LESSON 6: BLOCKS, TRANSACTIONS, AND HASHES

A Prison Professors Self-Directed Course

Estimated time: 65–100 minutes

Educational Boundary:

> *This lesson explains technology. It does not recommend any network, company, product, or digital asset. It does not provide individualized financial, investment, legal, or tax advice. You do not need internet access, an account, a wallet, or a digital asset to complete the lesson.*

OPENING GUIDANCE

Lesson 5 introduced a blockchain as a shared digital ledger. In this lesson, we will slow the process down and examine how a proposed transaction can become part of a block and how hashes help connect one block to the next.

When I encounter a technical subject, I find it useful to separate the process into smaller questions. What information is being proposed? Who or what checks it? How is it grouped? What evidence would reveal a later change? I encourage you to use that method here. You do not need to memorize a formula or become a cryptographer. Your goal is to build a mental model that you can explain accurately, test against an example, and improve through writing.

Careful wording is part of the lesson. People often say that blockchain records are "immutable," meaning they cannot change. That single word may hide important conditions and tradeoffs. By the end of the lesson, you should be able to explain why tamper-evident and tamper-resistant are usually more responsible descriptions.

PURPOSE AND ESSENTIAL QUESTION

Lesson 6 explains how transaction information moves toward an accepted block, what a cryptographic hash does, and why a chain of hash references can make altered records detectable.

Essential Question:

» How do transactions, blocks, and hashes work together to create an ordered record, and what does that structure prove—or fail to prove?

LEARNING OBJECTIVES

After completing this lesson, you should be able to:

- Describe the basic path from a proposed transaction to a recorded block.
- Define a cryptographic hash in beginner-friendly language.
- Explain how linked hashes help make altered records detectable.
- Distinguish tamper-evident from impossible to change.
- Identify one thing a hash can show and one thing it cannot prove.

KEY TERMS

TRANSACTION

Definition: A proposed action or change that a system may verify and add to its record. A transaction does not have to involve money.

Sample sentence: Recording that an organization issued a digital credential can be treated as a transaction.

BLOCK

Definition: A group of accepted records or transactions stored together in a blockchain.

Sample sentence: The validator included several accepted transactions in the next block.

HASH

Definition: A fixed-length result produced when a hash function processes data. A cryptographic hash is designed so that a change to the input produces a different result and so that reversing the result or finding two matching inputs is extremely difficult.

Sample sentence: The learner compared the two hashes to see whether the recorded data had changed.

CRYPTOGRAPHY

Definition: The study and use of mathematical methods that help protect information and support functions such as confidentiality, integrity, authentication, and verification.

Sample sentence: A blockchain uses cryptography as one part of its security design, but cryptography does not remove the need for sound rules and responsible participants.

TIMESTAMP

Definition: Time-related information attached to a record or block according to a system's rules.

Sample sentence: The block included a timestamp that helped place it within the network's sequence.

TAMPER-EVIDENT

Definition: Designed so that an unauthorized or unexpected change leaves evidence that can be detected.

Sample sentence: The linked references made the record tamper-evident because changing an earlier block caused later references to stop matching.

MAIN LESSON

BEGIN WITH A PROPOSED TRANSACTION

A transaction begins as a proposed change. In a payment system, it might propose transferring a digital asset. In a credential system, it might propose recording that an authorized organization issued or revoked a credential. In a supply system, it might propose changing the recorded quantity of an item.

The word transaction describes a system event, not necessarily a purchase. The important question is what change the system is being asked to recognize.

A proposed transaction is not automatically an accepted fact. The system first applies its rules. Depending on the network, those rules may check the format, authorization, digital signature, available balance, sequence number, or other conditions. A transaction that fails the applicable checks should not enter an accepted block.

Passing those checks still does not prove that every statement about the outside world is true. If an authorized person submits an incorrect course-completion record, the transaction may be technically valid even though the underlying claim is false. Blockchains can check information that exists within their rules. They cannot independently observe every real-world event.

THE BASIC PATH FROM TRANSACTION TO BLOCK

Different blockchain networks use different procedures, but a beginner can understand the general path in five stages.

1. Proposal: A participant creates and submits a transaction in the form required by the network.
2. Initial checking: Network software checks whether the transaction satisfies basic rules.
3. Selection and grouping: A miner, validator, or other authorized process selects acceptable transactions and groups them into a proposed block.
4. Network agreement: Participants apply the network's consensus rules to decide whether the proposed block becomes part of the accepted history.
5. Recording and distribution: The accepted block is linked to the earlier block and shared across the network's copies of the ledger.

Lesson 7 will examine nodes, validators, miners, and consensus mechanisms more closely. For now, notice that a transaction does not move directly from one person's request into permanent history. A system of checks, roles, and rules stands between the proposal and the accepted block.

There may also be a delay. A transaction can wait before selection. A block can be proposed before the network treats it as final. Some networks may temporarily disagree about the most recent history and later settle on one version under their rules. Words such as pending, included, confirmed, and finalized can describe different stages. They should not be treated as exact synonyms.

WHAT A HASH DOES

A hash function accepts input data and produces an output called a hash, hash value, or message digest. For a particular hash algorithm, the output has a fixed length even when the input varies in length.

Imagine placing a document into a machine that produces a short code. The same document processed by the same machine produces the same code. Change one character in the document, and the code changes. That analogy is useful, but it has limits. A real cryptographic hash function uses mathematics, not a physical machine, and the output may appear as a long string of numbers and letters.

Secure cryptographic hash functions are designed around several properties:

» Deterministic: The same input processed by the same hash function produces the same output.

- Fixed-length output: A chosen algorithm produces an output of a set length even when inputs have different lengths.
- Sensitive to change: A small change in the input should produce a different-looking output.
- One-way in practice: It should be computationally infeasible to work backward from the hash and recover the original input.
- Collision-resistant: It should be computationally infeasible to find two different inputs that produce the same hash.

Collision-resistant does not mean that matching outputs are mathematically impossible. A fixed-length output has a limited number of possible values, while possible inputs are far more numerous. Secure algorithms are designed to make a useful collision extraordinarily difficult to find. The strength also depends on using an appropriate algorithm and implementing it correctly.

A HASH IS NOT ENCRYPTION

Hashing and encryption are related to cryptography, but they perform different jobs.

Encryption transforms readable information into a protected form that an authorized person can reverse with the proper key. A cryptographic hash is intended to be one-way. It produces a digest used for comparison or as part of another security process. It is not a container that someone later opens to recover the original message.

A hash also does not automatically hide the input. If someone knows a small set of possible inputs, that person may hash each possibility and compare the results. Sensitive information should not be placed on a public blockchain merely because the system stores or references a hash. Privacy decisions require a broader analysis.

HOW HASHES LINK BLOCKS

In a simplified blockchain model, each block includes transactions, time-related information, and a reference derived from the previous block's data. That reference is commonly a cryptographic hash.

Suppose Block 20 contains a hash of Block 19. Block 21 then contains a hash of Block 20. If someone changes a transaction in Block 19, the hash calculated for Block 19 changes. The reference stored in Block 20 no longer matches the altered Block 19. That mismatch creates evidence that the chain has been changed.

Recalculating one hash would not be enough to hide the alteration. The person would also need to rebuild the later references and overcome the network's rules for accept-

ing history. As more blocks are added and more independent participants maintain the accepted chain, rewriting older history may become increasingly difficult.

The hash link provides tamper evidence. The network, consensus process, participant independence, software security, and incentives help provide tamper resistance. A hash alone does not make a record impossible to change.

WHAT A TIMESTAMP CAN AND CANNOT SHOW

A timestamp helps place a transaction or block in time according to a system's rules. It can support ordering and help participants evaluate whether information fits the expected sequence.

A timestamp should not be treated as perfect proof that an outside event occurred at an exact moment. Networks differ in who supplies time information, how much variation their rules permit, and when a proposed block becomes accepted. The timestamp belongs to the technical record. Additional evidence may be needed to prove when a real-world event happened.

WHY "IMMUTABLE" REQUIRES CAREFUL EXPLANATION

The word immutable literally suggests that something cannot change. In ordinary blockchain discussion, people often use it more loosely to mean that accepted history is difficult to alter without detection.

Several situations show why precision matters:

- A network may correct an error by adding a later transaction rather than erasing the earlier one.
- Recent blocks may be reorganized if the network temporarily received competing versions of history.
- Participants may adopt new software rules or split into separate networks.
- A private or permissioned system may give administrators defined powers to change or replace records.
- A poorly secured or highly concentrated network may be more vulnerable to coordinated alteration.

For those reasons, tamper-evident describes the evidence left by a change, while tamper-resistant describes the difficulty of making an accepted change succeed. Neither phrase claims that alteration is impossible under every condition.

A BRIEF APPLIED EXAMPLE: BNB SMART CHAIN

As of July 20, 2026, BNB Chain's official documentation describes BNB Smart Chain as a network in which validators produce blocks under its consensus rules. Its developer documentation also describes methods for retrieving transactions from a block and locating transaction data by a transaction hash.

This provides a practical use of hash language. A transaction hash can act as an identifier that lets software or a block explorer locate the transaction record. A block has its own hash-based references within the chain. These identifiers help people compare records and follow the sequence.

The transaction hash does not prove that every outside statement connected with the transaction was accurate. It also does not, by itself, prove that the transaction succeeded, became final, or was authorized by the right real-world person. A careful reviewer examines the transaction data, its status or receipt, the block, the network, and the surrounding evidence.

Mentioning BNB Smart Chain is an educational example. It is not an endorsement, an invitation to acquire BNB, or a request to conduct a transaction.

WHAT HASHES CANNOT PROVE

Hashes are useful for integrity checks, but their limits matter.

A matching hash can show that the compared data produced the same digest under the same function. It does not prove:

- That the original information was truthful.
- That the person who submitted it had lawful authority.
- That a record was fair, complete, private, or ethically collected.
- That the software or hash function was implemented correctly.
- That the broader network is secure or decentralized.
- That an outside event occurred exactly as described.

Other tools and procedures address those questions. Digital signatures can help check authorization. Governance rules can assign responsibility. Privacy controls can limit access. Audits and source documents can help verify outside facts. No single technical feature replaces the entire system of judgment.

APPLIED SCENARIO: A FOOD-DISTRIBUTION RECORD

Three nonprofit warehouses coordinate deliveries of shelf-stable food. Each warehouse submits daily records showing the number of cases received and distributed.

The organizations use a shared chain of blocks. Each new block contains accepted records and a hash reference to the previous block.

An auditor later receives two copies of the history. In the first copy, Block 8 says that Warehouse B received 80 cases. In the second copy, the entry says 60 cases. The two versions produce different hashes, and the later references match only the first copy.

The hash comparison reveals that one version was changed. It does not prove that 80 cases actually arrived. The original delivery receipt may have been wrong, the data-entry person may have made a mistake, or an authorized person may have submitted false information.

Reasoned conclusion: The linked hashes help the auditor detect a disagreement and identify which history matches the later chain. The organizations still need source documents, authorized submitters, correction procedures, and accountability. Technical integrity and factual truth are related, but they are not the same.

BENEFITS, LIMITATIONS, RISKS, AND MISCONCEPTIONS

POSSIBLE BENEFITS

- Hashes provide a practical way to compare data and detect change.
- Linked hashes help preserve an ordered relationship among blocks.
- Multiple participants can compare the chain and identify a conflicting version.
- Later blocks can make rewriting older accepted history more difficult.

IMPORTANT LIMITATIONS AND RISKS

- A hash cannot prove that the original input was true or lawfully authorized.
- Weak or outdated hash functions may not provide the expected protection.
- Poor implementation can undermine a strong cryptographic design.
- Public records or predictable inputs can create privacy risks even when hashes are used.
- Network concentration, software defects, governance failures, or attacks can weaken tamper resistance.
- A timestamp may not prove the exact time of an outside event.

COMMON MISCONCEPTIONS

- “A transaction always means a payment.” A transaction can represent many proposed changes to a system’s record.

- » “A transaction is permanent as soon as it is submitted.” It may be pending, rejected, included, confirmed, reorganized, or finalized depending on the network and stage.
- » “A hash encrypts the original information.” Hashing produces a one-way digest; encryption is designed to be reversed with the proper key.
- » “Different inputs can never share a hash.” Secure functions make useful collisions extremely difficult to find, not mathematically impossible.
- » “A matching hash proves the information is true.” It supports an integrity comparison, not verification of every outside fact.
- » “Immutable means nobody can ever change anything.” Responsible analysis examines the network’s rules and uses more precise terms such as tamper-evident and tamper-resistant.

OFFLINE EXERCISE: BUILD A PAPER HASH CHAIN

Estimated time: 20–25 minutes

Materials: Separate paper and a pencil. A calculator is optional.

This exercise uses a simple two-digit check code to demonstrate linked references. The code is not secure and is not a real cryptographic hash. It is intentionally simple enough to calculate on paper.

USE THE TOY RULE

FOR EACH BLOCK, CALCULATE:

Toy hash = previous hash + block number + quantity + number of words in the transaction. Keep only the final two digits.

Count Receive, Issue, the number, and the item as words. For example, “Receive 5 books” contains three words. Begin with previous hash 00.

BUILD THE CHAIN

- » Block 1

Previous hash: 00

Transaction: Receive 5 books

Calculation: 00 + 1 + 5 + 3 = 09

Toy hash: 09

- » Block 2

Previous hash: 09

Transaction: Issue 2 books

Calculation: 09 + 2 + 2 + 3 = 16

Toy hash: 16

» Block 3

Previous hash: 16

Transaction: Issue 1 book

Calculation: 16 + 3 + 1 + 3 = 23

Toy hash: 23

Copy the three blocks onto your paper. Draw an arrow from each toy hash to the next block's previous-hash line.

ALTER AN EARLIER RECORD

1. Change Block 1 from "Receive 5 books" to "Receive 6 books."
2. Recalculate Block 1. Its toy hash changes from 09 to 10.
3. Compare the new Block 1 hash with the previous hash written in Block 2. They no longer match.
4. Change Block 2's previous hash to 10 and recalculate its toy hash. It changes from 16 to 17.
5. Compare the new Block 2 hash with the previous hash in Block 3. They no longer match.
6. Change Block 3's previous hash to 17 and recalculate its toy hash. It changes from 23 to 24.

REFLECT ON THE SIMULATION

Write brief responses to these questions:

» What evidence revealed the change to Block 1?

» Why did changing an earlier record affect later references?

» Did the toy hash prove whether five or six books were actually received? Why or why not?

» What would a person have to change to make every link match the altered record?

» What important parts of a real blockchain did this paper activity leave out?

Important limitation of the simulation: This toy rule is a checksum-like classroom device. It has frequent collisions, is easy to reverse, and provides no security. A real blockchain uses secure cryptographic hash functions, network software, validation, digital signatures, and consensus rules. The exercise demonstrates linked references and tamper evidence only.

KNOWLEDGE CHECK

MULTIPLE CHOICE

1. Which sequence best describes the general path of a blockchain transaction?
   a. Proposal, basic checking, selection into a proposed block, network agreement, and recording
   b. Recording, deletion, advertising, and automatic proof of truth
   c. Payment, bank approval, password recovery, and printing
   d. Hashing, encryption, guaranteed acceptance, and permanent secrecy
2. Which statement best describes a cryptographic hash?
   a. A reversible container that hides a message until a key opens it
   b. A fixed-length result used to represent input data and help detect change
   c. A guarantee that the input came from an honest person
   d. A timestamp that proves when an outside event occurred
3. Why does changing information in an earlier block affect later hash references?
   a. The later block contains a reference derived from the earlier block's data.
   b. Every participant stores information in one physical computer.
   c. Hashes automatically correct false information.
   d. A block can contain only one transaction.

TRUE OR FALSE

» If a transaction's hash matches the recorded hash, the match proves that every statement in the transaction was true and lawfully authorized.

SHORT EXPLANATION

» In two or three sentences, explain the difference between saying a blockchain is tamper-evident and saying it is impossible to change.

SUMMARY AND PRACTICAL TAKEAWAYS

- A transaction is a proposed change, not automatically an accepted fact.
- Transactions pass through checks and network rules before becoming part of an accepted block.
- A cryptographic hash produces a fixed-length digest that changes when the input changes.
- Hashing is not the same as encryption and does not prove that the original information was true.
- Blocks use hash-based references to connect the accepted history in sequence.
- Altering an earlier block changes its hash and causes later references to stop matching unless the later chain is rebuilt and accepted.
- Tamper-evident means a change leaves detectable evidence. Tamper-resistant means the system makes accepted alteration difficult. Neither means that change is impossible under every condition.

The practical skill is learning to ask two separate questions: Does the record show evidence of alteration, and was the information accurate and authorized when it entered the system?

PROFILE JOURNAL ASSIGNMENT

Write one coherent journal entry of approximately 300–400 words. Do not submit a disconnected list of answers. Use the prompts below to organize a beginning, middle, and conclusion:

- Explain the path from a proposed transaction to a recorded block in your own words.
- Define a hash and describe how one block can use a hash to refer to an earlier block.
- Explain why a changed record can cause later references to stop matching.
- Distinguish tamper-evident or tamper-resistant from impossible to change.
- Identify one thing a matching hash can show and one thing it cannot prove.
- Connect careful wording to the critical-thinking or communication skills you are developing.

Your response should demonstrate your reasoning. You are not being asked to calculate a real cryptographic hash, create an account, open a wallet, purchase an asset, or conduct a transaction.

SUBMISSION REMINDER

Write your response on separate paper or in an approved institutional messaging system. Include your name or approved Profile identifier, Lesson 6: Blocks, Transactions, and Hashes, and the date you completed the entry.

Use one of the established Prison Professors Profile methods, subject to your facility's rules:

1. Send it by institutional email to Playbook@PrisonProfessors.org. Suggested subject: Web3 Lesson 6 — [date completed].
2. Send it by postal mail to: Prison Professors, 1205 BMC Drive, Suite 706, Cedar Park, TX 78613.
3. Send it to an approved family member or supporter who can enter it on your Profile at PrisonProfessors.org.

Keep a copy when circumstances permit. Never include passwords, private keys, seed phrases, authentication codes, account numbers, real wallet addresses, or other sensitive credentials.