

Practical-2

Aim: Write a program to create a block for blockchain, and add transactions to blocks in the blockchain.

Code:

```
import hashlib

import json

from datetime import datetime

class Blockchain:

    def __init__(self):

        self.chain = []

        self.transactions = [] # List to store pending transactions.

        self.create_block(proof=1, previous_hash='0')

    def add_transaction(self, sender, receiver, amount):

        self.transactions.append({'sender': sender, 'receiver': receiver, 'amount': amount})

        return self.get_previous_block()['index'] + 1

    def create_block(self, proof, previous_hash):

        block = {

            'index': len(self.chain) + 1,

            'timestamp': str(datetime.now()),

            'transactions': self.transactions, # Add transactions to the block.

            'proof': proof,

            'previous_hash': previous_hash,
```

```
}

self.transactions = [] # Clear pending transactions.

self.chain.append(block)

return block


def get_previous_block(self):

    return self.chain[-1]

# Instantiate blockchain

blockchain = Blockchain()

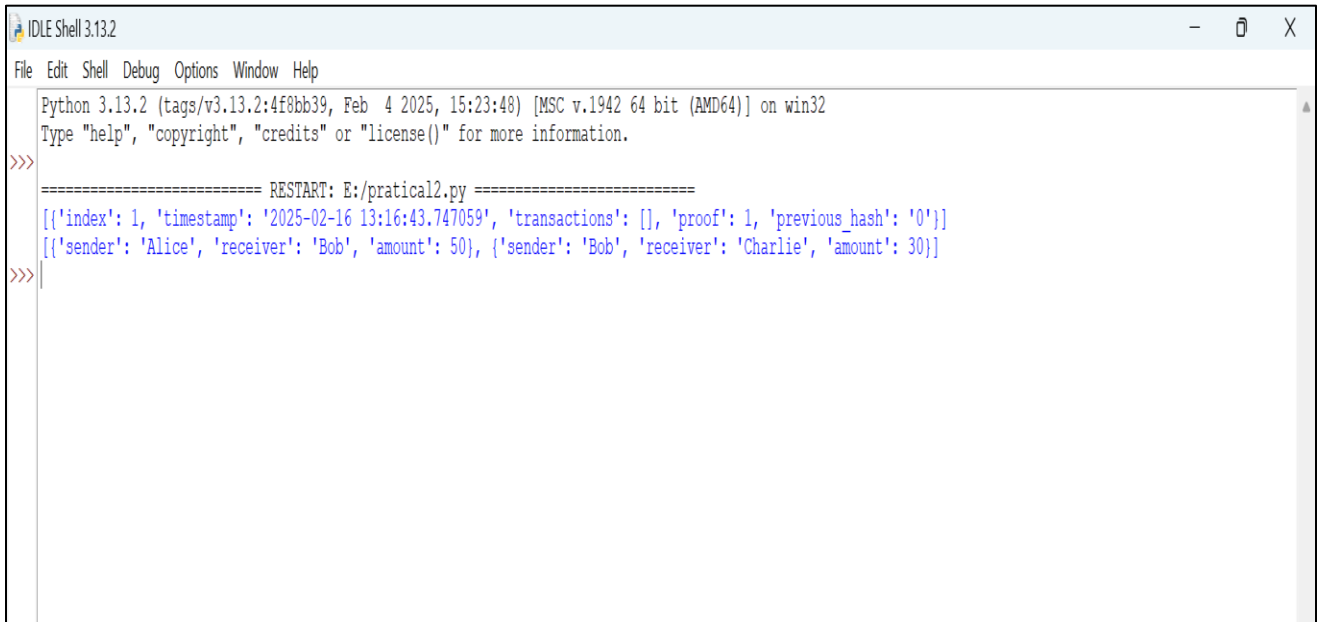
blockchain.add_transaction('Alice', 'Bob', 50)

blockchain.add_transaction('Bob', 'Charlie', 30)

print(blockchain.chain)

print(blockchain.transactions)
```

Output:



```
IDLE Shell 3.13.2
File Edit Shell Debug Options Window Help
Python 3.13.2 (tags/v3.13.2:4f80bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:/practical2.py =====
[{'index': 1, 'timestamp': '2025-02-16 13:16:43.747059', 'transactions': [], 'proof': 1, 'previous_hash': '0'}]
[{'sender': 'Alice', 'receiver': 'Bob', 'amount': 50}, {'sender': 'Bob', 'receiver': 'Charlie', 'amount': 30}]
>>>
```