**Lesson Plan – Checking for Broken Chain**

**Learning objectives -**

1. **Revise concepts of adding blocks to blockchain from previous class**
2. **Identify if your blockchain is broken**

**Materials required -**

**repl.it login credentials**

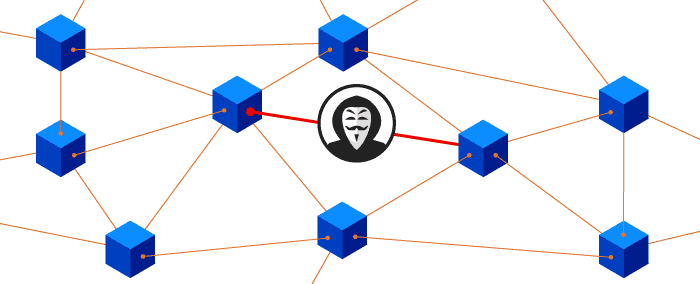
**How Hashing Maintains the Blockchain's Integrity**

In the previous lesson, we briefly touched upon the idea of hashing — generating a random string of characters from a given input. Let’s go a step further and explore why hashing is so fundamental to the design of the Blockchain.

In a blockchain, each block has a unique hash and a link to the previous block’s hash. If a participant decides to tamper with a block by modifying the transactions, the block’s unique hash gets changed. However, the following block does not update to reflect this change — it still points to the previous block’s hash. Thus, there is a mismatch between hashes and the link between blocks is broken. This results in an invalid copy of the blockchain.

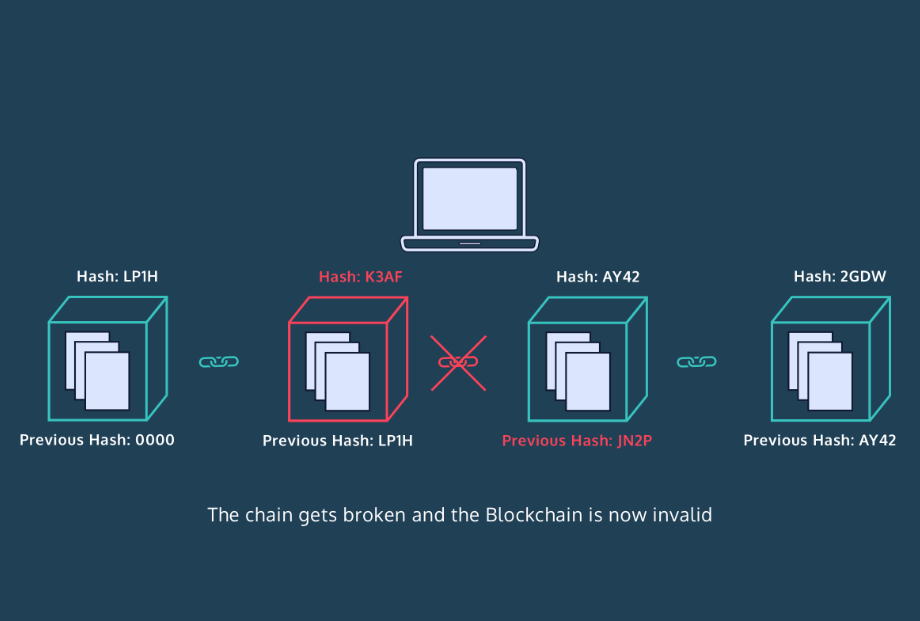
In this way, the records in the blockchain cannot be altered. In other words, the records are said to be immutable.

But what if someone tampers with a block and recalculates the hashes for every single block? Does hashing still guarantee that the blockchain is fully secure? Let’s find out in the next exercise.



**Key Terms we will come across:**

* **Hashing:** Generating a random string of characters from a given input.
* **Immutable:** Something whose records can’t be changed.
* **Encoding :** Encoding is the process where we transform original text format to new non-readable format. Just like hashing.



Hashing helps to maintain the integrity of the blockchain. In this exercise, we will see this in action. Let’s write a .validate\_chain() method that checks to see if blocks are linked to each other properly.

In order to validate the entire blockchain, we must loop through each of the blocks stored inside the blockchain itself. Then, we will check through each of the blocks to ensure that the previous hash value matches with the hash value inside our previous block.

Validation reference : <https://www.euromoney.com/learning/blockchain-explained/how-transactions-get-into-the-blockchain>

Introducing the lesson/project including the concepts (Time - 45 min)

Link to Repl it Project :

<https://replit.com/join/qaxkxdtqeo-priyankajetlea1>

https://replit.com/@ArindamJetLearn/Lesson7CheckingForBrokenChain#main.py

**Homework –**

# Write a program to find the sum of all items in a dictionary

<https://www.geeksforgeeks.org/python-program-to-find-the-sum-of-all-items-in-a-dictionary/>

Write a program to

* [Get Current Timestamp](https://pynative.com/python-timestamp/#h-get-current-timestamp) using
  + [Datetime to Timestamp](https://pynative.com/python-timestamp/#h-datetime-to-timestamp)
  + [Get Timestamp Using time Module](https://pynative.com/python-timestamp/#h-get-timestamp-using-time-module)
  + [Get Timestamp Using calendar Module](https://pynative.com/python-timestamp/#h-get-timestamp-using-calendar-module)

Also convert timestamp to string and datetime (format)

https://pynative.com/python-timestamp/