10 Blockchain Labs for Beginners (Using Python)

Each lab is **2 hours** long and designed to be engaging and beginner-friendly.

Lab 1: Introduction to Python for Blockchain

Objective: Teach students basic Python concepts needed for blockchain coding.
Variables, data types, loops, functions.
Install and use **Google Colab**.
Mini Task: Write a Python function to **simulate a simple ledger** (list of transactions).

Lab 2: Creating a Simple Blockchain

Objective: Build a basic blockchain using Python lists.
Define a **Block** structure (data, timestamp, hash, previous hash).
Create a **chain of blocks**.
Mini Task: Manually **add new blocks** and print the blockchain.

Lab 3: Adding Hashing to Blockchain

Objective: Introduce **SHA-256 hashing** to secure blockchain data.
Use Python's **hashlib** to hash block contents.
Ensure each block **links correctly** using previous hashes.
Mini Task: Try modifying a block and see how it breaks the chain.

Lab 4: Proof of Work (Mining a Block)

Objective: Implement a basic **proof-of-work mechanism**.
Introduce **Nonce** and make mining require computing a valid hash.
Simulate how Bitcoin miners solve puzzles.
Mini Task: Adjust difficulty levels and test how mining time changes.

Lab 5: Smart Contracts with Python (Intro to Web3.py)

Objective: Interact with Ethereum blockchain using Python.
Install and use **Web3.py**.
Connect to a **test Ethereum network**.
Mini Task: Write a Python script to **check an Ethereum account balance**.

Lab 6: Deploying a Simple Smart Contract

Objective: Write and deploy a smart contract using Python.
Introduce **Solidity basics** (very simple contract).

Deploy a **Hello World smart contract** using Web3.py.
Mini Task: Call a function in the contract from Python.

Lab 7: Simulating Cryptocurrency Transactions

Objective: Create a **simple transaction system** in Python.
Assign users **wallets with balances**.
Allow users to **send tokens** to each other.
Mini Task: Prevent transactions if balance is insufficient.

Lab 8: Building a Simple Blockchain-Based Voting System

Objective: Use blockchain for **secure voting**.
Store votes as transactions.
Prevent duplicate voting using simple validation.
Mini Task: Simulate an election and count votes securely.

Lab 9: NFT Basics with Python

Objective: Introduce **Non-Fungible Tokens (NFTs)**.
Explain **how NFTs are different from cryptocurrencies**.
Use Python to generate **unique NFT metadata**.
Mini Task: Create a JSON-based **NFT collection**.

Lab 10: Secure Blockchain Messaging System

Objective: Show how blockchain can be used for **secure messaging**.
Use **hashing and digital signatures** for security.
Store messages on a **private blockchain ledger**.
Mini Task: Encrypt messages and check **tamper-proof security**.

Final Project (Students Choose a Topic!)

Objective: Apply everything learned in labs to build a **mini blockchain application**.
Ideas for projects:
Decentralized Expense Tracker – Record expenses on a personal blockchain.
Blockchain-Based Certificate System – Store digital certificates securely.
Smart Contract Lottery – Players enter, and a winner is chosen using blockchain.
NFT Collection Generator – Create a set of unique NFT images and metadata.

Summary

Google Colab for labs (easier for beginners).
VS Code for final project (real-world experience).
10 hands-on labs covering **blockchain basics, smart contracts, transactions, security, and NFTs**.
A final **mini-project to reinforce learning** and allow creativity.