Blockchain

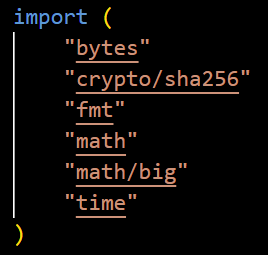
A **Blockchain** is a growing list of records, called *blocks* that are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data.

Algorithm of Blockchain

The following steps are involved in the creation of this Blockchain:

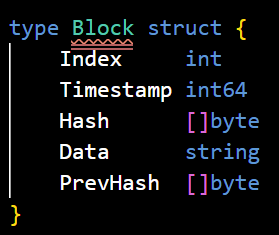
**1) Importing Packages:** To start the building process of our blockchain, we firstly need to import some necessary packages which will help to provide useful functions for our code. Some of the packages are:

**“bytes” | “crypto/sha256” | “fmt” | “math” | “math/big” | “time”**

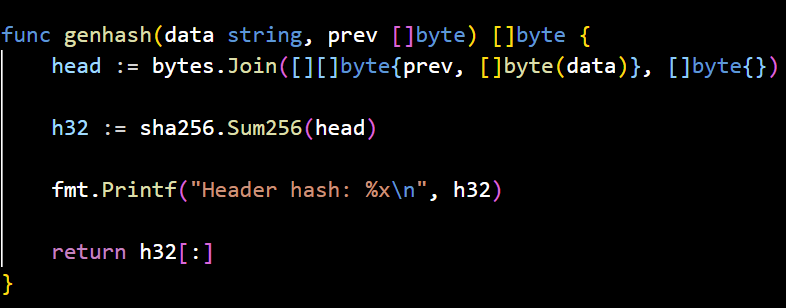
****

* **fmt:** It stands for the format package. This package allows to format basic strings, values, or anything and print them or collect user input from the console, or write into a file using a writer or even print customized fancy error messages, This package is all about formatting input and output.
* **crypto/sha256:** SHA-256 belongs to the SHA-2 family of cryptographic hashes. It produces the 256 bit digest of a message. SHA stands for Secure Hash Algorithm. Return a fresh instance of the hash object.
* **bytes:** package bytes implements functions for the manipulations of byte slices. It is analogous to the facilities of the strings package.

**2) Creating Blocks:** The blocks are the building blocks of any blockchain which contain the main data that has been entered and need to be kept secured inside a blockchain. Along with the primary data, the block also contains the hash value of the current block as well as that of the previous block. It also includes the timestamp of the transactions.



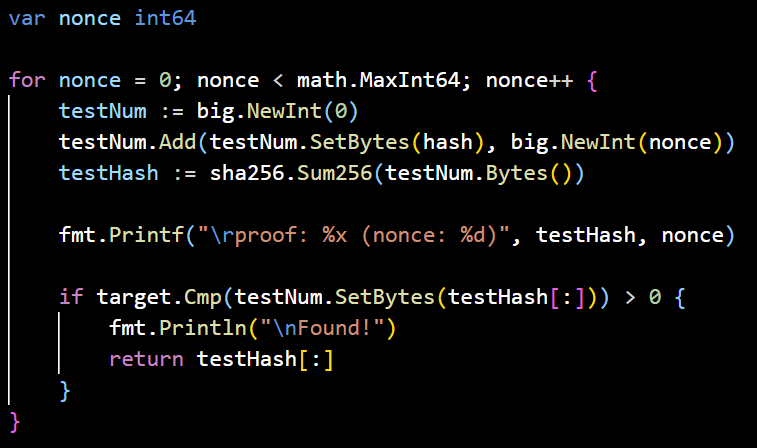
**3) Hashing:** A hash function is a mathematical function takes an input string of any length and converts it to a fixed-length output string. The fixed-length output is known as the hash value.



For the purpose of creating this blockchain, we are using the SHA256 algorithm to generate hashes for our blocks.

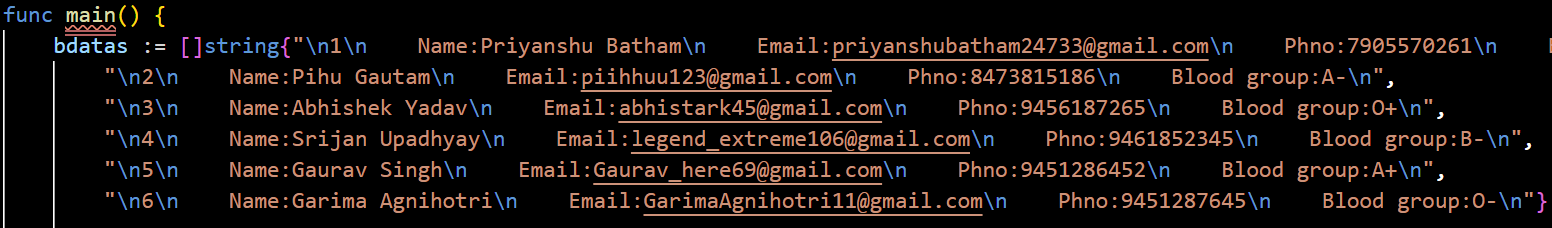
**4) Setting PoW:** Proof of work (PoW) is a decentralized consensus mechanism that requires members of a network to expend effort solving an arbitrary mathematical puzzle to prevent anybody from gaming the system.

In this blockchain we will be setting a high difficulty level of 15 in order to improve the security functionality of our system. For this purpose we will be using a variable nonce which can easily describe itself as “a random or semi-random number that is generated for a specific use, the term stand for ‘number used once’ and is commonly referred to as a cryptographic nonce.



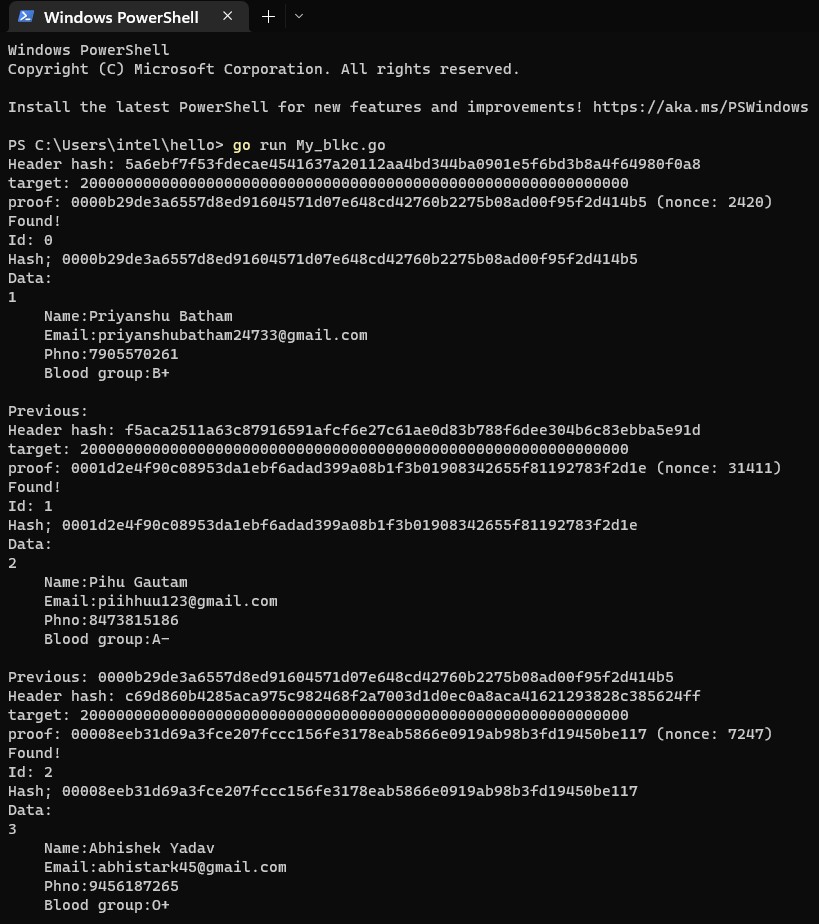
**5) Data Entry:**  So far we have designed the structure of the blockchain and have set the logics of PoW and hashing also. Now this is the stage where we are ready to record or store the data inside of our own blockchain. For now we will be creating six blocks which will store the following data of the users:

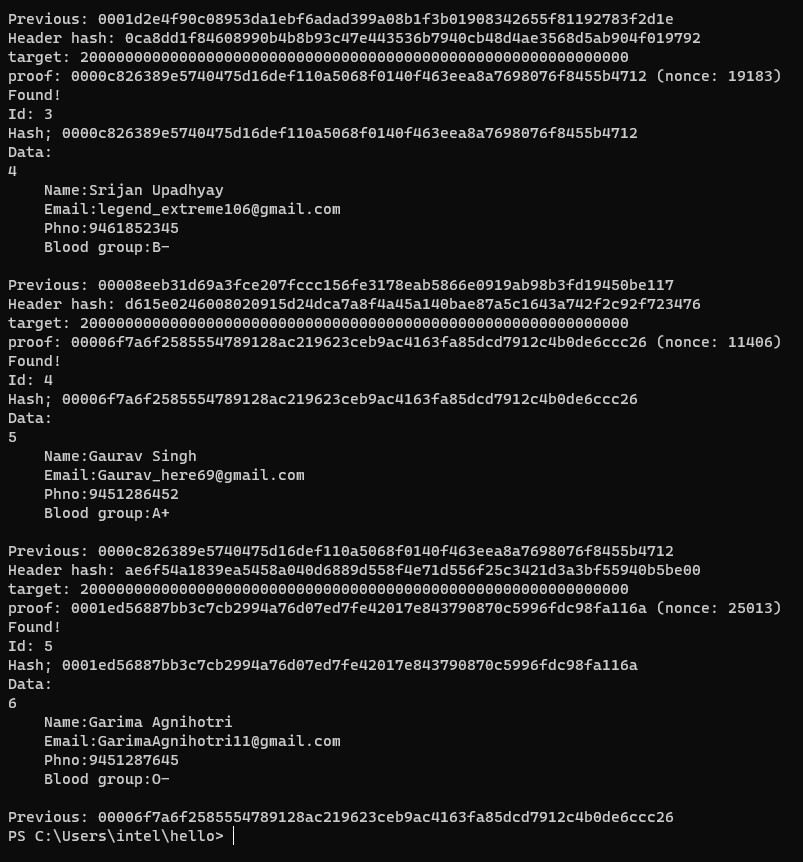
* Name:
* Email:
* Phno:
* Blood Group



**6) Run the code:** Finally our blockchain is successfully created and this sums up the process of building our own blockchain using the Go language. Now to check the proper working of our blockchain we will be executing the code using the “go run My\_blkc.go” command in any preferred terminal.

**The Output Window:**



****