블록체인 (Blockchain)

02. Theory

소프트웨어 꼰대 강의

노기섭 교수

(kafa46@cju.ac.kr)

Contents

- Blockchain History
- How to develop blockchain system
- Blockchain Overview
- Hash Function

Early Contributors in Blockchain Technology

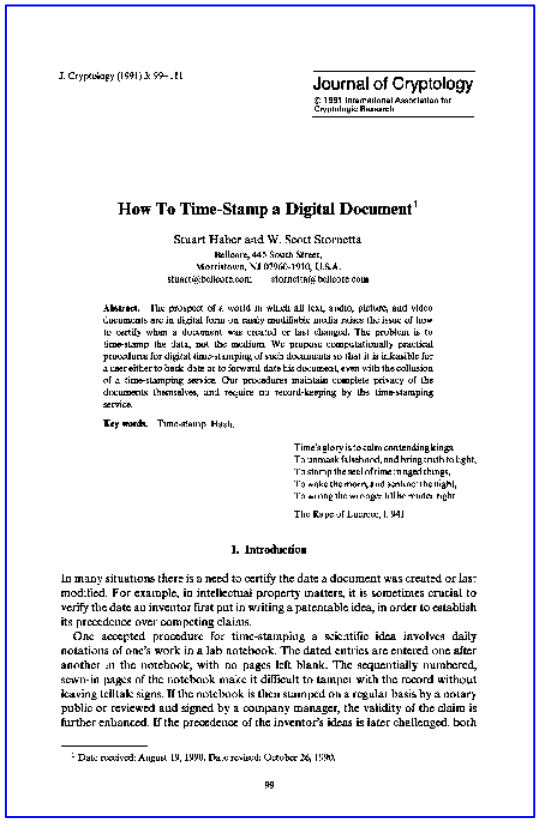
Stuart Haber

https://en.wikipedia.org/wiki/Stuart_Haber

W. Scott Stornetta

https://en.wikipedia.org/wiki/W._Scott_Stornetta

https://link.springer.com/article/10.1007/BF00196791



J. Cryptology (1991) 3: 99–111

Journal of Cryptology

© 1991 International Association for Cryptologic Research

How To Time-Stamp a Digital Document[1]

Stuart Haber and W. Scott Stornetta

Bellcore, 445 South Street,
Morristown, NJ 07960-1910, U.S.A.
stuart@bellcore.com stornetta@bellcore.com

Abstract. The prospect of a world in which all text, audio, picture, and video documents are in digital form on easily modifiable media raises the issue of how to certify when a document was created or last changed. The problem is to time-stamp the data, not the medium. We propose computationally practical procedures for digital time-stamping of such documents so that it is infeasible for a user either to back-date or to forward-date his document, even with the collusion of a time-stamping service. Our procedures maintain complete privacy of the documents themselves, and require no record-keeping by the time-stamping service.

Key words. Time-stamp, Hash.

Time's glory is to calm contending kings
To unmask falsehood, and bring truth to light,
To stamp the seal of time in aged things,
To wake the morn, and sentinel the night,
To wrong the wronger till he render right

The Rape of Lucrece, l. 941

1. Introduction

In many situations there is a need to certify the date a document was created or last modified. For example, in intellectual property matters, it is sometimes crucial to verify the date an inventor first put in writing a patentable idea, in order to establish its precedence over competing claims.

One accepted procedure for time-stamping a scientific idea involves daily notations of one's work in a lab notebook. The dated entries are entered one after another in the notebook, with no pages left blank. The sequentially numbered, sewn-in pages of the notebook make it difficult to tamper with the record without leaving telltale signs. If the notebook is then stamped on a regular basis by a notary public or reviewed and signed by a company manager, the validity of the claim is further enhanced. If the precedence of the inventor's ideas is later challenged, both

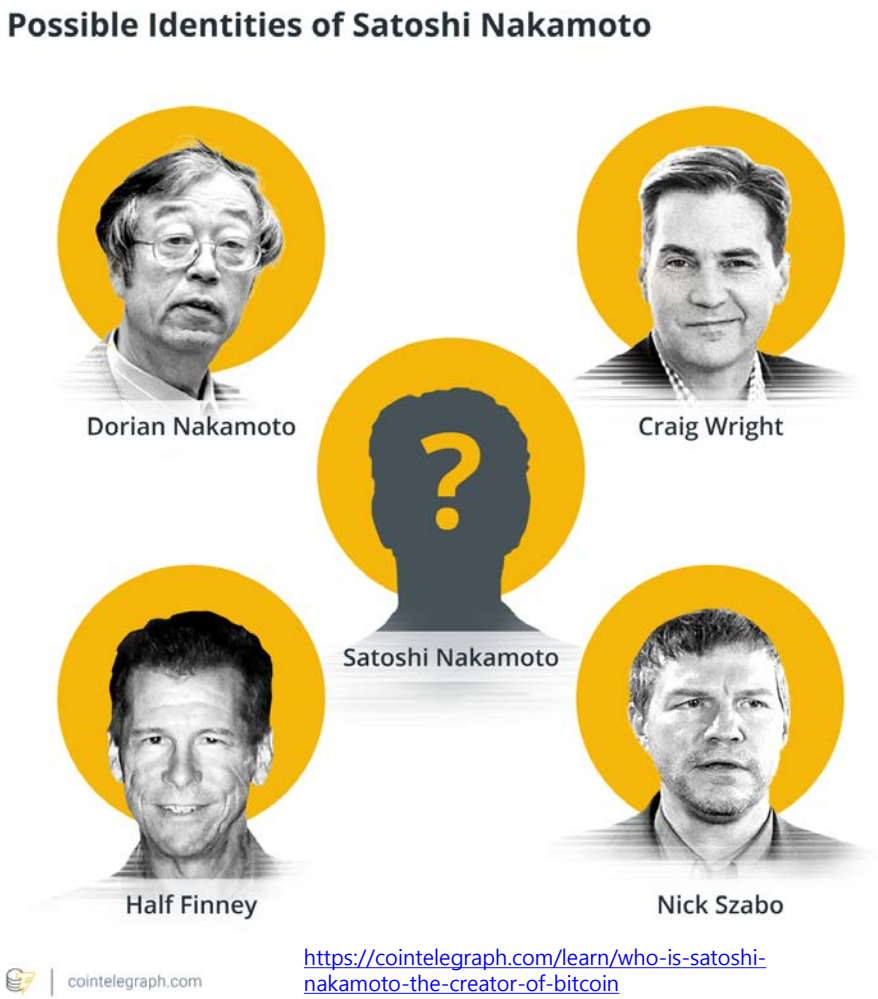
[1] Date received: August 19, 1990. Date revised: October 26, 1990.

99

Stuart Haber is an American cryptographer and computer scientist, known for his contributions in cryptography and privacy-preserving technologies and widely recognized as the co-inventor of the blockchain.

His 1991 paper "How to Time-Stamp a Digital Document",[1] co-authored with **W. Scott Stornetta**, won the 1992 Discover Award for Computer Software and is considered to be one of the most important papers in the development of cryptocurrencies.

Who is the inventor of Bitcoin?



https://cointelegraph.com/learn/who-is-satoshi-nakamoto-the-creator-of-bitcoin

'사토시 나카모토' 라는 사람?
정확히 누구인지 알 수 없음.

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

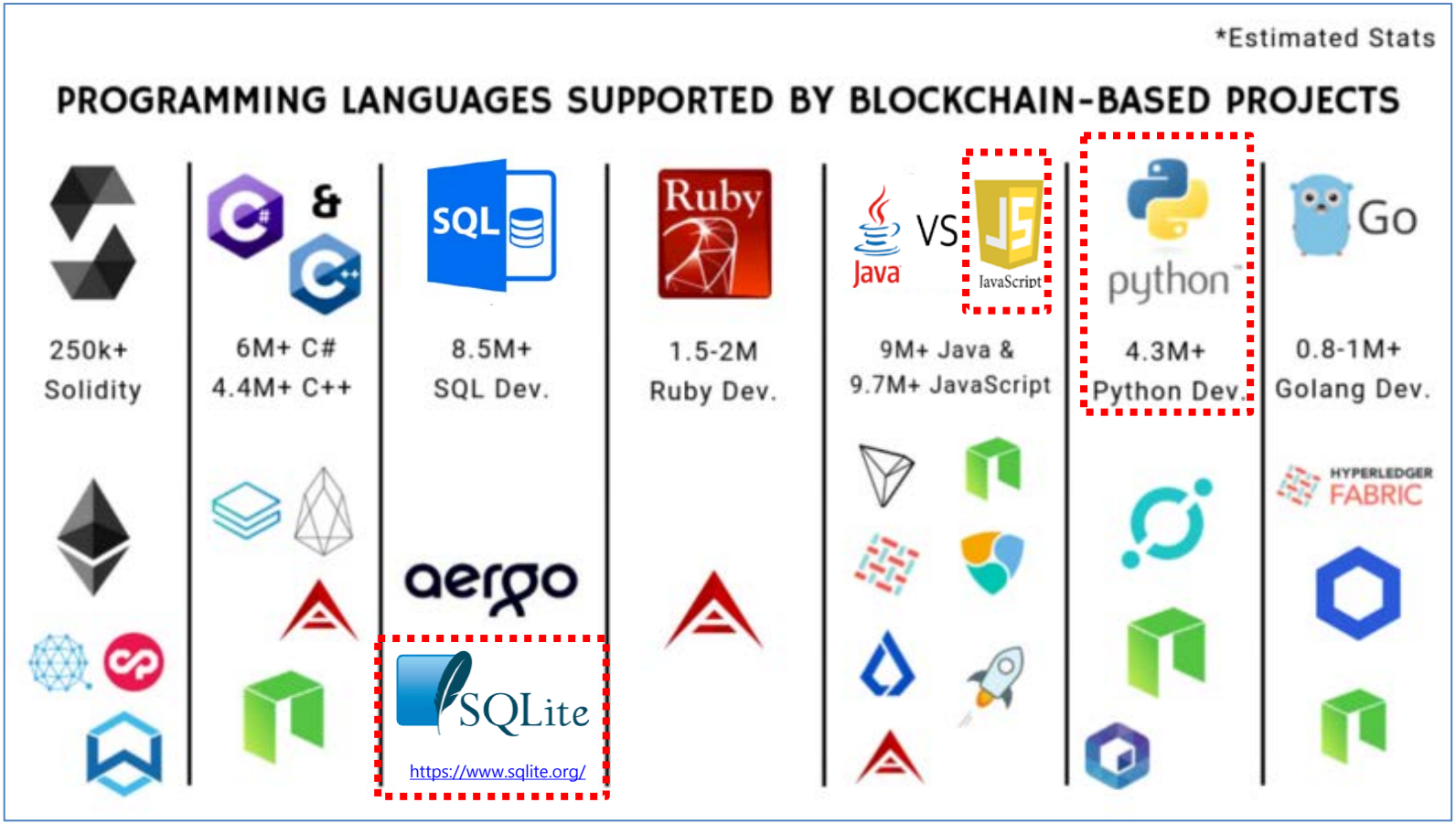
1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

논문 다운로드: https://bitcoin.org/bitcoin.pdf

Popular programming languages used in blockchain

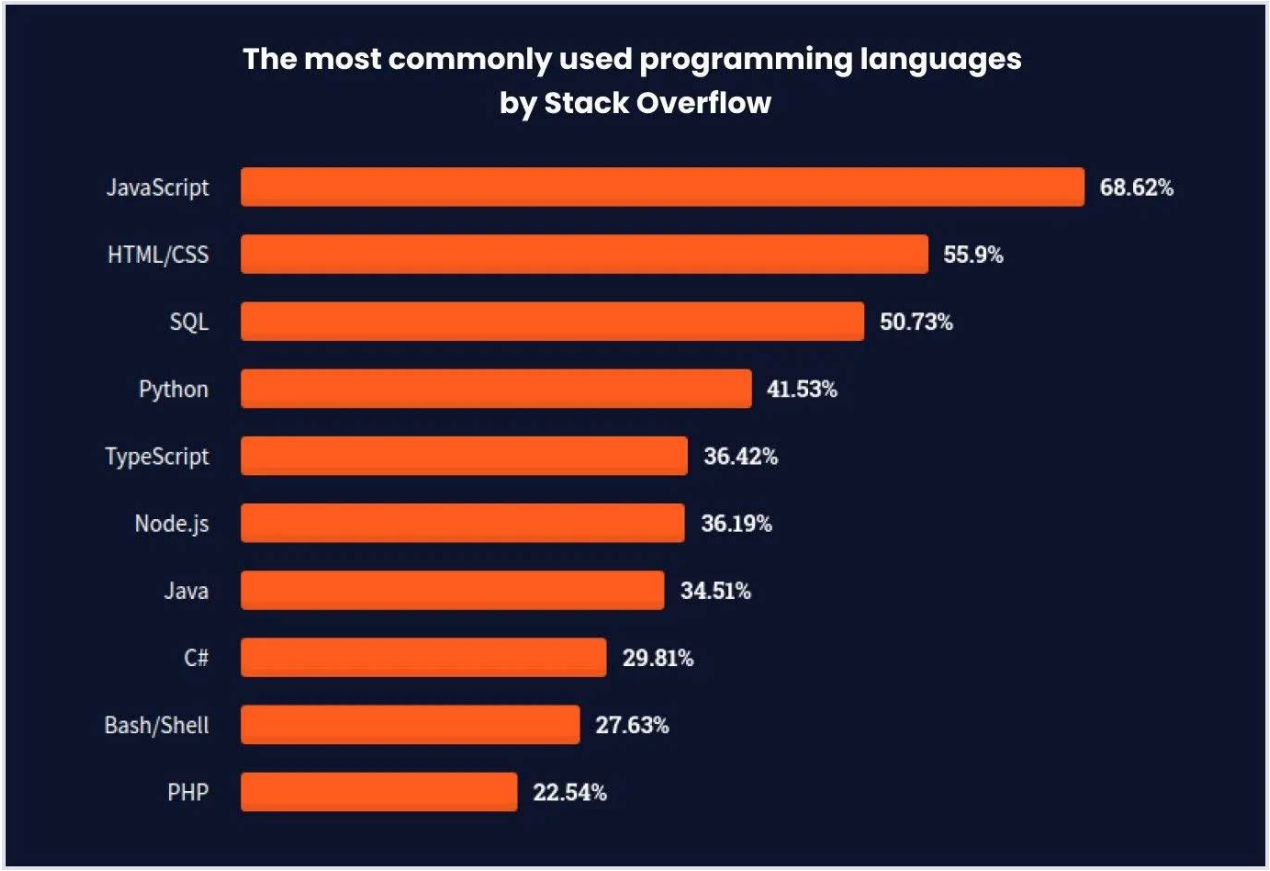
- The most popular programming languages used in blockchain development



https://www.freecodecamp.org/news/the-most-popular-programming-languages-used-in-blockchain-development-5133a0a207dc/

Top programming languages to learn in 2022

Top programming languages to learn in 2022



https://insights.stackoverflow.com/survey/2021#most-popular-technologies-language-prof

블록체인 특성

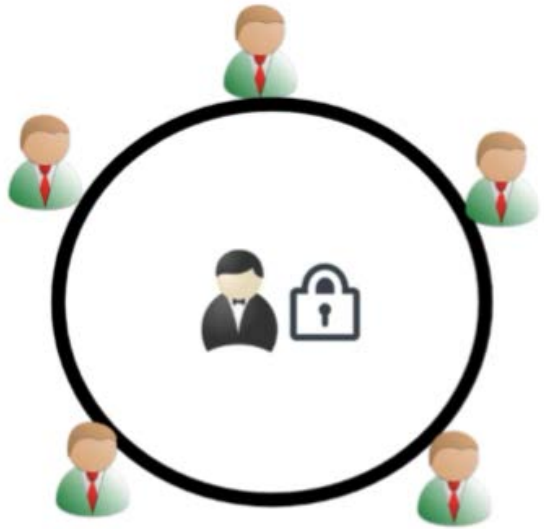
■ 블록체인 특성

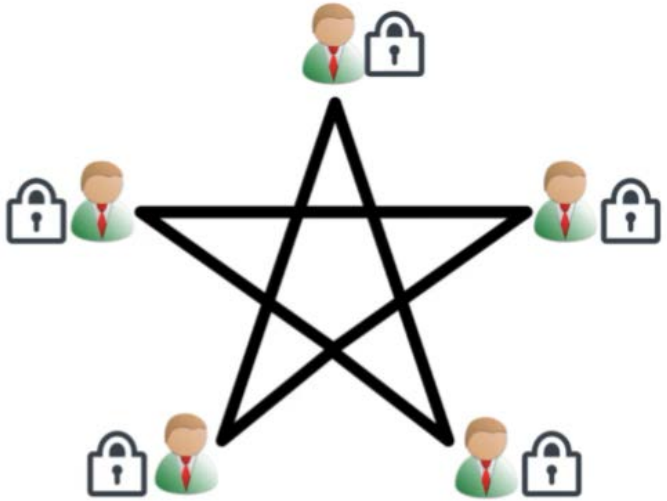
코딩으로 구현해보면 가장 확실하게 이해할 수 있습니다.

중앙화 (기존 은행 네트워크)
중앙 기관이 모든 정보 관리
신뢰기관을 통한 데이터 관리

⇔

분산화 (블록체인 기반 네트워크)
모든 참여자가 모든 정보 관리
합의 프로토콜을 사용해 데이터 관리

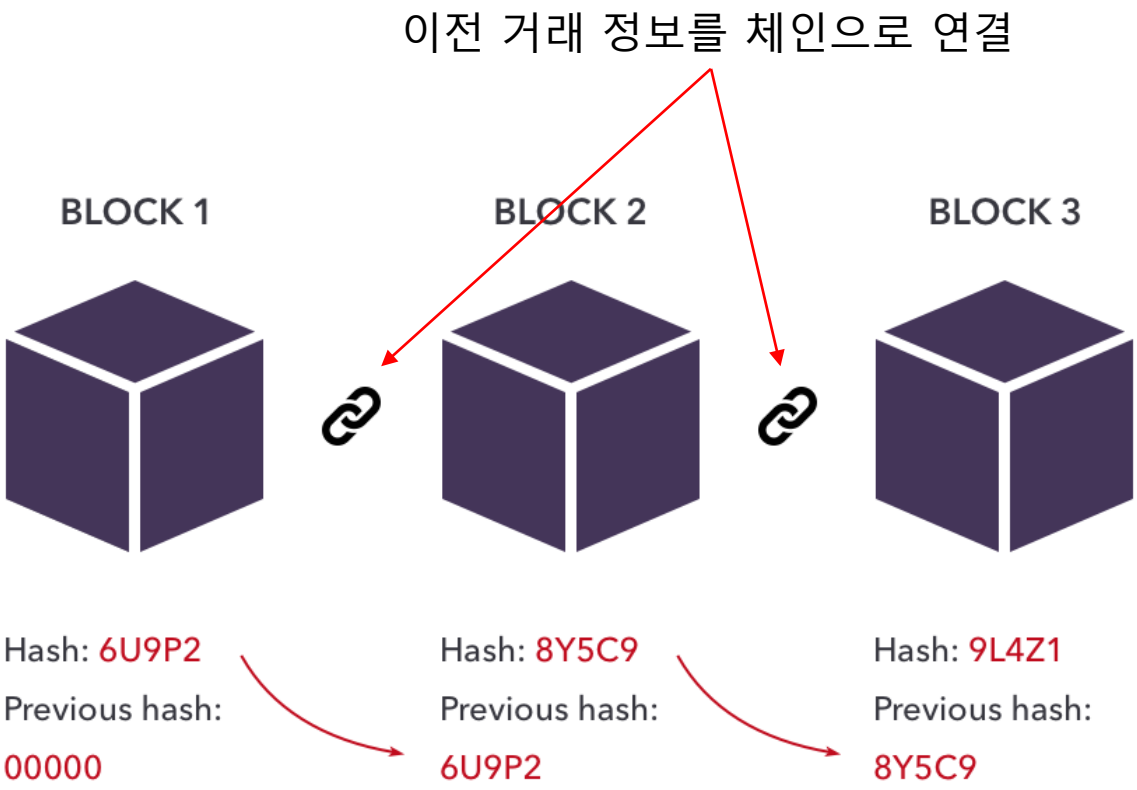




모든 기록은 중앙(은행)에 기록

중앙서버 없음. 모든 거래는 모든 참여자가 관리

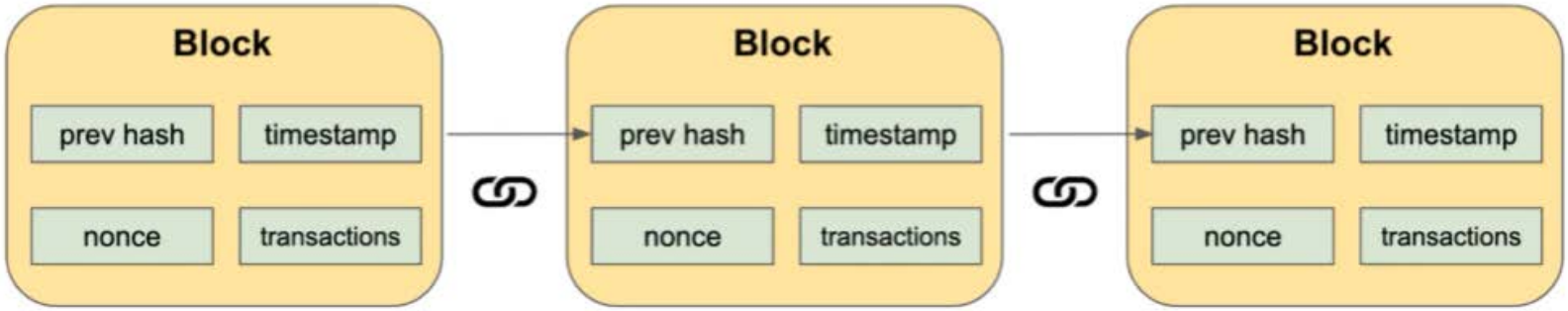
블록체인 핵심 기술



연속적인 해시 값을 적용
(해시 함수의 특성 활용)

핵심 기술

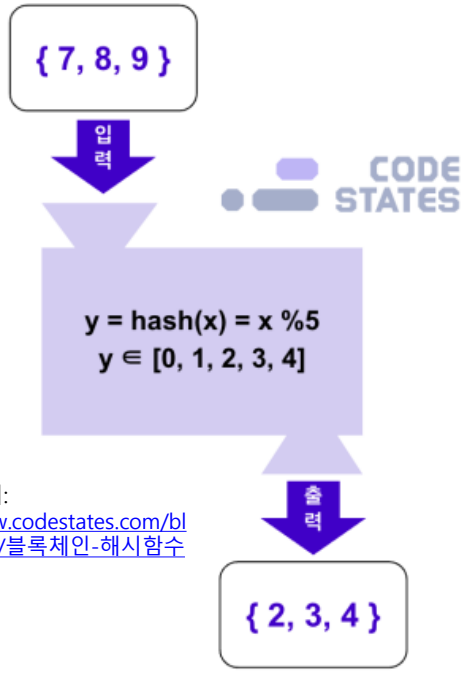
1. 해시 암호화 (Hash Cryptography)
2. 불변 원장 (Immutable Ledger)
3. 마이닝 (mining) & 넌스 (Nonce)
4. 분산 P2P (Distributed Peer to Peer)
5. 합의 프로토콜 (Consensus Protocol)



Hash (해시)

해시 함수

- 1950년대 등장 → 1970년대 (암호학 분야에서 연구) → 1990년대 (다양한 알고리즘 MD-5, SHA-256 등)
- 나머지 연산 (% operator in python)
- 임의 길이의 입력 → 고정된 길이로 출력



이미지 출처:
https://www.codestates.com/blog/content/블록체인-해시함수

Hash Function 특징

1. **단방향성 (One Directional)**
2. **해시 충돌(Collision)**
3. **고정된 길이 출력 (Fixed Length of Output)**
   (대용량 데이터 무결성 검증)

입력 값으로 출력을 쉽게 생성
출력 값으로 입력을 추론하는 것은 매우 어려움

Blockchain 적용

- **Transaction 검증**
- **Block 검증**
- **Merkle Tree 검증**

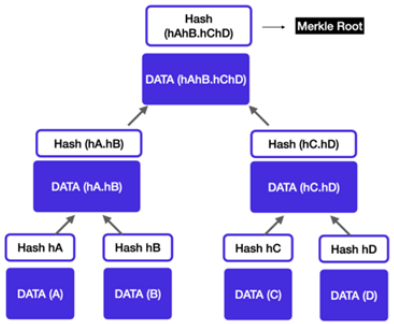
SHA-256

- **SHA는 미국 표준 기술 연구소(NIST)에서 개발한 해시 함수 알고리즘**
- **SHA-256**
  - 입력: 크기 제한 없음 (컴퓨터가 수용할 수 있는 용량이라면 뭐든 가능)
  - 출력: 256 비트(32 바이트)의 해시 값 생성
- **Python example**

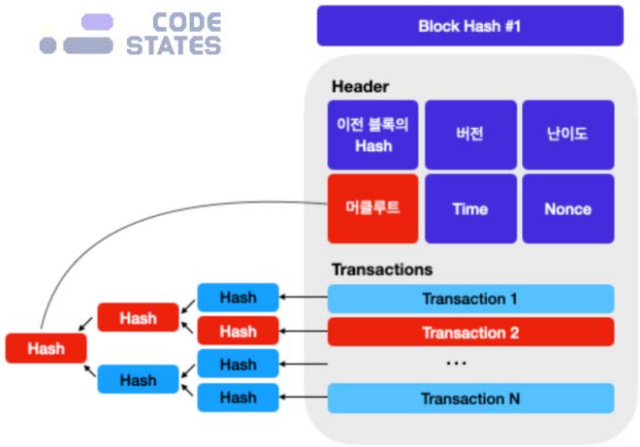
```
>>> import hashlib
>>> info = 'hello world'
>>> info.encode()
b'hello world'
>>> info_encode = info.encode()
>>> info_encode
b'hello world'
>>> hashlib.sha256(info_encode)
<sha256 HASH object @ 0x7fbd0213ab70>
>>> hashlib.sha256(info_encode).hexdigest()
'b94d27b9934d3e08a52e52d7da7dabfac484efe37a5380ee9088f7ace2efcde9'
>>> hashed = hashlib.sha256(info_encode).hexdigest()
>>> len(hashed)
64
```

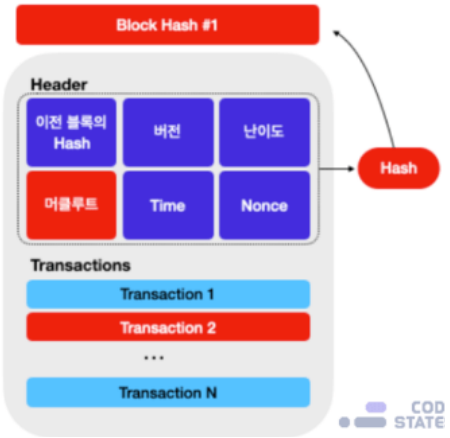
Merkle Tree (머클 트리)

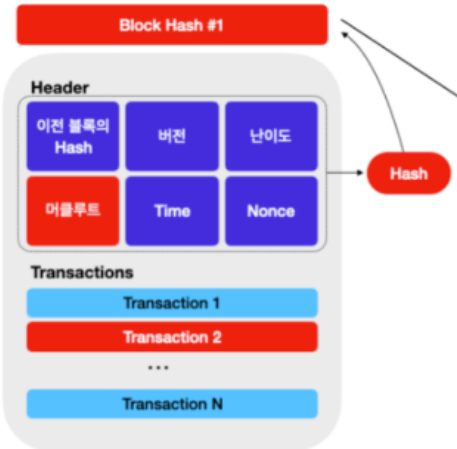
Binary Tree의 한 종류 (Hash Tree 라고도 부름)



리프 노드가 홀수일 경우
마지막 해시를 복사해요^^







이미지 출처:
https://www.codestates.com/blog/content/블록체인-해시함수

Merkle Path in Blockchain

Genesis Block

이미지 출처: https://namu.wiki/w/창세기

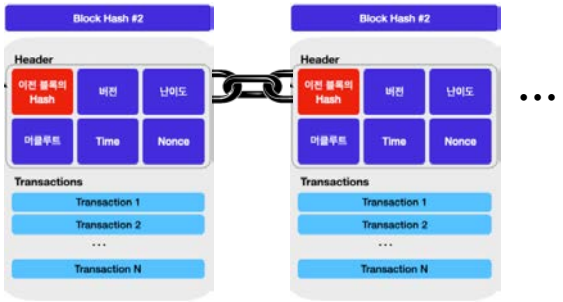
이미지 출처:
https://rnrministry.org/shop_view/?idx=16

Bitcoin Blockchain Size
(2023. 7월 11일 현재)
495.23 GB

https://ycharts.com/indicators/bitcoin_blockchain_size

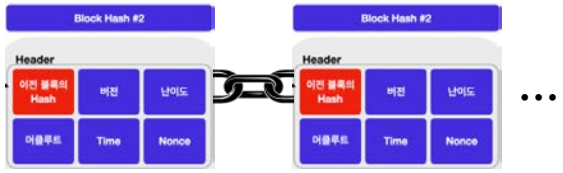
Full node

- Genesis Block부터 현재까지 모든 정보를 갖고 있는 노드(서버 혹은 컴퓨터)



Light Node

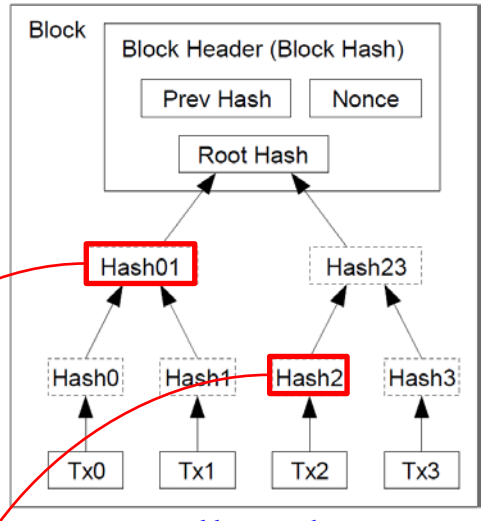
- 필요한 정보만 가지고 있는 노드 (서버 혹은 컴퓨터)



거래내역까지
보유하려면..ㅠㅠ

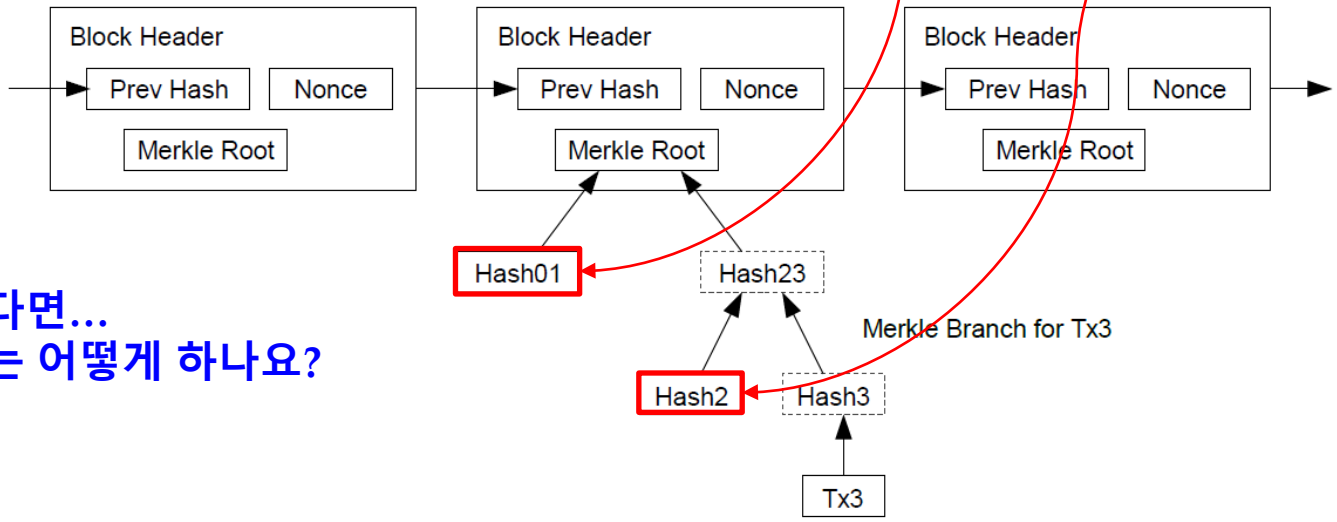
Validation of Transactions in Blockchain Network

- **일반 사용자 (대부분 고성능/대용량 컴퓨터 아님)**
  - 가장 긴 체인의 헤더만 보유
  - 검증이 필요한 Transaction (거래) 내용이 있다면?
    - 필요한 정보는 Full Node에게 요청하여 블록 유효성 검증
    - 검증할 Transaction이 속해 있는 블록 중에서
    - 검증에 필요한 정보만 요청



Full Node

Light Node



그렇다면...
우리는 어떻게 하나요?

다음 강의

➔ 개발환경 세팅

➔ 블록체인 클래스 작성

(blockchain.py)

수고하셨습니다 ..^^..