

Bitcoin Transaction Demo Kit

Live-event kit for Block 7: Transactions - Send & Receive.

Audience format: 8 tables x 10 people = 80 participants

Session length: 25-35 minutes inside a larger event, or 45 minutes as a standalone activity

Core idea: people physically create transactions, nodes validate them, the mempool orders them by fees, miners compete to create blocks, and everyone updates the ledger.

Facilitator Review of the Original Game Plan

The concept is strong. The practical demo is exactly the right improvement over a purely explanatory setup because people feel the system: waiting in the mempool, fee competition, validation, mining randomness, and ledger updates.

Recommended fixes before using it live:

- Use "signature cards," not actual locked boxes for every transaction.

Lockable boxes are memorable but slow for 80 people. Use one fake private-key envelope/stamp per wallet. A signed transaction must include the wallet's matching stamp/sticker/secret word. Keep one lockbox at the front as a quick self-custody prop.

- Separate validation from mining clearly.

Nodes validate transactions before mempool entry. Nodes then validate the winning block again before it is added, mainly to catch stale balances and double-spends.

- Track balances with one table ledger, not independent node ledgers only.

Each node keeps a sheet, but table captain also maintains the "public ledger" so disputes don't eat the round.

- Treat fees as a visible market.

The mempool role should physically sort transactions from highest fee to lowest and explain: "Miners usually pick the most profitable transactions first."

- Adjust dice difficulty for live pacing.
- Low difficulty: roll at least one 6 with 3 dice. Expected win is very fast.
- Medium difficulty: roll at least two 6s with 3 dice. Better for a 1-2 minute mining race.
- High difficulty: roll three 6s with 3 dice. This is a great demonstration of difficulty, but too slow for normal rounds. Use only as a 30-second "wow, blocks got harder" demo, then lower it or allow both miners to keep cumulative sixes.
- Make block rewards explicit as a coinbase transaction.

The miner does not just "get tokens"; the block contains a special reward transaction: NEW BTC -> Miner.

- Do not let wallets pre-check perfectly.

Allow one or two invalid transactions on purpose: insufficient funds, fake signature, low-fee stuck transaction, double spend. That makes nodes valuable.

Recommended Live Format for 8 Tables

Use a center-stage mirrored network. The moderator demonstrates one canonical transaction flow at the front of the room. Every table recreates the same process at the same time using its own cards, wallets, nodes, mempool, miners, and ledger.

This is stronger than letting every table improvise because:

- all 80 participants learn the same sequence
- the moderator can introduce one variation at a time
- table captains can check whether their table matches the center-stage ledger
- invalid transactions become obvious teachable moments
- the room feels like many independent nodes verifying the same shared history

Room Structure

Area | Role | What happens

Center stage | Lead moderator + projected ledger | Announces the transaction scenario, demonstrates card movement, shows expected balances, calls mining rounds.

8 participant tables | Mini Bitcoin networks | Recreate the same transaction flow with local roles and materials.

Roaming facilitators | Support/checking | Help tables stay synchronized, answer questions, and spot mistakes.

Roles per Table: 10 People

Use fixed roles so the table behaves like a tiny Bitcoin network.

Count | Role | What they do

2-3 | Wallets | Hold UTXO cards, create transactions, sign with private-key card/stamp. Suggested: Wallet A, B, C.

2-3 | Nodes | Verify signatures, UTXO ownership, balances, double-spends, and block validity.

2-3 | Miners | Roll dice, choose transactions from the mempool, build candidate blocks, earn reward + fees.

1 | Mempool Manager | Holds valid unconfirmed transactions and sorts by fee.

1 | Table Captain / Auditor | Keeps the table ledger aligned with the projected stage ledger.

Recommended for 10 participants: 3 wallets, 3 nodes, 2 miners, 1 mempool manager, 1 table captain.

If a table has 8 participants: use 2 wallets, 2 nodes, 2 miners, 1 mempool, 1 captain.

If a table has 12 participants: add Wallet D and Miner 3.

Center-Stage Mirroring Method

The moderator leads the room through a transaction script. Each table performs the same action locally.

The Basic Call-and-Response Pattern

- Moderator announces the ledger state.

Example: "Wallet A has 5 BTC. Wallet B has 3 BTC. Wallet C has 2 BTC."

- Moderator announces a transaction attempt.

Example: "Wallet A wants to send 2 BTC to Wallet B with a 1 BTC miner fee."

- Wallet A at each table builds the transaction.

Wallet A takes one or more UTXO cards, fills a transaction card, adds the recipient, amount, fee, and change output if needed, then signs it with the private-key stamp/card.

- Transaction goes to the first node.

Node checks: Is it complete? Is it signed by A? Does A control the UTXO? Is the amount + fee valid?

- Transaction passes through other nodes.

Other nodes independently verify. If valid, they pass it to the mempool. If invalid, they reject it and explain why.

- Mempool manager sorts it by fee.

Valid transaction card goes into the mempool basket/fee strip. Higher fees move to the front.

- Moderator calls mining.

Miners roll dice. The winning miner chooses the highest-fee valid transactions, adds a coinbase reward transaction, and creates a candidate block.

- Nodes validate the block.

Nodes verify the dice result, included transactions, signatures, UTXOs, fees, and reward.

- Table updates the blockchain and ledger.

If valid, block is attached to the table chain. Wallet balances/UTXO set update. Miner receives reward + fees.

- Moderator shows expected result.

Tables compare their ledger to the center-stage ledger. If they differ, they find the mistake.

Why UTXO Cards Matter

For this version, use physical UTXO cards instead of only balances. This shows that Bitcoin spends previous outputs, not an account balance like a bank.

A UTXO card should show:

- UTXO ID: A1, A2, B1, etc.
- Owner wallet: A / B / C
- Amount: ____ BTC
- Status: unspent / spent

When Wallet A spends a 5 BTC UTXO to pay B 2 BTC with a 1 BTC fee, the transaction consumes the 5 BTC UTXO and creates:

- 2 BTC output to B
- 2 BTC change output back to A
- 1 BTC fee to the miner once mined

This is the most important upgrade to the demonstration. It makes double-spend prevention visible.

Center-Stage Transaction Script

Run these scenarios in order. Every table mirrors the same action.

Starting UTXO Set

Wallet | Starting UTXOs | Total

Wallet A | A1 = 5 BTC | 5 BTC
Wallet B | B1 = 3 BTC | 3 BTC
Wallet C | C1 = 2 BTC | 2 BTC
Miner 1 | none | 0 BTC
Miner 2 | none | 0 BTC

Transaction 1: Valid Send

Moderator call: "Wallet A sends 2 BTC to Wallet B with a 1 BTC fee."

Table action:

- Wallet A selects UTXO A1 = 5 BTC
- Wallet A creates transaction:
 - input: A1 5 BTC
 - output: 2 BTC to B
 - change: 2 BTC back to A
 - fee: 1 BTC
- Wallet A signs
- Nodes validate
- Mempool accepts
- Miners roll
- Winning miner includes it in a block
- Nodes validate block
- Ledger updates

Expected after block:

Participant | Result

Wallet A | 2 BTC change UTXO
Wallet B | 3 BTC old + 2 BTC new = 5 BTC
Winning miner | 6 BTC reward + 1 BTC fee = 7 BTC

Transaction 2: Insufficient Funds Rejection

Moderator call: "Wallet C tries to send 10 BTC to Wallet A with a 1 BTC fee."

Table action:

- Wallet C only has C1 = 2 BTC
- Wallet C creates the transaction anyway
- Nodes check the UTXO set
- Nodes reject it before mempool

Expected result:

- Transaction never enters mempool
- No miner can include it
- Ledger does not change

Teaching line:

"A wallet can ask to do anything. Nodes enforce what is actually valid."

Transaction 3: Bad Signature Rejection

Moderator call: "Wallet B sends 1 BTC to Wallet C, but the card is signed with Wallet A's private key."

Expected result:

- Nodes reject: wrong signature
- Transaction does not enter mempool
- Ledger does not change

Teaching line:

"Owning bitcoin means being able to produce the right signature for the UTXO you are spending."

Transaction 4: Fee Market

Moderator call: "Two valid transactions enter the mempool. One pays a 0 BTC fee. One pays a 2 BTC fee. Block space only fits one."

Expected result:

- Mempool sorts higher fee first
- Miner selects the higher-fee transaction
- Low-fee transaction waits

Teaching line:

"The mempool is not first-in-first-out. Fees matter when block space is scarce."

Transaction 5: Double Spend Attempt

Moderator call: "Wallet A tries to spend the same UTXO twice: once to B and once to C."

Expected result:

- Nodes may see both attempts
- Only one can be valid in a block
- Once the UTXO is spent, the second transaction becomes invalid

Teaching line:

"Double spending is stopped because nodes track which outputs are already spent."

Transaction 6: Empty Block / Reward Only

Moderator call: "No transactions are waiting, but miners are still mining."

Expected result:

- Winning miner creates a block with only the coinbase reward
- Nodes accept it if reward is correct

Teaching line:

"Mining secures the chain even if the block has no user transactions."

Learning Outcomes

By the end, participants should be able to explain:

- Wallets create and sign transactions; they do not "send coins through the internet" like objects.
- Nodes enforce the rules before transactions and blocks are accepted.
- The mempool is a waiting room for valid, unconfirmed transactions.
- Fees influence which transactions miners include first.
- Mining is a probabilistic race to produce the next valid block.
- Difficulty adjusts so blocks do not come too quickly or too slowly.
- The blockchain is the shared history after nodes accept valid blocks.
- A miner can mine an empty block and still receive the block reward.
- Invalid transactions and invalid blocks are rejected even if a miner proposes them.

Table Kit Contents

Per Table Kit

Pack each table in a labelled bin or large envelope.

Item	Quantity per table	Purpose
------	--------------------	---------

Table sign	1	Table 1-8
Role cards	10	Wallets, nodes, miners, mempool, captain
Wallet balance cards	4	Starting balances for Wallets A-D
Wallet private-key envelopes/stamps	4	Fake signing authority
Transaction cards	40	Enough for 4-5 rounds
Invalid transaction cards	4	Seeded teachable moments
Node validation checklist	2	One per node
Node ledger sheets	2	Node copy of balances
Public table ledger	1	Table captain source of truth
Mempool basket/tray	1	Physical waiting room
Fee sorting strip	1	High fee -> low fee line
Block envelopes/boxes	6	One per mined block
Blockchain string/tape strip	1	Connect block envelopes in order
Dice	6	3 dice per miner
Miner reward cards	8	Coinbase transaction cards
BTC tokens/sats tokens	80-120	Optional tactile accounting
Pens/markers	6	Transactions and ledgers
Round timer card	1	Pacing reminder
Debrief card	1	Final discussion prompts

Whole-Room Extras

Item | Quantity | Purpose

Lead facilitator script | 1 | Main stage pacing
Projected timer | 1 | Keeps 8 tables synchronized
Front-of-room demo lockbox | 1 | Memorable "private key" prop
Giant mempool basket | 1 | Optional stage visual
Giant block box | 1 | Optional stage visual
Spare transaction cards | 80 | Backup
Spare dice | 16 | Backup
Tape rolls | 8-12 | Blockchain strips and signs
Clipboards | 8-12 | Captains/nodes

Suggested Starting Balances

Use whole numbers so the room can do the math quickly.

Wallet | Starting balance | Personality prompt

Wallet A | 10 BTC | Wants to buy coffee from B
Wallet B | 10 BTC | Sells coffee, pays rent to C
Wallet C | 10 BTC | Runs the venue, pays staff D
Wallet D | 10 BTC | Staff member, sends money to A
Miner 1 | 0 BTC | Earns rewards and fees
Miner 2 | 0 BTC | Earns rewards and fees

Block reward: 6 BTC for demo math.

Block size limit: 4 transactions.

Fee options: 0, 1, 2, or 3 BTC. High fee confirms first; zero-fee may wait.

Note: real Bitcoin does not use whole BTC fees like this. This is intentionally simplified.

Transaction Card Fields

Every card should include:

- Transaction ID: T_____
- From wallet: A / B / C / D
- To wallet: A / B / C / D / Miner 1 / Miner 2
- Amount: _____ BTC
- Fee: 0 / 1 / 2 / 3 BTC
- Total cost to sender: amount + fee
- Signature mark: sticker/stamp/secret word
- Node status: valid / invalid
- Reason if invalid: bad signature / insufficient funds / double spend / malformed

Round Structure

Run 4 rounds for a 25-30 minute demo. Run 6 rounds for a 45 minute version.

Round 0: Setup and Explainer - 5 minutes

Lead facilitator says:

"Each table is a tiny Bitcoin network. Wallets create transactions. Nodes check the rules. Valid transactions wait in the mempool. Miners compete to build the next block. Nodes decide whether that block becomes part of the chain."

Table captains confirm:

- wallets have balances
- nodes have ledger sheets
- mempool has basket
- miners have dice
- block envelopes are ready

Step 1: Wallets Create Transactions - 2 minutes

Wallets fill transaction cards and sign them using their private-key envelope/stamp.

Rules:

- Sender must pay amount + fee.
- Wallets may create more than one transaction, but risky double-spends may be rejected.
- Encourage varied fees so the mempool has something to sort.

Step 2: Nodes Validate Transactions - 2 minutes

Nodes check:

- Is the transaction complete?
- Does the signature match the sender?
- Does sender have enough balance for amount + fee?
- Has this same balance already been committed to another pending transaction?

Valid transactions go to the mempool. Invalid transactions are returned face-up with a reason.

Step 3: Mempool Sorts by Fee - 1 minute

Mempool manager places valid transactions in order:

3 BTC fee -> 2 BTC fee -> 1 BTC fee -> 0 BTC fee

Then announces:

"Block space is limited. The next miner can include up to 4 transactions. Highest fees are first."

Step 4: Miners Mine - 1-3 minutes

Miners roll 3 dice repeatedly.

Recommended pacing:

- Round 1: Low difficulty, at least one 6.
- Round 2: Medium difficulty, at least two 6s.

- Round 3: High difficulty demo, three 6s. Time cap at 45 seconds; if no one wins, difficulty adjusts down.
- Round 4+: Choose difficulty based on room energy.

When a miner wins, they shout: "Block found!"

Winning miner builds a block:

- up to 4 highest-fee mempool transactions
- one coinbase/reward transaction: NEW BTC -> Miner, 6 BTC

If the mempool is empty, miner builds an empty block with only the reward.

Step 5: Nodes Validate the Block - 2 minutes

Nodes check:

- Did the miner meet the dice difficulty?
- Are all transactions from the mempool?
- Are signatures valid?
- Are balances still sufficient in block order?
- Is the block reward correct?
- Are fees paid to the miner correctly?

If valid: attach block to chain and update ledgers.

If invalid: discard block, return transactions to mempool if still valid, mine again.

Step 6: Update Balances - 2 minutes

For each transaction in the accepted block:

- subtract amount + fee from sender
- add amount to receiver
- add fee to winning miner
- add 6 BTC block reward to winning miner

Unconfirmed transactions remain in the mempool.

Difficulty Adjustment Script

After every 2 rounds, lead facilitator asks:

"Were blocks coming too fast or too slow?"

Then says:

- "Too fast? The network raises difficulty."
- "Too slow? The network lowers difficulty."
- "The goal is predictable block timing, not easy winning."

For the live demo:

Difficulty | Dice rule | Use case

Low | At least one 6 on 3 dice | First round, fast success

Medium | At least two 6s on 3 dice | Main live setting

High | Three 6s on 3 dice | Short demonstration only

High, cumulative | Collect three 6s across multiple rolls | If you want high difficulty but still finish

Seeded Teachable Moments

Give each table captain four optional "incident cards." Use 1-2 only.

- Insufficient Funds

Wallet A tries to send 9 BTC with a 3 BTC fee while only holding 10 BTC.

- Bad Signature

Wallet C transaction has Wallet B's signature mark.

- Low Fee Delay

Wallet D sends a valid 0-fee transaction. It stays in the mempool while higher-fee transactions confirm.

- Double Spend Attempt

Wallet B creates two transactions spending the same balance. Nodes must catch or block validation must reject the second.

- Empty Block

If a miner wins before transactions arrive, they may mine an empty block and still receive the reward.

30-Minute Run of Show

Time | Activity

0:00-0:05 | Explain roles and table setup

0:05-0:11 | Round 1: easy block, simple successful transactions

0:11-0:18 | Round 2: fee market and medium difficulty

0:18-0:23 | Round 3: invalid transaction/double-spend + block validation

0:23-0:27 | Difficulty adjustment demo / empty block if useful

0:27-0:30 | Debrief: wallet vs node vs miner vs blockchain

45-Minute Run of Show

Time | Activity

0:00-0:06 | Explain roles and self-custody prop

0:06-0:13 | Round 1: valid transactions, low difficulty

0:13-0:21 | Round 2: fee competition, medium difficulty

0:21-0:29 | Round 3: bad signature / insufficient funds

0:29-0:36 | Round 4: double-spend and block validation

0:36-0:41 | Difficulty adjustment and empty block demo

0:41-0:45 | Debrief and table report-outs

Debrief Questions

Ask the room:

- Who created transactions?
- Who decided whether a transaction was valid?
- Did miners get to change the rules?
- Why did high-fee transactions confirm first?
- What happened to transactions that did not fit in the block?
- Why might a miner mine an empty block?
- What did the blockchain represent at your table?
- What would happen if one node lied but the rest rejected the block?
- How is this different from a bank updating a private ledger?
- What was the private key in this demo, and why did it matter?

Print Pack Checklist

For 8 tables, print:

- 8 table signs
- 80 role cards
- 32 wallet balance cards
- 320 transaction cards minimum; 400 preferred
- 32 invalid/incident cards
- 16 node validation checklists
- 16 node ledger sheets
- 8 public table ledgers
- 8 mempool fee sorting strips
- 48 block labels/envelopes
- 16 miner reward sheets/cards
- 8 debrief cards

Packing List: 8 Tables of 10

item | per_table | total_for_8_tables | notes

Table sign | 1 | 8 | Large visible Table 1-8
Role cards | 10 | 80 | 4 wallets 2 nodes 2 miners 1 mempool 1 captain
Wallet balance cards | 4 | 32 | Wallets A-D
Private-key envelopes or stamps | 4 | 32 | Fake signing authority; do not use real keys
Transaction cards | 40 | 320 | 50 per table preferred if longer session
Incident cards | 4 | 32 | Insufficient funds bad signature low fee double spend
Node validation checklist | 2 | 16 | One per node
Node ledger sheet | 2 | 16 | Node copies of wallet/miner balances
Public table ledger | 1 | 8 | Table captain source of truth
Mempool basket/tray | 1 | 8 | Physical waiting room
Fee sorting strip | 1 | 8 | High fee to low fee
Block envelopes or small boxes | 6 | 48 | One per accepted block
Blockchain string/tape strip | 1 | 8 | Attach blocks in order
Dice | 6 | 48 | 3 dice per miner
Miner reward cards | 8 | 64 | Coinbase/reward transaction cards
BTC/sats tokens | 100 | 800 | Optional tactile accounting
Pens/markers | 6 | 48 | Plus spares
Clipboards | 3 | 24 | Captain and nodes
Tape roll | 1 | 8 | Plus 4 room spares
Lead facilitator script | 0.125 | 1 | Whole room
Projected timer | 0.125 | 1 | Whole room
Demo lockbox | 0.125 | 1 | Whole room self-custody prop
Spare dice | 2 | 16 | Whole room backup
Spare transaction cards | 10 | 80 | Whole room backup