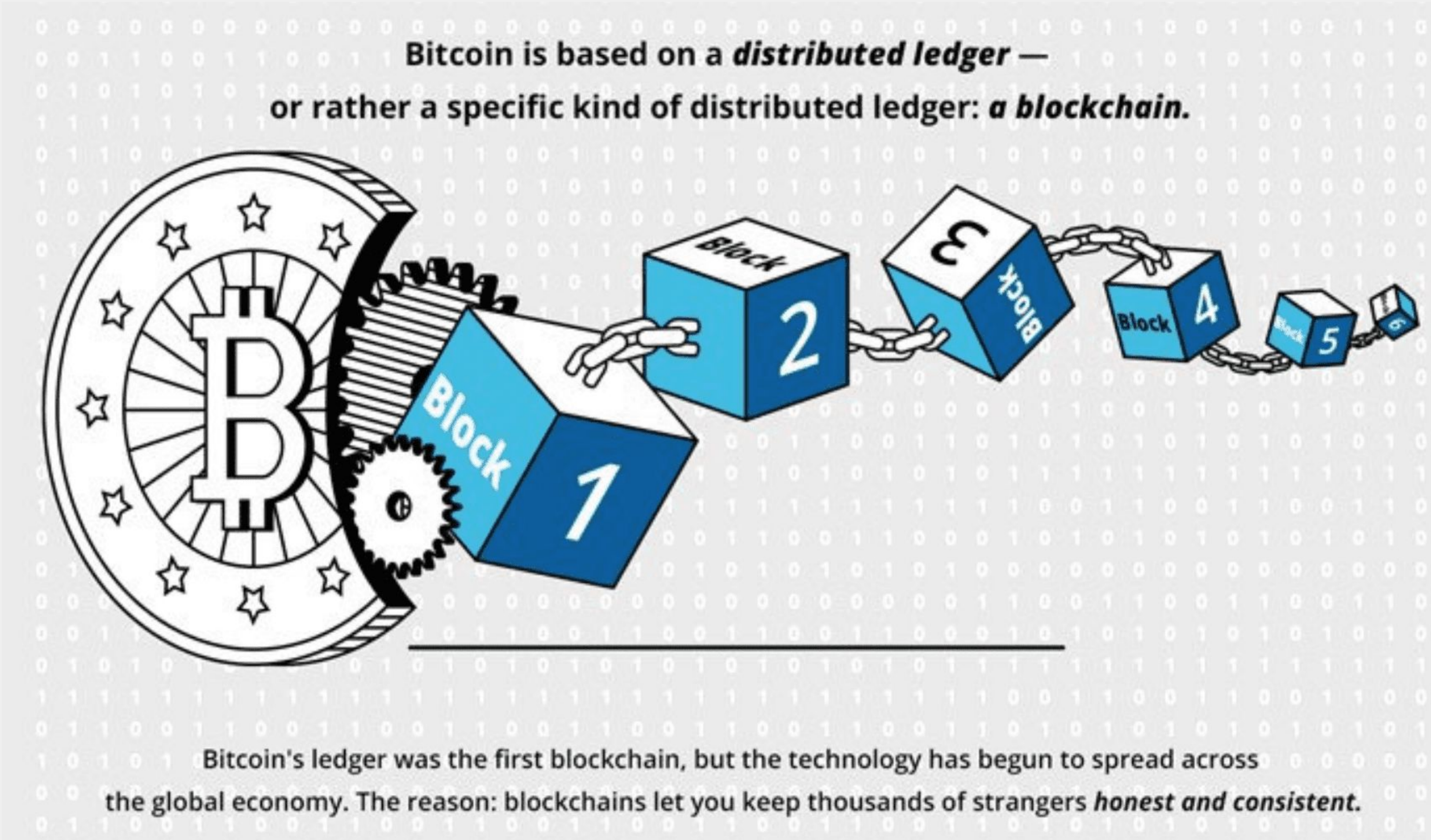


Blockchain

By Pat and Ken

Blockchain == Bitcoin?

Source (and recommended reading!): https://www.investopedia.com/terms/b/blockchain.asp



Bitcoin is based on a ***distributed ledger*** —
or rather a specific kind of distributed ledger: ***a blockchain.***

Bitcoin's ledger was the first blockchain, but the technology has begun to spread across the global economy. The reason: blockchains let you keep thousands of strangers ***honest and consistent***.

Objectives of this presentation:

1. What is blockchain
2. How can we implement it in the CS classroom
3. Bitcoin as a specific application
4. Other Applications

Part 1 - What is blockchain?

What is blockchain?

SOURCE: https://www.investopedia.com/terms/b/blockchain.asp

“The goal of blockchain is to allow digital information to be recorded and distributed, but not edited.”

Blockchain technology was first outlined in 1991 by Stuart Haber and W. Scott Stornetta, two researchers who wanted to implement a system where document timestamps could not be tampered with.

But it wasn’t until almost two decades later, with the launch of Bitcoin in January 2009, that blockchain had its first real-world application.

Prior knowledge: **What is a hash?**

Hashing can be its own lesson(s).

Hash – a number generated from a one way function (**hash function**). The mechanics (**hash algorithm**) of the function are unknown making the output unpredictable. Because it is a function, the same input always gives the same output

Example:

`SHA-256 hashFunction(Mike) =`

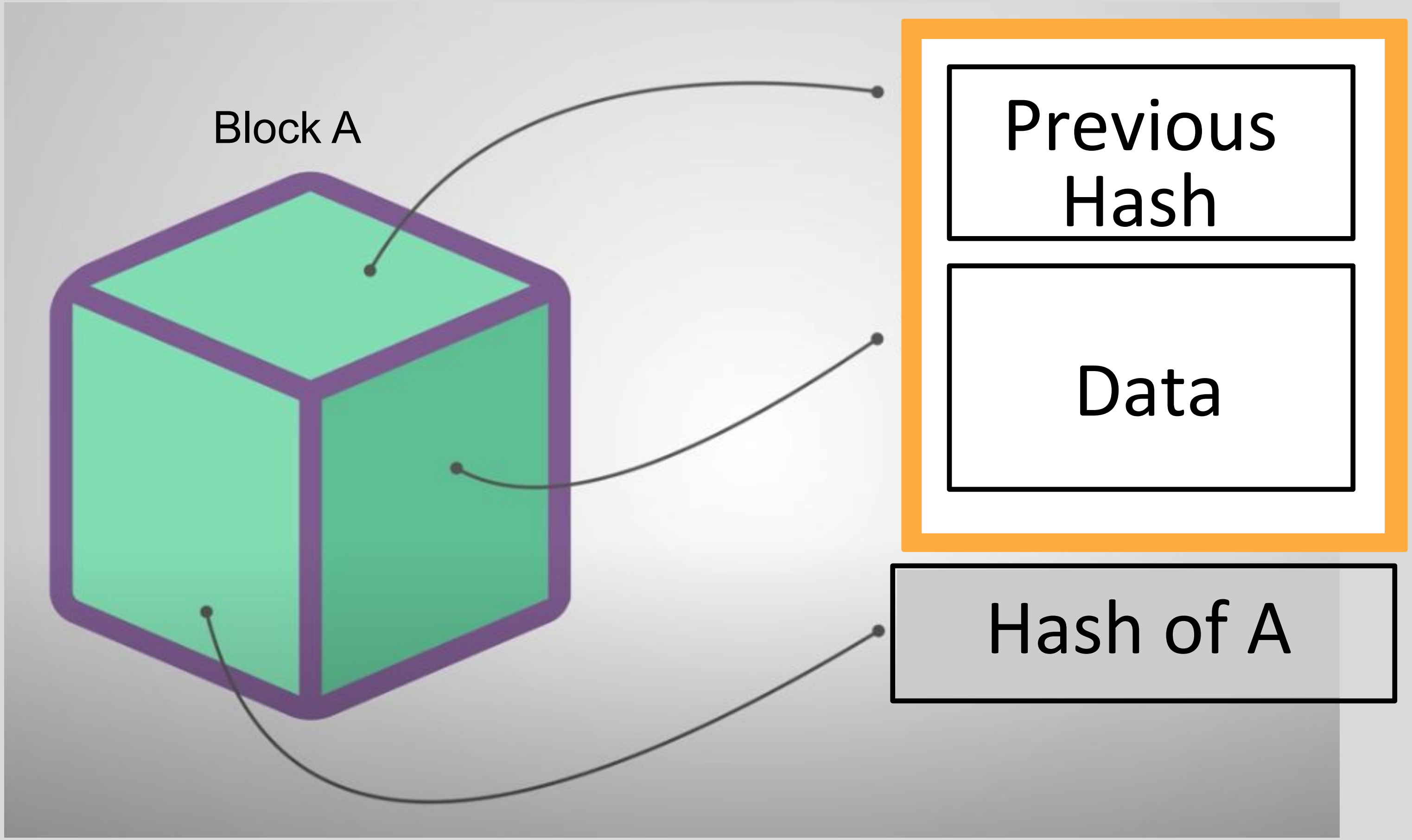
9DC415325A95C6E2558BF141A8772A175DE49B08F0A027C8720AD942D6EC63F7

`SHA-256 hashFunction(Mikez) =`

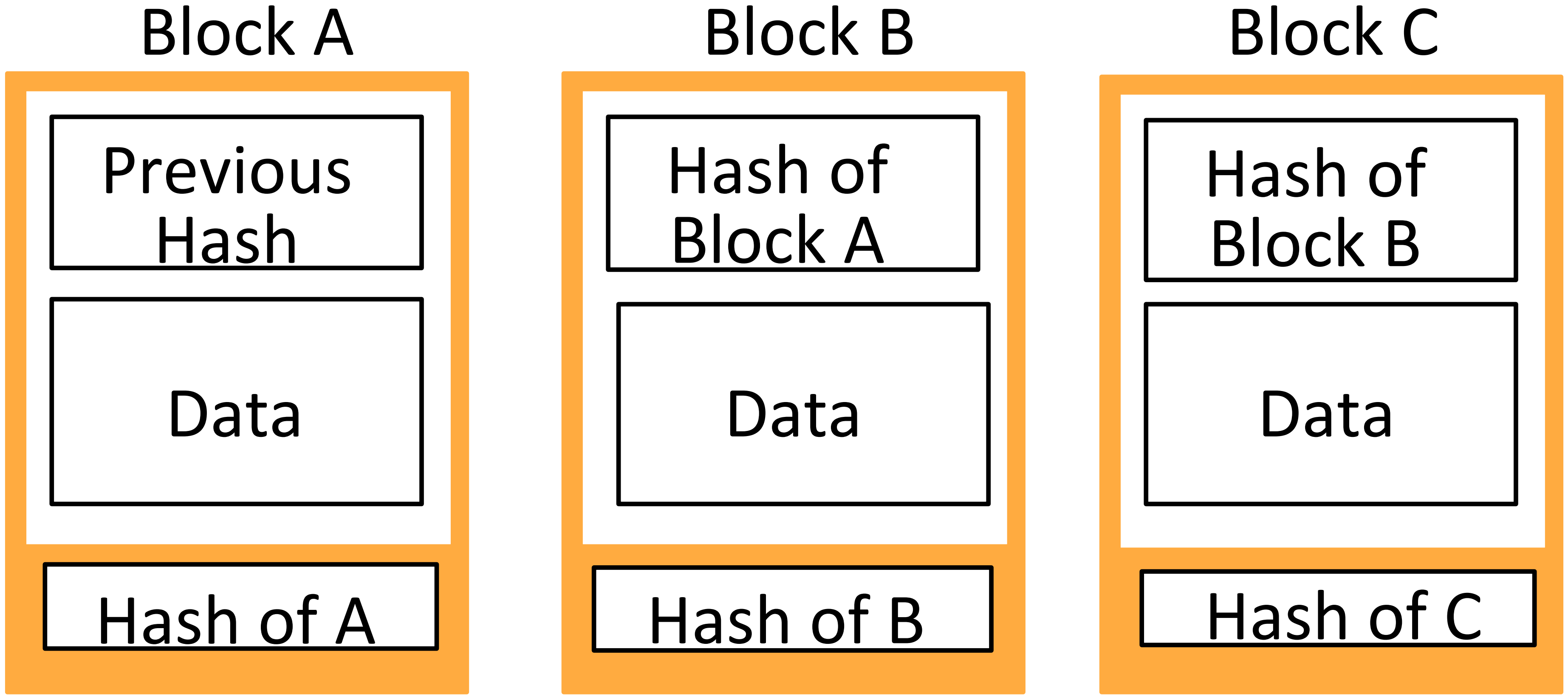
DE4839CC06D8F31C700C5834410845B2942D3336D1428329A875953AF8001DC3

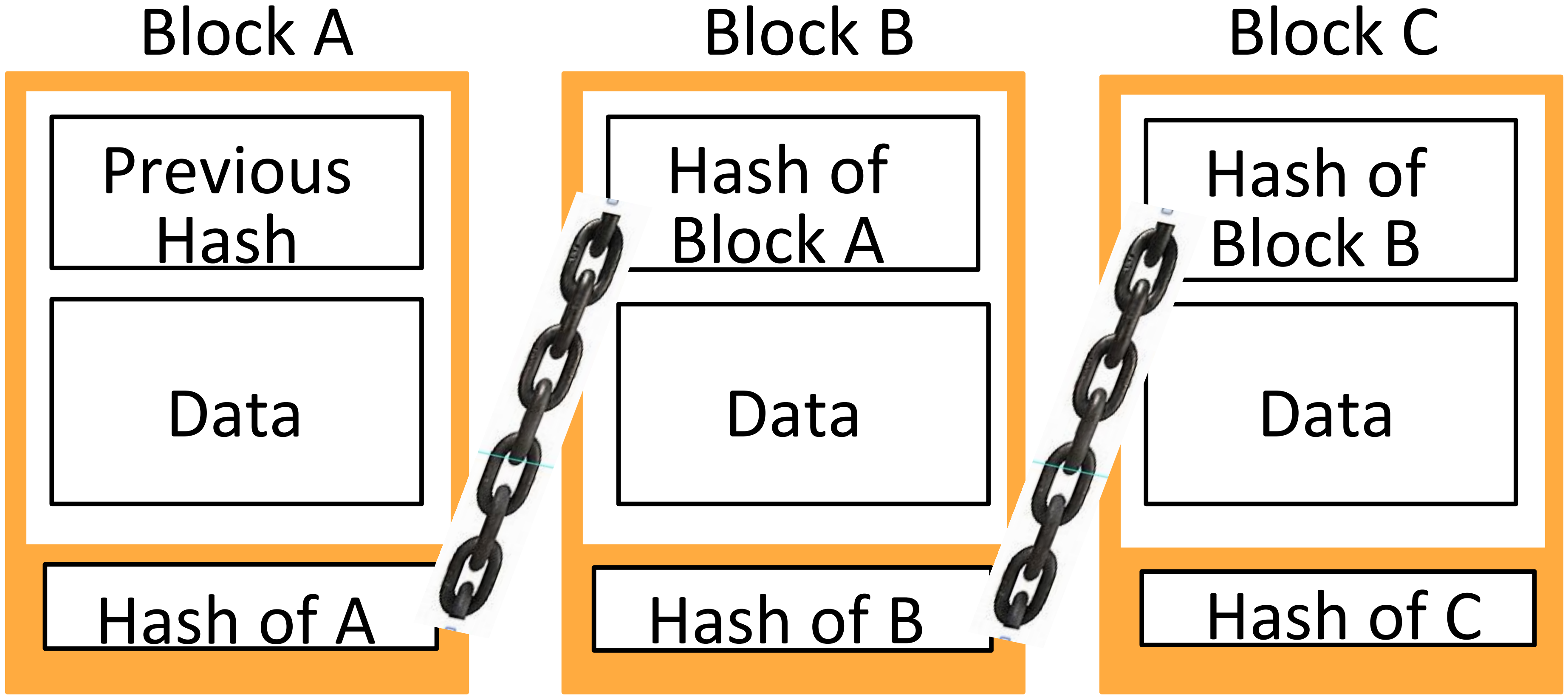
A WEBSITE TO DEMONSTRATE SHA-256 HASHING:
https://passwordsgenerator.net/sha256-hash-generator/

What goes in a block?



So where is the chain?

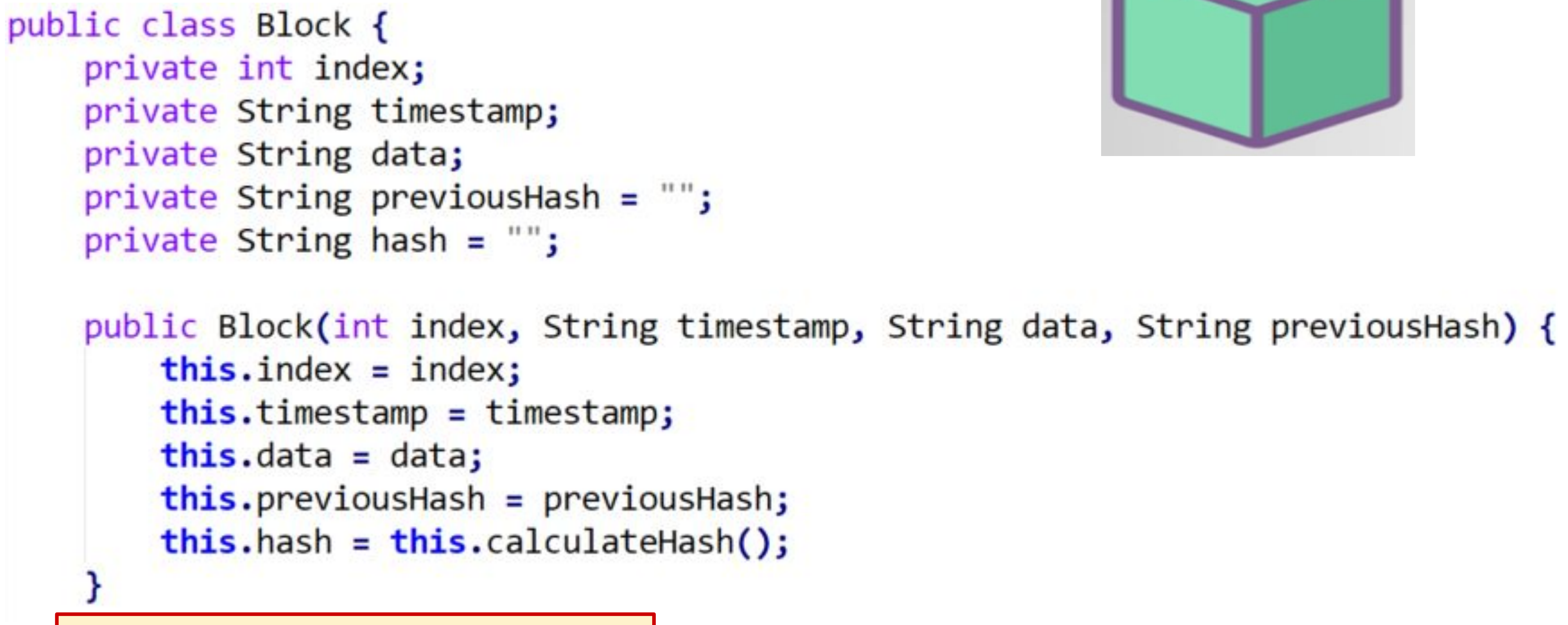



Block A
Previous Hash
Data
Hash of A
Block B
Hash of Block A
Data
Hash of B
Block C
Hash of Block B
Data
Hash of C

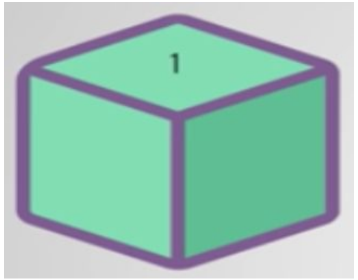
Part 2 - How can we code a blockchain?

Coding Example of B-l-o-c-k-c-h-a-i-n Implementation

File: Block.java



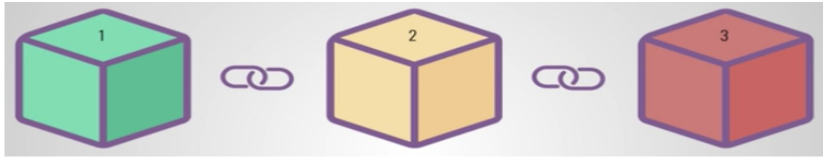
```java
public class Block {
    private int index;
    private String timestamp;
    private String data;
    private String previousHash = "";
    private String hash = "";

    public Block(int index, String timestamp, String data, String previousHash) {
        this.index = index;
        this.timestamp = timestamp;
        this.data = data;
        this.previousHash = previousHash;
        this.hash = this.calculateHash();
    }
```



Once the block is made, nothing can be changed. The block is immutable!

MODIFIED FROM:
https://www.savjee.be/2017/07/Writing-tiny-blockchain-in-JavaScript/

Coding Example of B-l-o-c-k-c-h-a-i-n Implementation



"It's an array!"

"It's an ArrayList!"

"No, it's . . . "

We can only add blocks to the chain

File: Blockchain.java

```
public class Blockchain {

    private LinkedList<Block> chain;

    public Blockchain() {
        this.chain = new LinkedList<Block>();
    }

    public Blockchain(Block firstBlock) {
        this();
        chain.push(firstBlock);
    }

    public void addBlock(Block newBlock) {
        //verify
        this.chain.push(newBlock);
    }
```

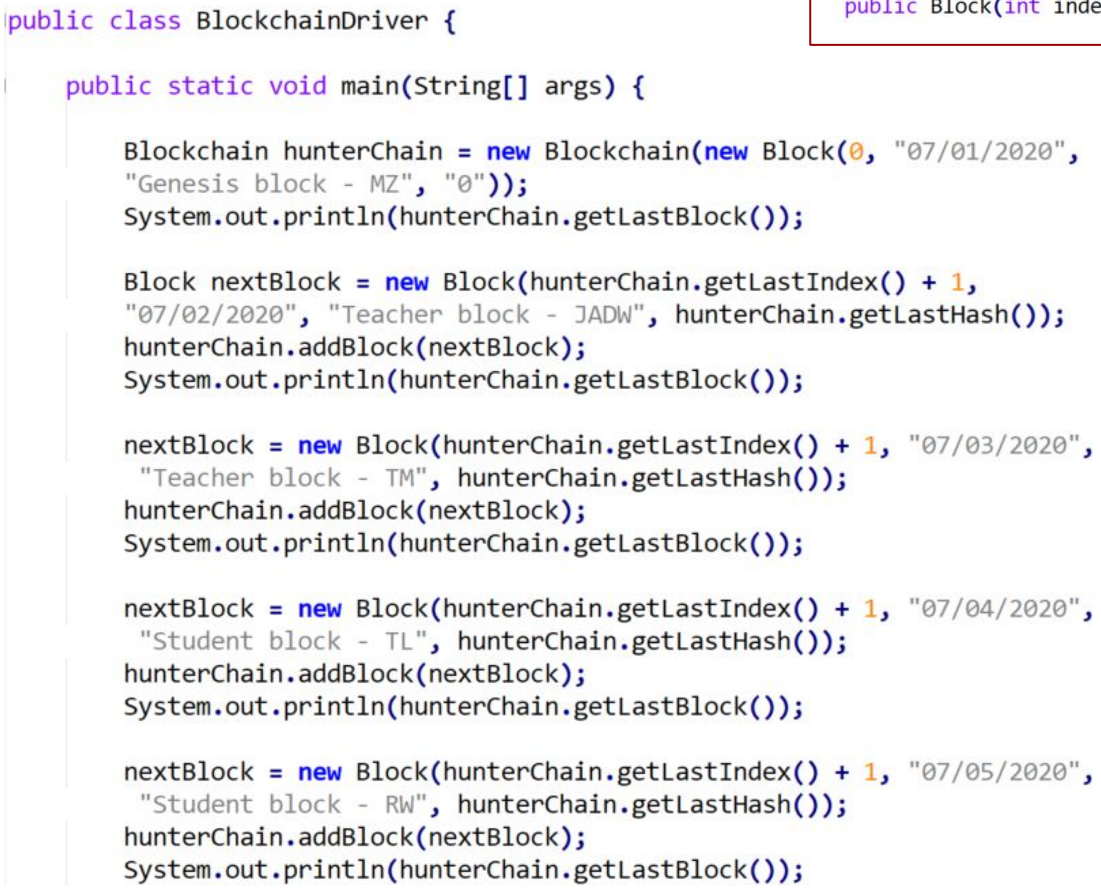
Modified from:
https://www.savjee.be/2017/07/Writing-tiny-blockchain-in-JavaScript/

Coding Example of B-l-o-c-k-c-h-a-i-n Implementation

File: BlockchainDriver.java

Reminder of Block constructor:

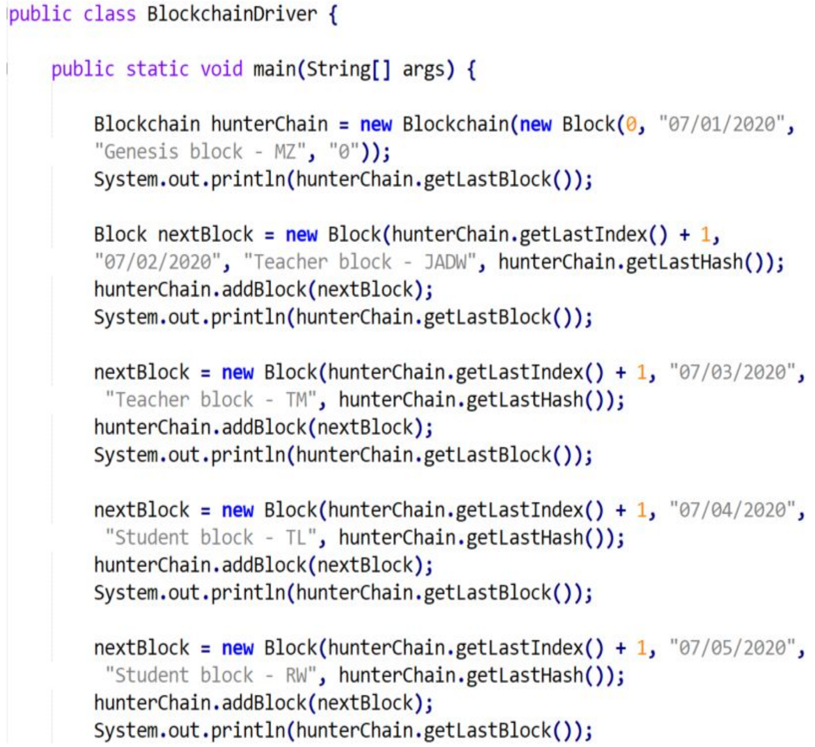
```
public Block(int index, String timestamp, String data, String previousHash)
```



```
public class BlockchainDriver {

    public static void main(String[] args) {

        Blockchain hunterChain = new Blockchain(new Block(0, "07/01/2020",
        "Genesis block - MZ", "0"));
        System.out.println(hunterChain.getLastBlock());

        Block nextBlock = new Block(hunterChain.getLastIndex() + 1,
        "07/02/2020", "Teacher block - JADW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/03/2020",
         "Teacher block - TM", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/04/2020",
         "Student block - TL", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/05/2020",
         "Student block - RW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());
```

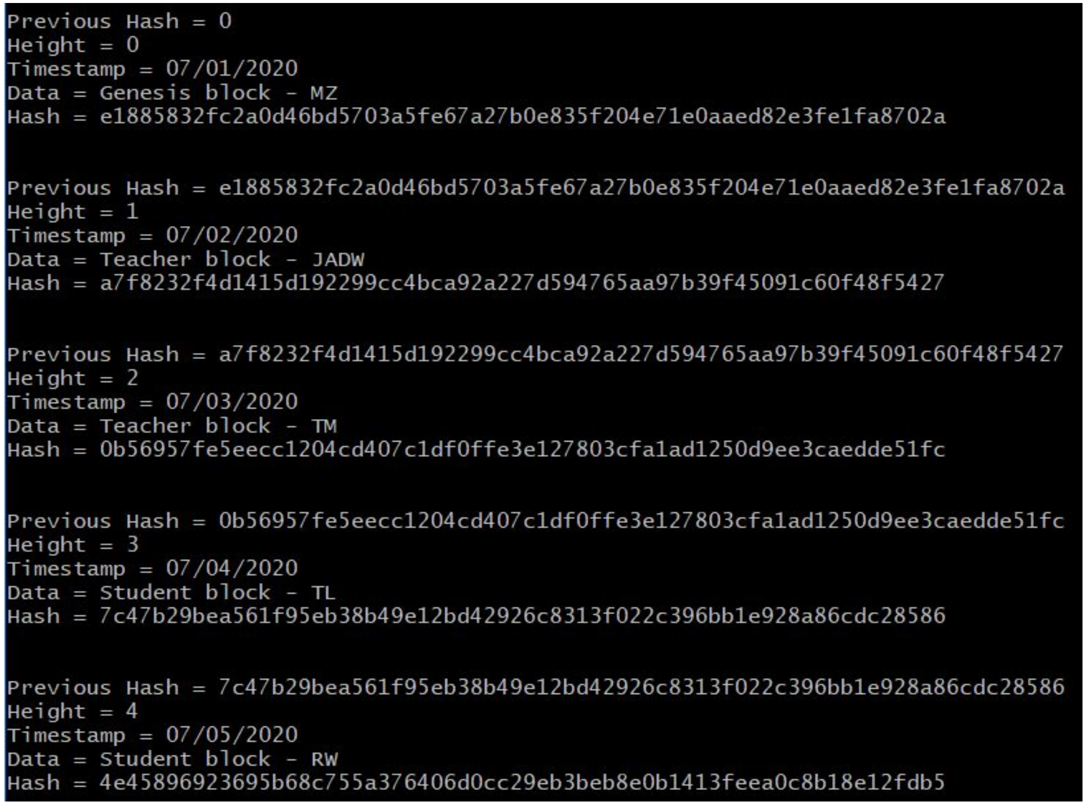
Coding Example of B-l-o-c-k-c-h-a-i-n Implementation

File: BlockchainDriver.java



```java
public class BlockchainDriver {

    public static void main(String[] args) {

        Blockchain hunterChain = new Blockchain(new Block(0, "07/01/2020",
        "Genesis block - MZ", "0"));
        System.out.println(hunterChain.getLastBlock());

        Block nextBlock = new Block(hunterChain.getLastIndex() + 1,
        "07/02/2020", "Teacher block - JADW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/03/2020",
         "Teacher block - TM", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/04/2020",
         "Student block - TL", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/05/2020",
         "Student block - RW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());
```

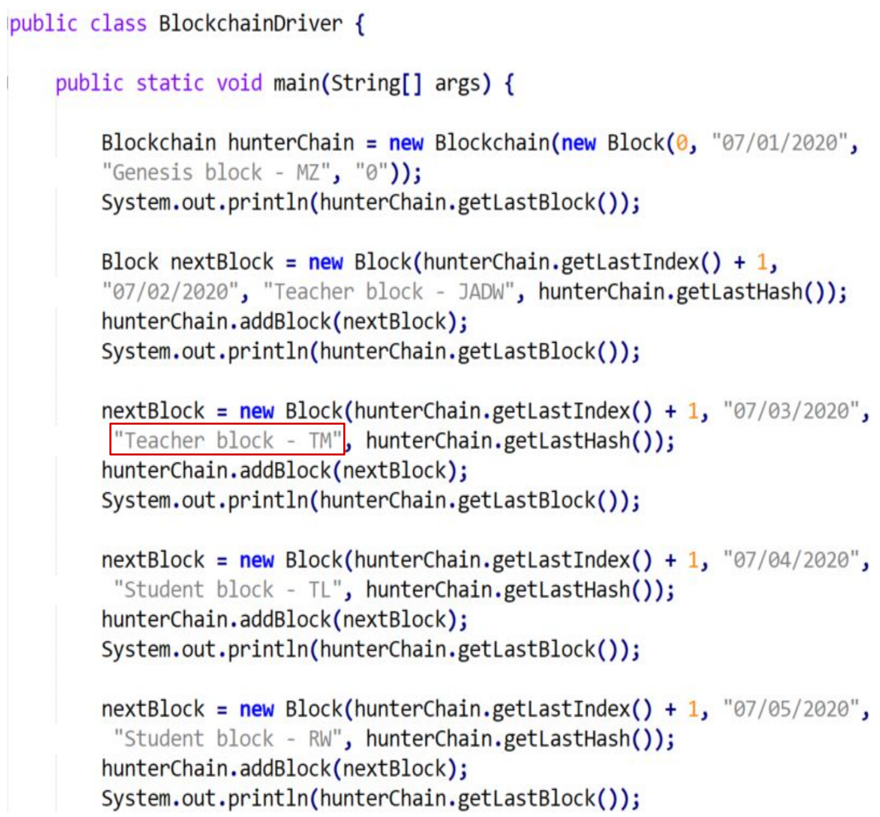
Output:



```
Previous Hash = 0
Height = 0
Timestamp = 07/01/2020
Data = Genesis block - MZ
Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a


Previous Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a
Height = 1
Timestamp = 07/02/2020
Data = Teacher block - JADW
Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427


Previous Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427
Height = 2
Timestamp = 07/03/2020
Data = Teacher block - TM
Hash = 0b56957fe5eecc1204cd407c1df0ffe3e127803cfa1ad1250d9ee3caedde51fc


Previous Hash = 0b56957fe5eecc1204cd407c1df0ffe3e127803cfa1ad1250d9ee3caedde51fc
Height = 3
Timestamp = 07/04/2020
Data = Student block - TL
Hash = 7c47b29bea561f95eb38b49e12bd42926c8313f022c396bb1e928a86cdc28586


Previous Hash = 7c47b29bea561f95eb38b49e12bd42926c8313f022c396bb1e928a86cdc28586
Height = 4
Timestamp = 07/05/2020
Data = Student block - RW
Hash = 4e45896923695b68c755a376406d0cc29eb3beb8e0b1413feea0c8b18e12fdb5
```

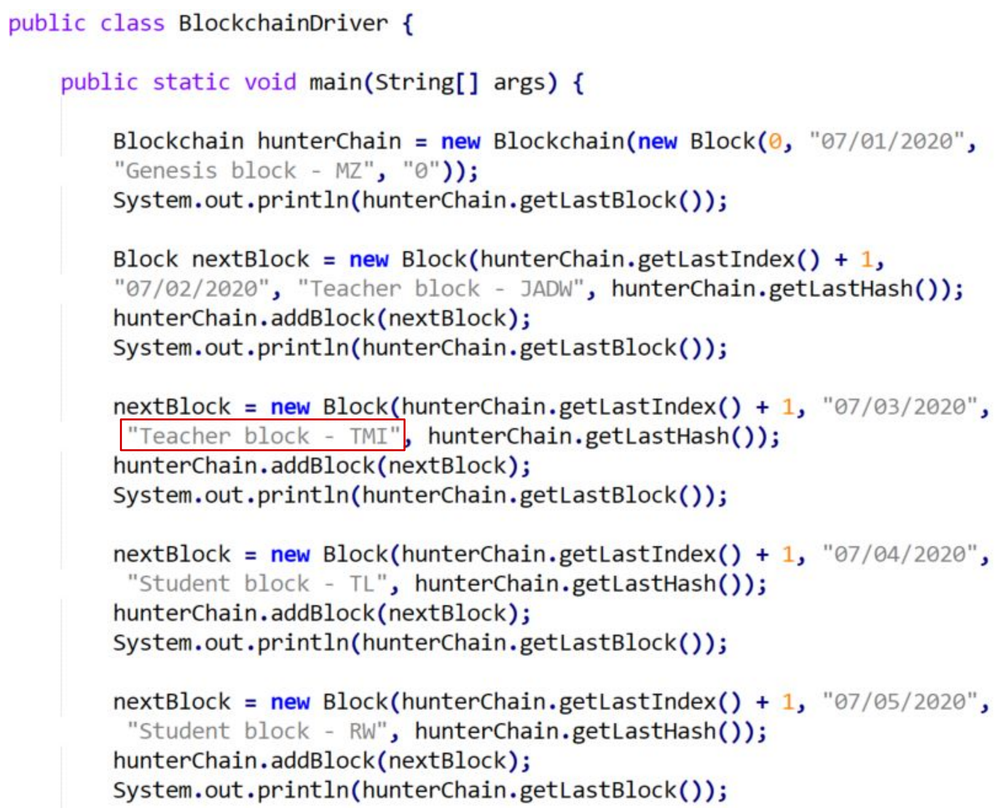
Coding Example of B-l-o-c-k-c-h-a-i-n Implementation

Original File: BlockchainDriver.java



```
public class BlockchainDriver {

    public static void main(String[] args) {

        Blockchain hunterChain = new Blockchain(new Block(0, "07/01/2020",
        "Genesis block - MZ", "0"));
        System.out.println(hunterChain.getLastBlock());

        Block nextBlock = new Block(hunterChain.getLastIndex() + 1,
        "07/02/2020", "Teacher block - JADW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/03/2020",
         "Teacher block - TM", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/04/2020",
         "Student block - TL", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/05/2020",
         "Student block - RW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());
```

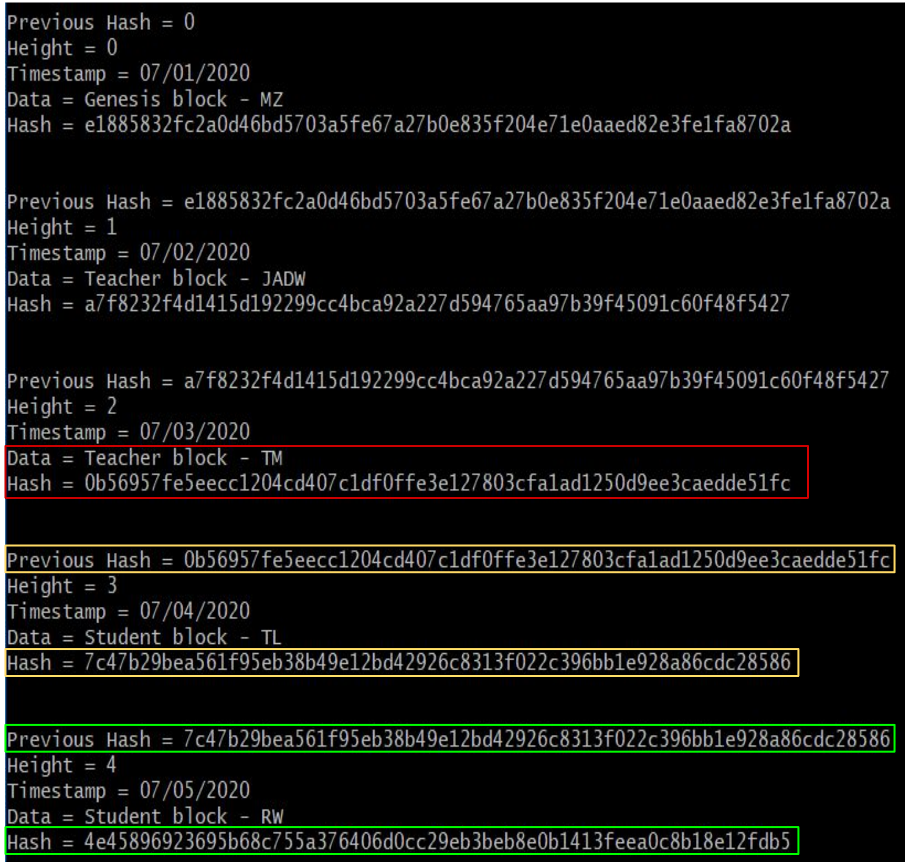
Modified File: BlockchainDriver.java



```
public class BlockchainDriver {

    public static void main(String[] args) {

        Blockchain hunterChain = new Blockchain(new Block(0, "07/01/2020",
        "Genesis block - MZ", "0"));
        System.out.println(hunterChain.getLastBlock());

        Block nextBlock = new Block(hunterChain.getLastIndex() + 1,
        "07/02/2020", "Teacher block - JADW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/03/2020",
         "Teacher block - TMI", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/04/2020",
         "Student block - TL", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());

        nextBlock = new Block(hunterChain.getLastIndex() + 1, "07/05/2020",
         "Student block - RW", hunterChain.getLastHash());
        hunterChain.addBlock(nextBlock);
        System.out.println(hunterChain.getLastBlock());
```

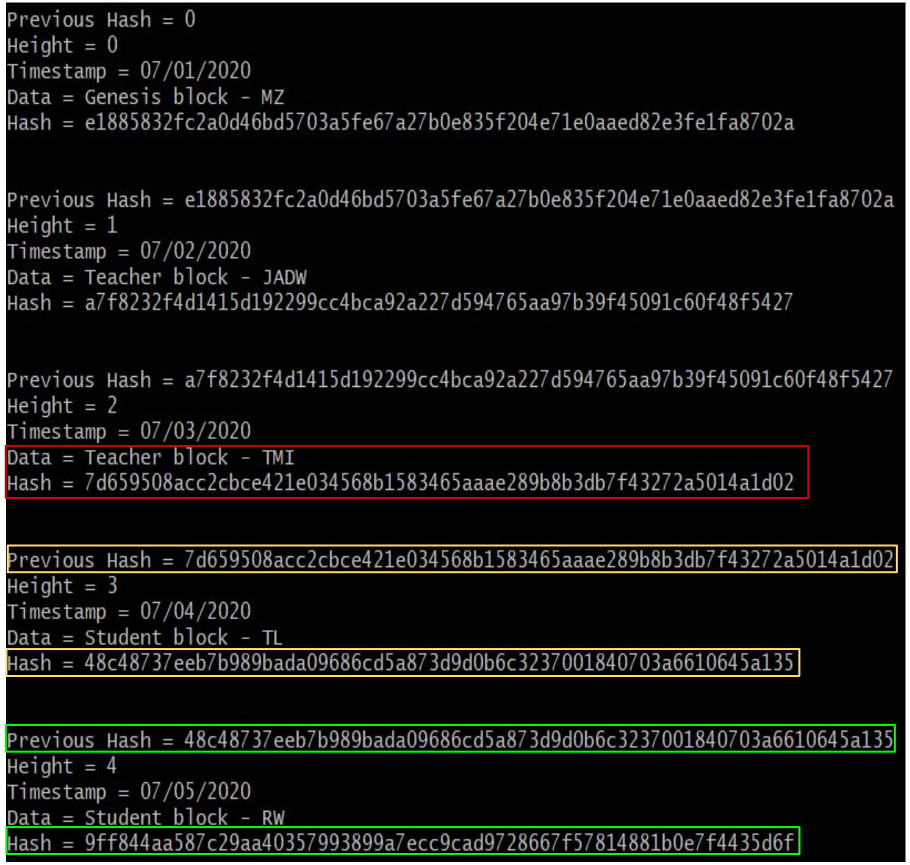
Coding Example of B-l-o-c-k-c-h-a-i-n Implementation

Original Output:



```
Previous Hash = 0
Height = 0
Timestamp = 07/01/2020
Data = Genesis block - MZ
Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a


Previous Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a
Height = 1
Timestamp = 07/02/2020
Data = Teacher block - JADW
Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427


Previous Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427
Height = 2
Timestamp = 07/03/2020
Data = Teacher block - TM
Hash = 0b56957fe5eecc1204cd407c1df0ffe3e127803cfa1ad1250d9ee3caedde51fc


Previous Hash = 0b56957fe5eecc1204cd407c1df0ffe3e127803cfa1ad1250d9ee3caedde51fc
Height = 3
Timestamp = 07/04/2020
Data = Student block - TL
Hash = 7c47b29bea561f95eb38b49e12bd42926c8313f022c396bb1e928a86cdc28586


Previous Hash = 7c47b29bea561f95eb38b49e12bd42926c8313f022c396bb1e928a86cdc28586
Height = 4
Timestamp = 07/05/2020
Data = Student block - RW
Hash = 4e45896923695b68c755a376406d0cc29eb3beb8e0b1413feea0c8b18e12fdb5
```

Modified Output:

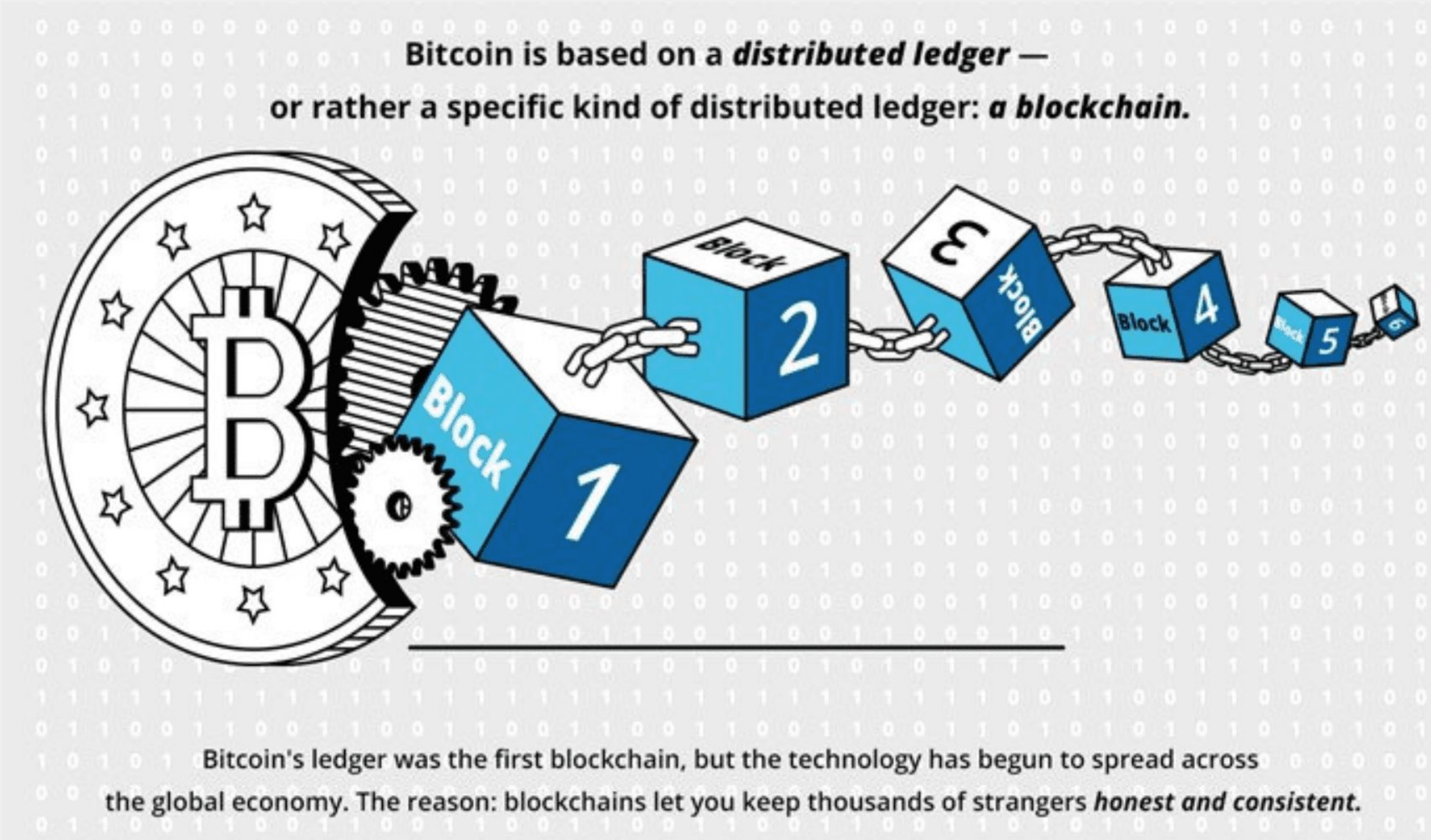


```
Previous Hash = 0
Height = 0
Timestamp = 07/01/2020
Data = Genesis block - MZ
Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a


Previous Hash = e1885832fc2a0d46bd5703a5fe67a27b0e835f204e71e0aaed82e3fe1fa8702a
Height = 1
Timestamp = 07/02/2020
Data = Teacher block - JADW
Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427


Previous Hash = a7f8232f4d1415d192299cc4bca92a227d594765aa97b39f45091c60f48f5427
Height = 2
Timestamp = 07/03/2020
Data = Teacher block - TMI
Hash = 7d659508acc2cbce421e034568b1583465aaae289b8b3db7f43272a5014a1d02


Previous Hash = 7d659508acc2cbce421e034568b1583465aaae289b8b3db7f43272a5014a1d02
Height = 3
Timestamp = 07/04/2020
Data = Student block - TL
Hash = 48c48737eeb7b989bada09686cd5a873d9d0b6c3237001840703a6610645a135


Previous Hash = 48c48737eeb7b989bada09686cd5a873d9d0b6c3237001840703a6610645a135
Height = 4
Timestamp = 07/05/2020
Data = Student block - RW
Hash = 9ff844aa587c29aa40357993899a7ecc9cad9728667f57814881b0e7f4435d6f
```

Part 3 - So I’ve heard of cryptocurrencies. How does the Bitcoin blockchain work?

How does Bitcoin use blockchain?

Source (and recommended reading!): https://www.investopedia.com/terms/b/blockchain.asp

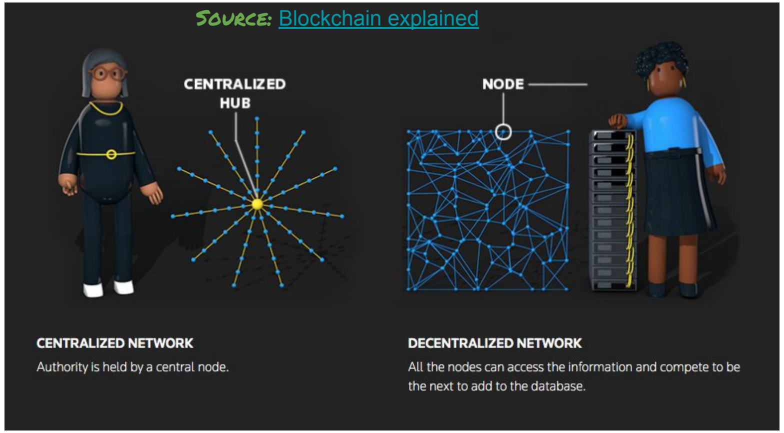


Bitcoin is based on a ***distributed ledger*** —
or rather a specific kind of distributed ledger: ***a blockchain.***

Bitcoin's ledger was the first blockchain, but the technology has begun to spread across the global economy. The reason: blockchains let you keep thousands of strangers ***honest and consistent.***

What modifications to a basic blockchain would be needed to implement a cryptocurrency like bitcoin?

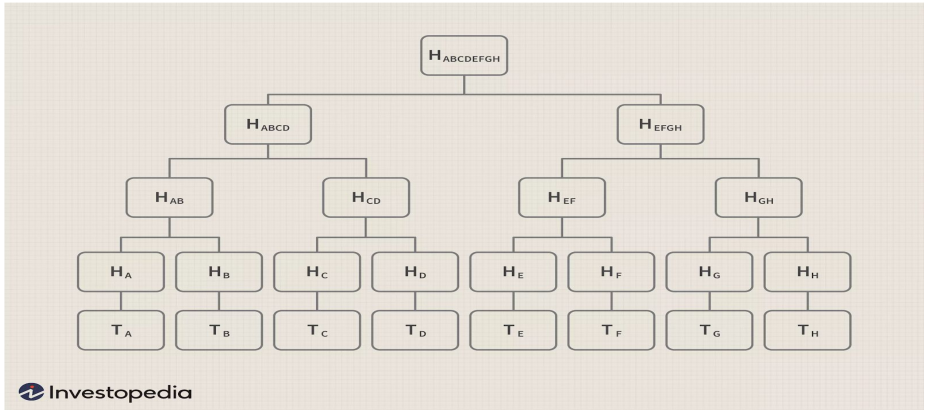
Decentralized - Distributed Ledger



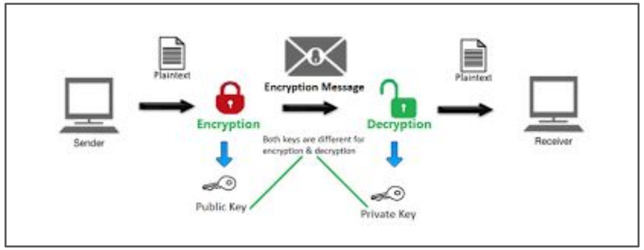
CENTRALIZED NETWORK
Authority is held by a central node.

DECENTRALIZED NETWORK
All the nodes can access the information and compete to be the next to add to the database.

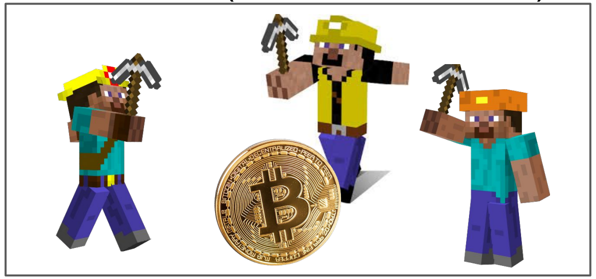
Merkle Tree

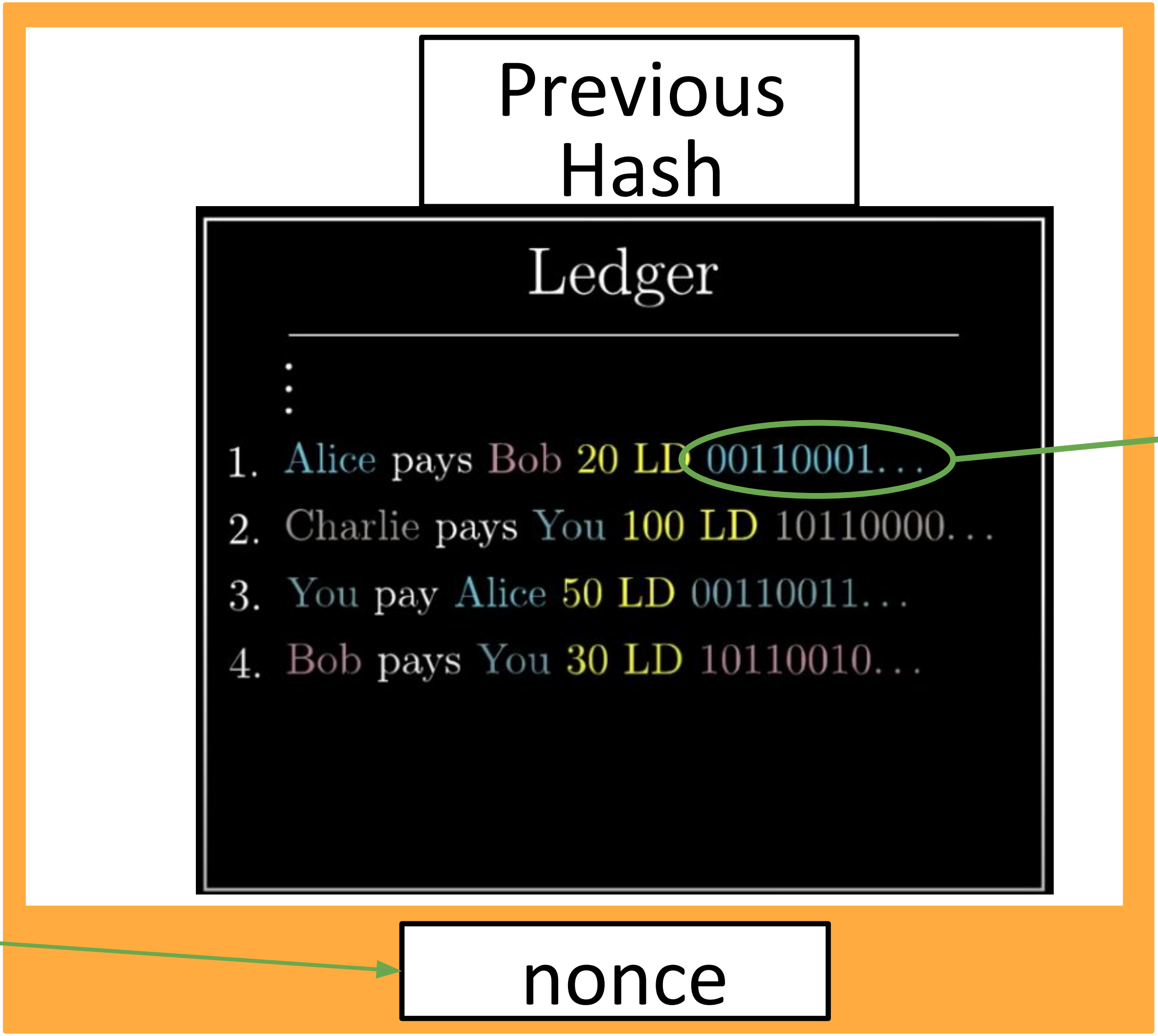


Public - Private Key Encryption



Miners (and Proof of Work)





Ledger

⋮

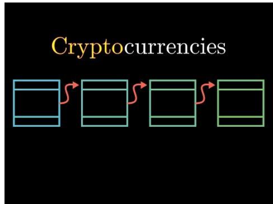
1. Alice pays Bob 20 LD 00110001. . .
2. Charlie pays You 100 LD 10110000. . .
3. You pay Alice 50 LD 00110011. . .
4. Bob pays You 30 LD 10110010. . .

Public - Private Key Encryption verifies each transaction with a signature.

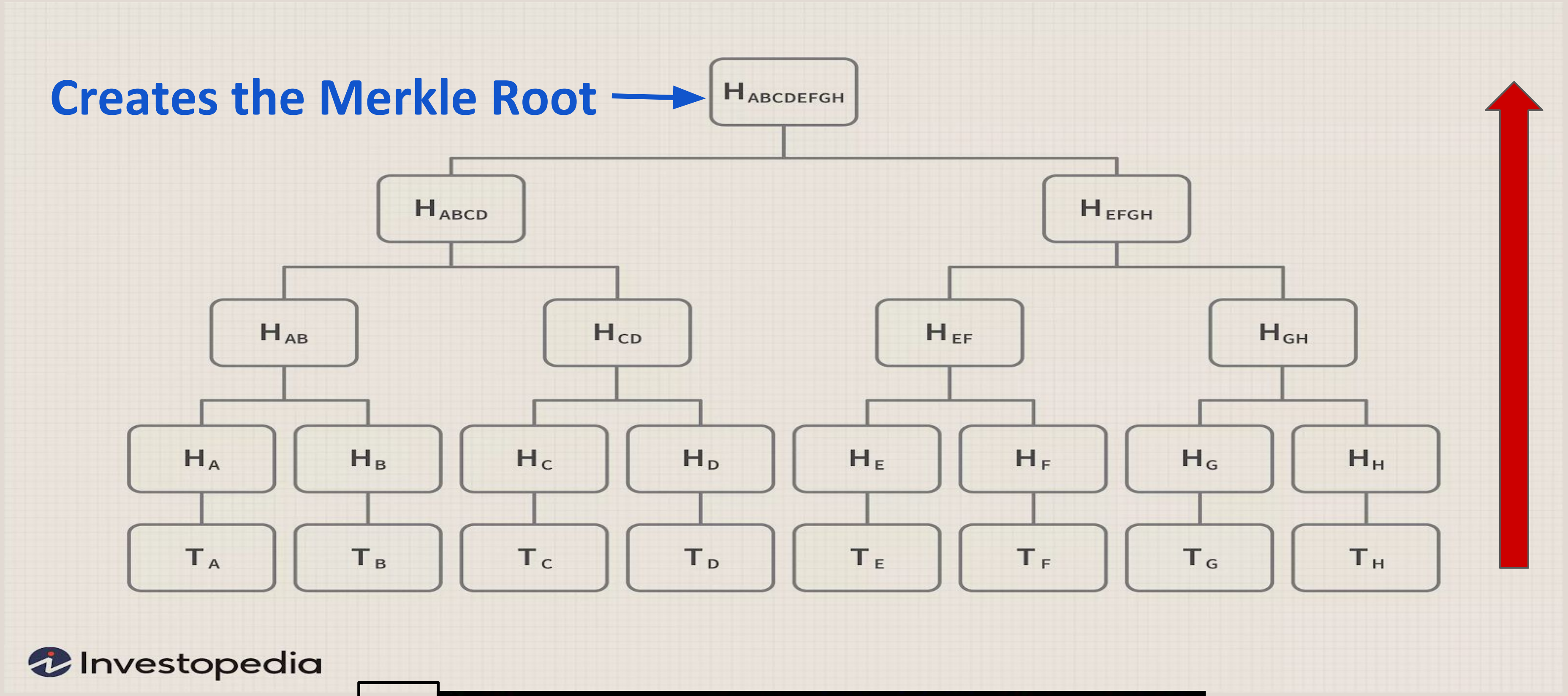
__**Definition:**__

nonce -
a “number used once”

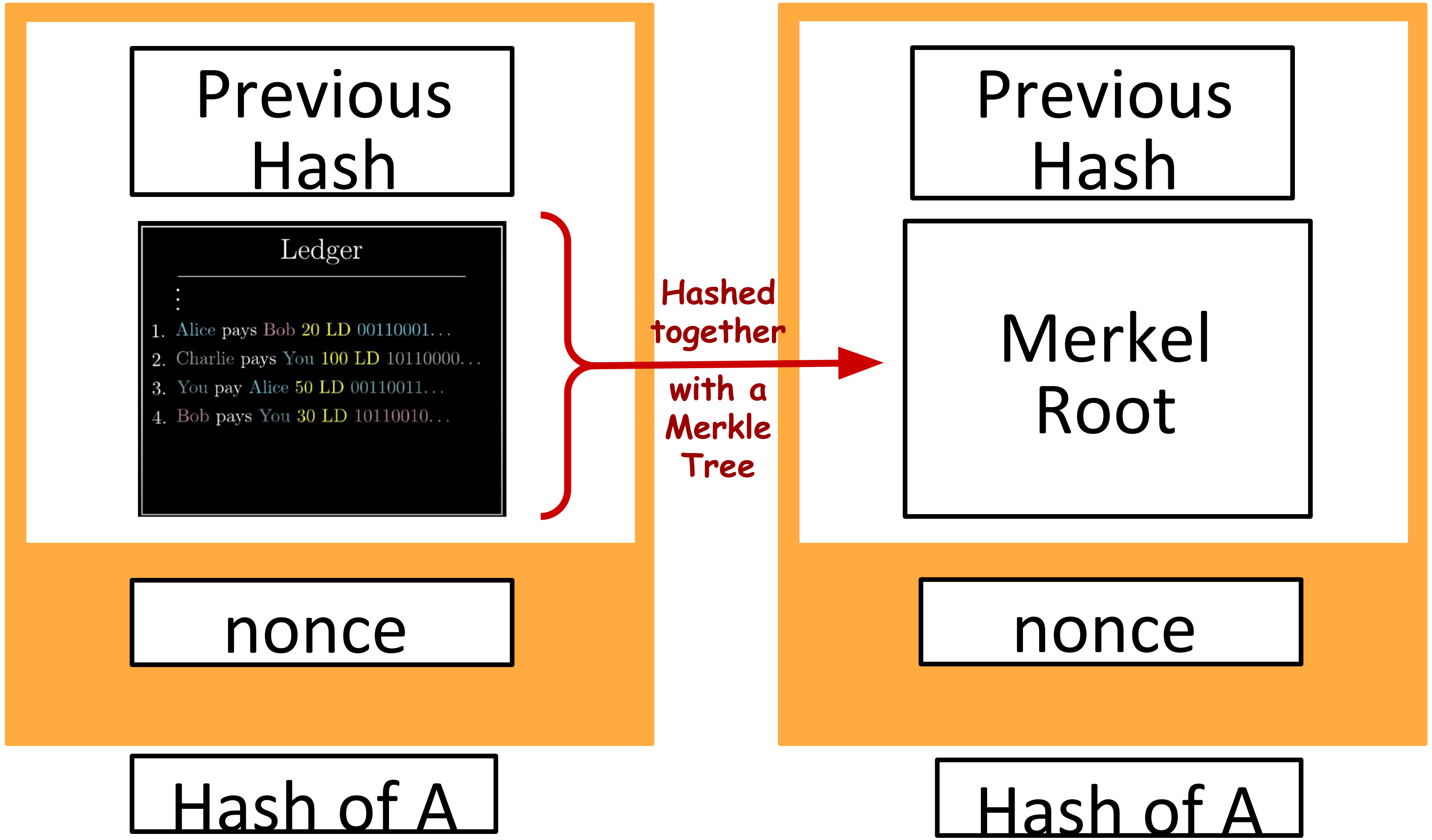
SOURCE:



Merkle Tree or Binomial Hash Tree



T_A Alice pays Bob 20 LD 00110001

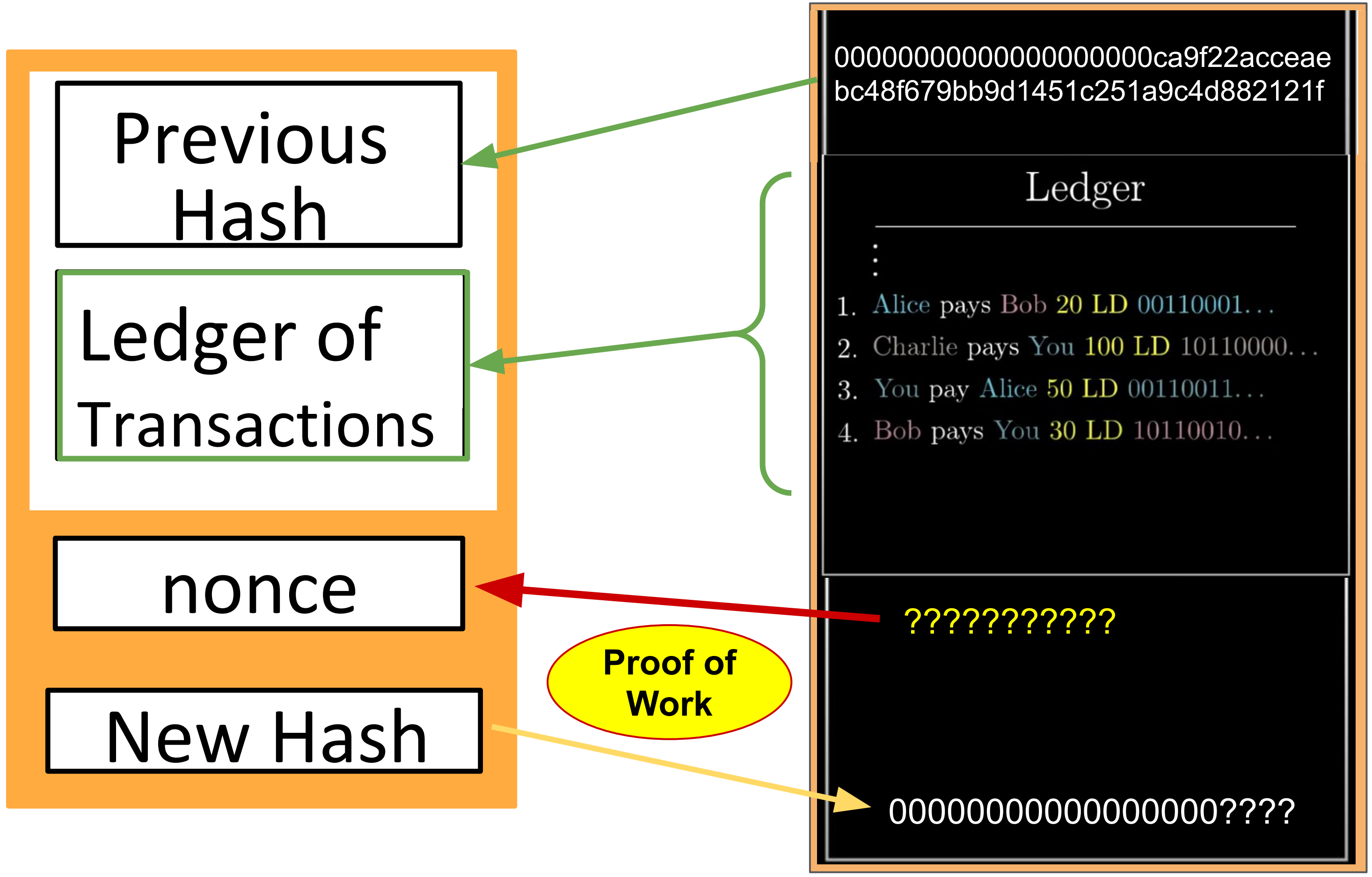


Ledger

⋮

1. Alice pays Bob 20 LD 00110001...
2. Charlie pays You 100 LD 10110000...
3. You pay Alice 50 LD 00110011...
4. Bob pays You 30 LD 10110010...

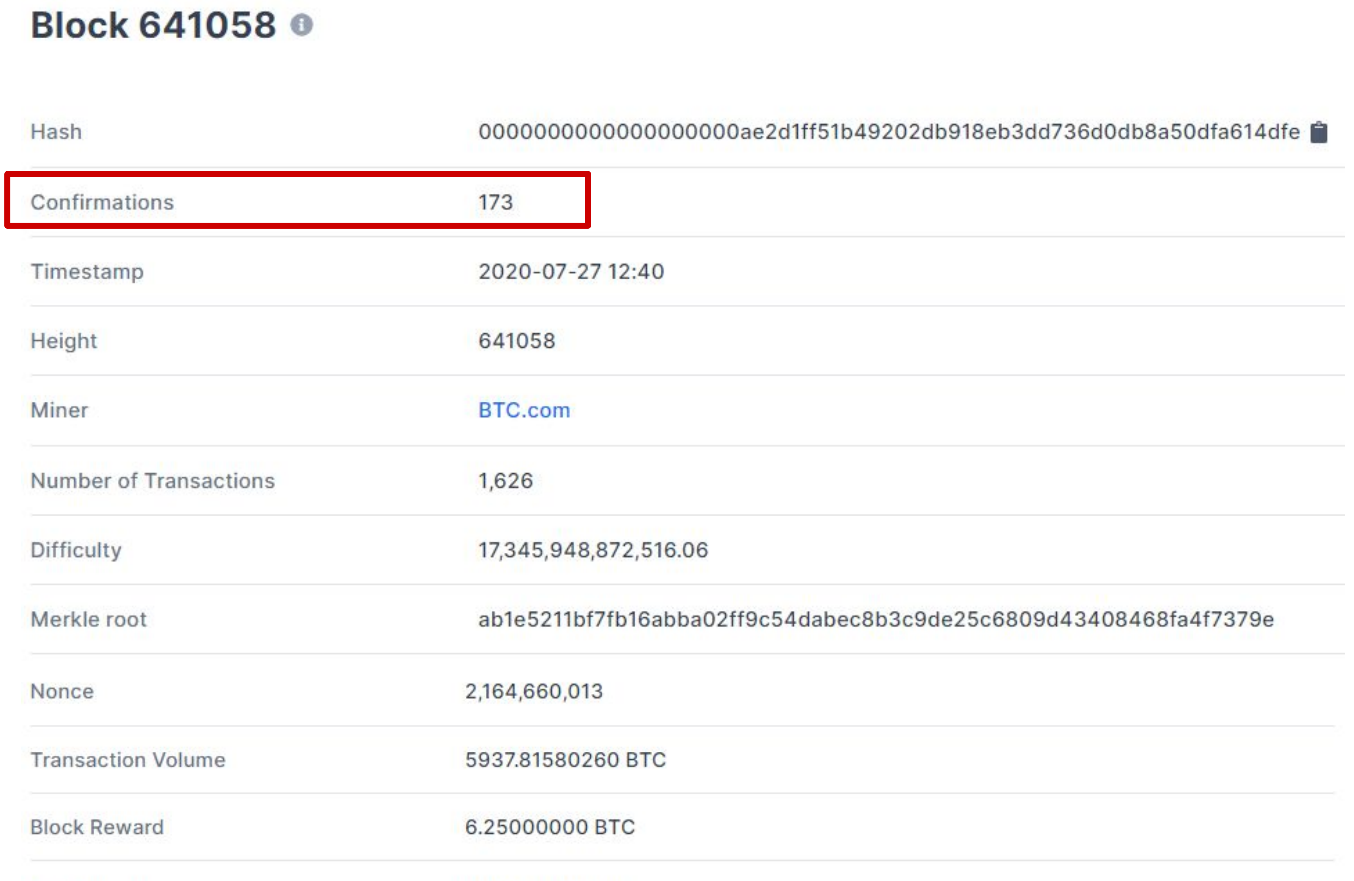
Creating a new Bitcoin block:


Previous Hash
Ledger of Transactions
nonce
New Hash
Proof of Work
000000000000000000ca9f22acceae bc48f679bb9d1451c251a9c4d882121f
Ledger
⋮
1. Alice pays Bob 20 LD 00110001...
2. Charlie pays You 100 LD 10110000...
3. You pay Alice 50 LD 00110011...
4. Bob pays You 30 LD 10110010...
???????????
000000000000000000????

Bitcoin Miners

Look at actual Bitcoin blocks!

on https://www.blockchain.com/explorer



Block 641058

Hash	0000000000000000000ae2d1ff51b49202db918eb3dd736d0db8a50dfa614dfe
Confirmations	173
Timestamp	2020-07-27 12:40
Height	641058
Miner	BTC.com
Number of Transactions	1,626
Difficulty	17,345,948,872,516.06
Merkle root	ab1e5211bf7fb16abba02ff9c54dabec8b3c9de25c6809d43408468fa4f7379e
Nonce	2,164,660,013
Transaction Volume	5937.81580260 BTC
Block Reward	6.25000000 BTC

Proof of Dumplings

WHEN & WHERE: @3PM a file will be placed in our project repository called "ProofOfDumplings.java" (The link will also be posted on Slack.)

WHAT TO DO: The file will contain an input phrase (the data). You must find a nonce value (an int or long) which, when appended to the end of the input phrase, generates a hash value with 7 leading zeros. The hash must be generated using the SHA-256 function.

HOW TO WIN: The first person to report their **nonce** and **resulting hash value** (digest) on Slack will receive ~~a $20 gift card for dumplings~~ "all the points."

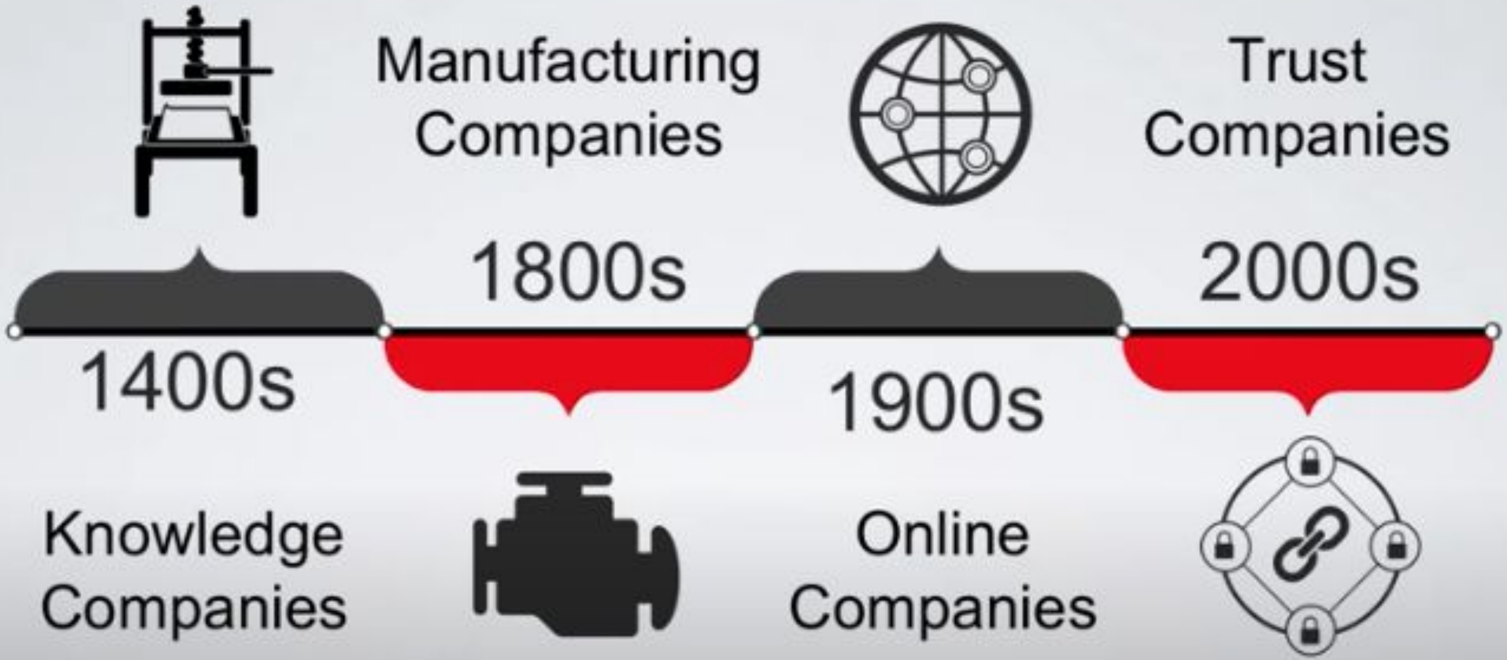
Part 4 - Blockchain… what is it good for?

(Absolutely… everything?)

Conclusion: Why is blockchain technology so popular right now?

- **Open ledgers** - data transparency (with privacy)
- **Security** - the data is really hard to tamper
  - Hashing makes altering data computationally expensive
  - Distributed network
    - Everyone (each node) has a copy of the data
    - Data is continuously verified as new blocks are added
- **Peer-2-Peer** - no need for middle man (Trusted Third Party)
  - Transactions are verified and secured with public and private encryption keys
  - ***_Smart Contracts_** - parties using a blockchain can set rules in the code that automatically respond to recorded data (i.e. - money payments)

Blockchain: A Potential For Change



(Printing Press) spread knowledge

(Engine) filled the power gap

(Internet) filled the distance gap - reduced brick and mortar and increased online access to services and products

(Blockchain) changes the way we trust

SOURCE:

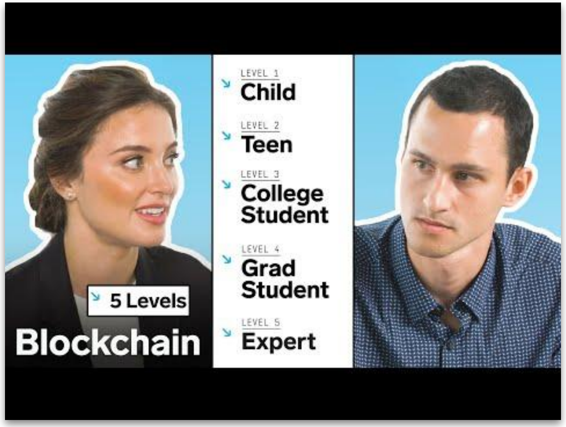
"Blockchain will be part of the everyday infrastructure of our world" - 2nd Video

Already in place:

- Walmart - Supply Chain
- Bitcoin
- Other blockchain projects by Hyperledger

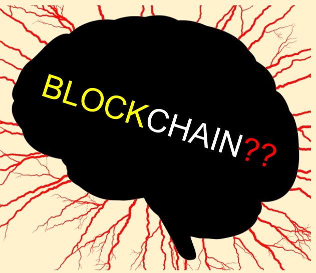
In the future:

- Healthcare - Medical records
- Elections - secure voting
- Property Exchange
- Public records
- Education
- Energy / Climate
- . . .



ADDITIONAL ACTIVITIES & RESOURCES

1. Brainstorm Blockchain Applications & Write a Proposal
2. The Blockchain Game - an unplugged activity
3. Code a Blockchain
4. Create a Merkle Tree to Find a Merkle Root
5. Article - What happens if you lose your Bitcoin private encryption key?



Research - Ideate - Design

Some Resources:
Video -
https://www.youtube.com/watch?v=r43LhSUUGTQ
Articles -
https://blockchainfutureslab.wordpress.com/

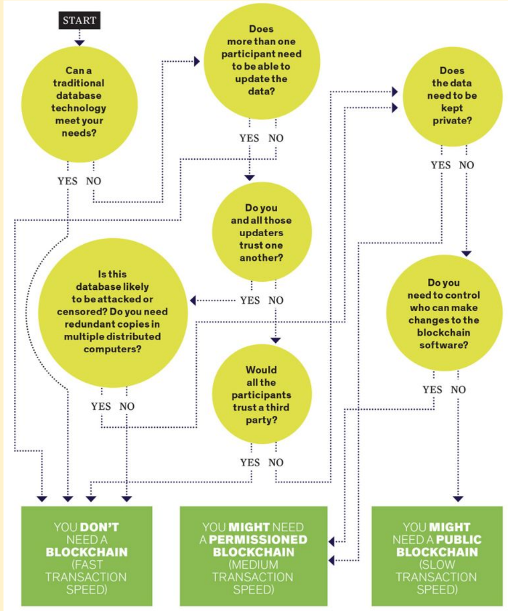
Student Task

Come up with ideas for where blockchains can be used in your everyday life. Then write a proposal to explaining how blockchain technology will revolutionize the industry you choose.

Guiding Questions

- ❏ When is a traditional database not good enough?
- ❏ Do multiple parties need to access & update the data?
- ❏ Is the database likely to be attacked or censored?
- ❏ Is there a need for privacy?
- ❏ Do you want to cut out the middleman in transactions?

SOURCE: Do You Need a Blockchain?


START
Can a traditional database technology meet your needs?
YES NO
Does more than one participant need to be able to update the data?
YES NO
Do you and all those updaters trust one another?
YES NO
Would all the participants trust a third party?
YES NO
Is this database likely to be attacked or censored? Do you need redundant copies in multiple distributed computers?
YES NO
Does the data need to be kept private?
YES NO
Do you need to control who can make changes to the blockchain software?
YES NO
YOU DON'T NEED A BLOCKCHAIN (FAST TRANSACTION SPEED)
YOU MIGHT NEED A PERMISSIONED BLOCKCHAIN (MEDIUM TRANSACTION SPEED)
YOU MIGHT NEED A PUBLIC BLOCKCHAIN (SLOW TRANSACTION SPEED)

The Blockchain Game

Source: https://www.instructables.com/id/The-Blockchain-Game/

Brief Description:
Students become the miners to determine hash/nonce values which need to be verified by 51% of the class.

- No coding.
- There are handouts provided.
- Students may need calculators but no computers.
- Teacher uses spreadsheet to demonstrate/store the blockchain.

Code a B-l-o-c-k-c-h-a-i-n

Part 1 - Ideate:

Determine an application for blockchain - what kind of data would we want to make permanent and tamper-proof?

Part 2 - Code:

RECOMMENDED RESOURCE: https://www.savjee.be/2017/07/Writing-tiny-blockchain-in-JavaScript/

Create a Block class to store whatever data is necessary for your application. (Remember, you can add blocks, but you can't alter their information. They should be immutable! Make whatever getter methods you need, but do not make any setters.)

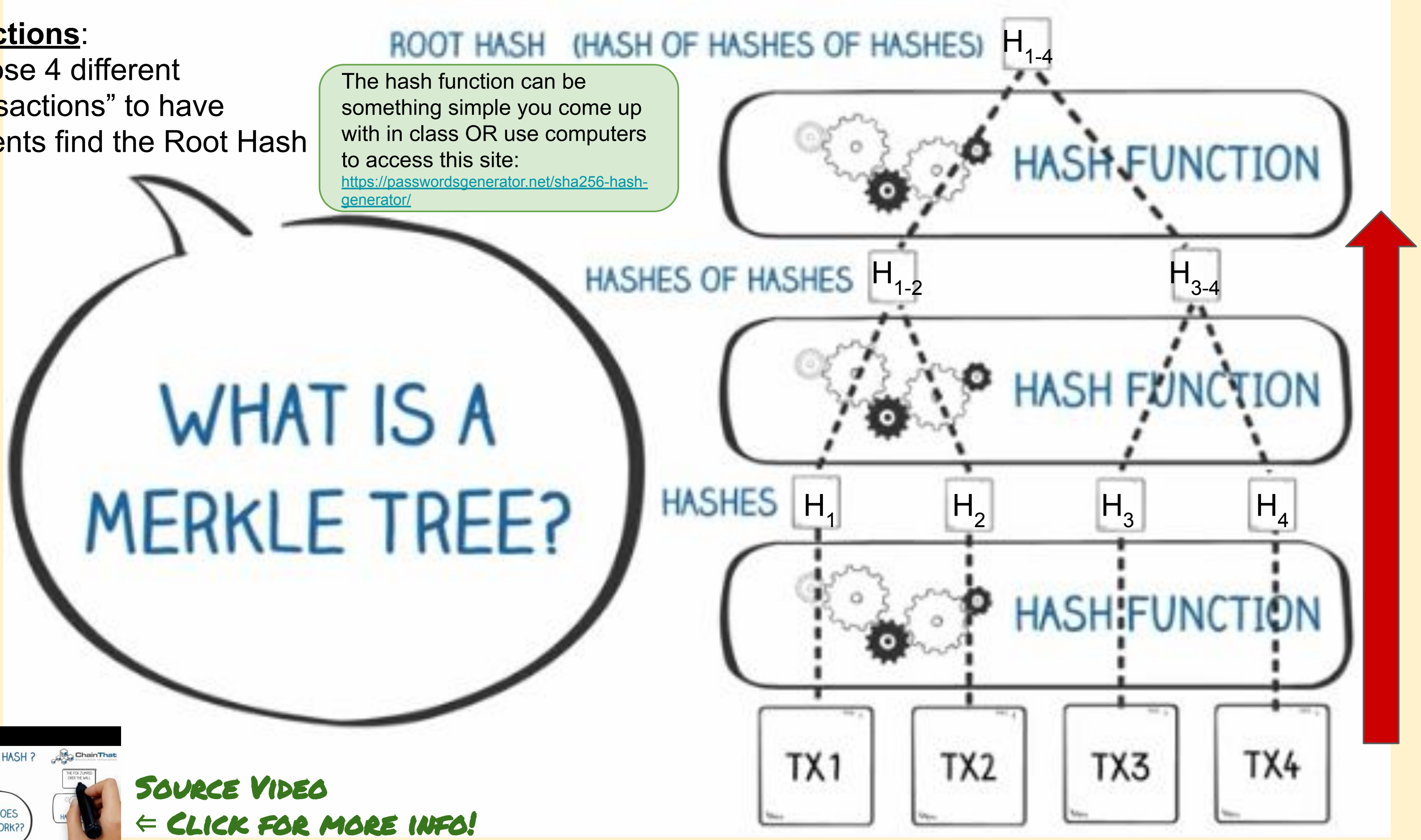
Implement a Blockchain class using your choice of list and demonstrate how the hash from each block affects the hash of the next block, therefore making each block dependent on another and our "chain" secure.

Extension:

Make a transaction class and store multiple transactions in a block with a Merkle tree/root.

Directions:
Choose 4 different "transactions" to have students find the Root Hash

The hash function can be something simple you come up with in class OR use computers to access this site: https://passwordsgenerator.net/sha256-hash-generator/



Source Video
⇐ Click for more info!

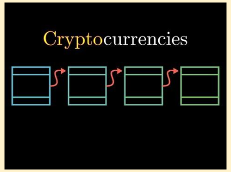
Need more cryptography?

Public - Private Key Encryption

- What happens if you lose your private key? [*Read the linked article*]

  If a user misplaces their private key, they will lose access to their bitcoin wallet, as was the case with this man who made national headlines in December of 2017.

- Go into more depth on how transactions are handled/verified for Bitcoin.



  - Watch this youtube video from 3Blue1Brown to start the conversation with students about how to create a Transaction class that verifies public and private keys: https://www.youtube.com/watch?v=bBC-nXj3Ng4
  - Use this website to see code implementation in Javascript for a Transaction class with public-private keys: https://www.savjee.be/2018/10/Signing-transactions-blockchain-javascript/