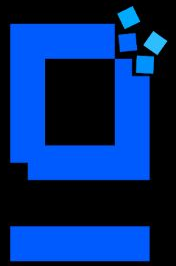
# Blockchain In Rust

## #01: Blocks & Hashing



geeklaunch

not a geek? start today!

# Before we start

Download and install Rust if you want to code along: https://www.rust-lang.org/

Optionally, you may also want to install Git: https://git-scm.com/

# Blockchains for Programmers

# Cryptocurrency Blockchains

Two main data structures

- The blocks in the blockchain (our sole focus in this video)
- The transactions within the blocks (future videos)

Ancillary data

- Wallets
- Addresses
- Balances
- Peers

# Generic Blockchains (with PoW support)

Blockchain ≈ chronological, sequential list of *blocks*

Blocks contain this information:

- **Index**: this block's location within the list of blocks
- **Payload**: any relevant information or events that have occurred for/in the block
- **Timestamp**: gives our blockchain a sense of time
- **Nonce**: special number used for mining (for PoW verification)
- **Previous block hash**: cryptographic fingerprint of previous block
- **Hash**: cryptographic fingerprint of all of the above data concatenated together

# Concept: Hashing

# What is Hashing?

In a nutshell, a hash algorithm consists of a set of irreversible computations that can be performed on a datum to generate a (usually) unique byte sequence.

```
MD5(“GeekLaunch”) = “e76485e55ba4c16aac30bd446b73d96e”

SHA-1(“GeekLaunch”) = “c333e84f729c67d6b591e056e1b51e0077a9c030”

SHA-256(“GeekLaunch”) =
“a17d5669f2148e2982baab7c0b4c7d81100c7cf52c45a8d7deb429aeba156ea6”
```

# What we will be using

# Rust Programming

What is Rust?

Rust is a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety. - https://www.rust-lang.org/

The Rust Programming Language (Free Book) - https://doc.rust-lang.org/book/2018-edition/index.html

# Java vs. Rust Overview

Java

- Compile once, run anywhere
- Requires virtual machine (JVM)
- Strongly typed
- Classical type system
- Taught at most universities
- Developed by Oracle

My three words: *Simple*, *Safe*, *Slow*

Famous uses: Android

Rust

- Interoperable with C/++
- Friendly, intelligent compiler
- Simple “garbage collection” rules
- “Pointers” are always safe
- Not taught at most universities (yet)
- Developed by Mozilla

My three words: *Complex*, *Safe*, *Fast*

Famous uses: Firefox

c.f. My three words for C: *Simple*, *Unsafe*, *Fast*

# Java Code vs. Rust Code

Java Code

```
class Block {
    public int index;
    public long timestamp;
    public BlockHash prevBlockHash;
    public BlockHash hash;
    public String payload;

    public Block (int index, long timestamp, BlockHash prevBlockHash, String payload) {
        this.index = index;
        this.timestamp = timestamp;
        this.prevBlockHash = prevBlockHash;
        this.hash = new BlockHash(new int[] {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0});
        this.payload = payload;
    }
}
```

Rust Code

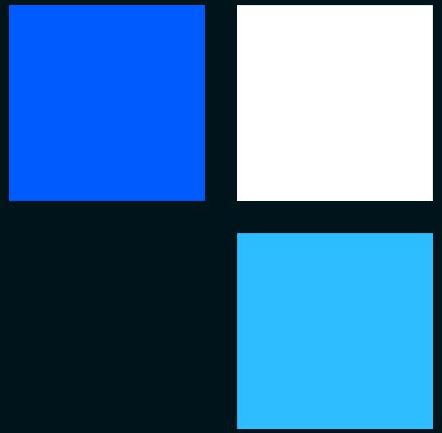
```rust
pub struct Block {
    pub index: u32,
    pub timestamp: u64,
    pub prev_block_hash: BlockHash,
    pub hash: BlockHash,
    pub payload: String,
}

impl Block {
    pub fn new (index: u32, timestamp: u64, prev_block_hash: [u8; 16], payload: String) -> Self
{
        Block {
            index,
            timestamp,
            prev_block_hash,
            hash: [0; 16],
            payload,
        }
    }
}
```

JS Code

```
class Block {
    constructor (index, timestamp, prevBlockHash, payload) {
        this.index = index;
        this.timestamp = timestamp;
        this.prevBlockHash = prevBlockHash;
        this.payload = payload;
        this.hash = Array(16).fill(0);
    }
}
```

But JavaScript is too high-level...



Let’s get coding!

# Where to start

Get the project starter code from the GitHub repository GeekLaunch/blockchain-rust on tag `start-here`

https://github.com/GeekLaunch/blockchain-rust/tree/start-here

```
$ git clone https://github.com/GeekLaunch/blockchain-rust.git

$ git checkout start-here
```