

BLOCKCHAIN

ASSIGNMENT-1

NAME-SRAVANI

ROLL NO-2303A510G7

BATCH-30

Question-1:

Objective:

To learn blockchain interaction by creating a cryptocurrency wallet, checking wallet balance, and simulating transactions using Python and Web3

Requirements:

- Install Python 3.x
- Set up VS Code with Python extension
- Install required Python libraries:
- pip install web3
- Use a test blockchain network (Ethereum Sepolia / Ganache local blockchain)
- Basic understanding of blockchain wallets and private keys

Practical Description:

Step 1: Environment Setup

- Install Python and VS Code
- Install Web3.py library
- Create a Python file named wallet\_interaction.py

Step 2: Wallet and Blockchain Interaction Script

Create a Python script that:

- Connects to a blockchain network
- Loads a wallet using a private key
- Fetches wallet address

- Checks wallet balance
- Demonstrates transaction preparation (without real funds)

Code:

```
import tkinter as tk
from tkinter import messagebox
from web3 import Web3

# ----- Blockchain Setup -----

GANACHE_URL = "http://127.0.0.1:7545"

try:
    web3 = Web3(Web3.HTTPProvider(GANACHE_URL))
    connected = web3.is_connected()
except:
    connected = False

# Sample wallet (Ganache default account)
SAMPLE_ADDRESS = "0x0000000000000000000000000000000000000000"

# ----- Functions -----

def check_balance():
```

if not connected:

```
    messagebox.showinfo("Simulation Mode",  
        "Blockchain not connected.\nSimulated Balance: 10 ETH")
```

```
    return
```

try:

```
    balance_wei = web3.eth.get_balance(SAMPLE_ADDRESS)
```

```
    balance_eth = web3.from_wei(balance_wei, 'ether')
```

```
    messagebox.showinfo("Wallet Balance",
```

```
        f"Wallet Address:\n{SAMPLE_ADDRESS}\n\nBalance: {balance_eth} ETH")
```

except Exception as e:

```
    messagebox.showerror("Error", str(e))
```

def simulate\_transaction():

```
    tx_details = (
```

```
        "Transaction Simulation\n\n"
```

```
        "From: Your Wallet\n"
```

```
        "To: Receiver Wallet\n"
```

```
        "Amount: 1 ETH\n\n"
```

```
        "Note: This is only a simulation.\n"
```

```
        "No real transaction is performed."
```

```
)
```

```
messagebox.showinfo("Transaction", tx_details)
```

# ----- GUI Setup -----

```
root = tk.Tk()

root.title("Blockchain Wallet Simulator")

root.geometry("400x300")

root.resizable(False, False)
```

```
title_label = tk.Label(

    root,

    text="Blockchain Wallet (Python + Web3)",

    font=("Arial", 14, "bold")

)

title_label.pack(pady=15)
```

```
status_text = "Connected to Blockchain" if connected else "Simulation Mode (Offline)"

status_label = tk.Label(root, text=status_text, fg="green" if connected else "red")

status_label.pack(pady=5)
```

```
balance_btn = tk.Button(

    root,

    text="Check Wallet Balance",

    width=25,

    command=check_balance

)

balance_btn.pack(pady=10)
```

```
tx_btn = tk.Button(

    root,

    text="Simulate Transaction",

    width=25,
```

```

        command=simulate_transaction
    )
    tx_btn.pack(pady=10)

```

```

exit_btn = tk.Button(
    root,
    text="Exit",
    width=25,
    command=root.destroy
)
exit_btn.pack(pady=10)

```

```

root.mainloop()

```

OUTPUT:



