

ASSIGNMENT – 4

NAME: SRAVANI

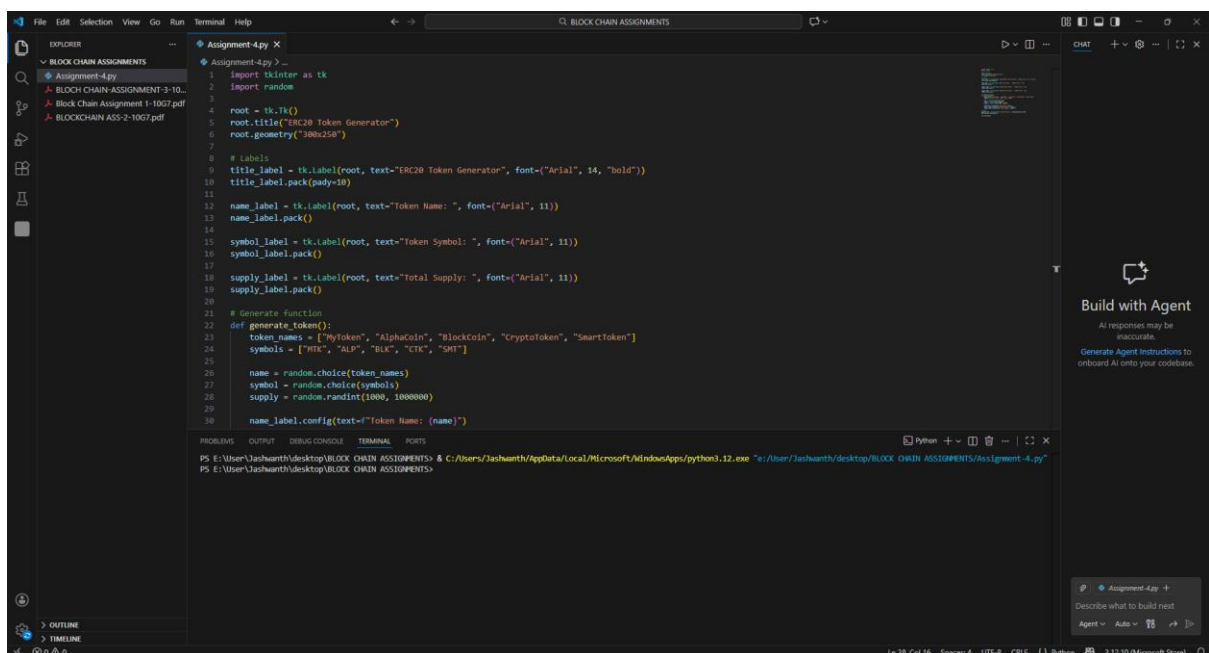
HT NO:2303A510G7

BATCH NO:30

Objective

To design, implement, and deploy a basic ERC-20 compliant token smart contract using Solidity on the Ethereum blockchain.

Output:



The screenshot shows a Visual Studio Code editor with a Python script named 'Assignment-4.py' open. The script is a Tkinter GUI for an ERC-20 Token Generator. It includes labels for the title, token name, token symbol, and total supply, along with a 'Generate' button. The terminal window at the bottom shows the command prompt and the execution of the script.

```
1 import tkinter as tk
2 import random
3
4 root = tk.Tk()
5 root.title("ERC20 Token Generator")
6 root.geometry("380x250")
7
8 # Labels
9 title_label = tk.Label(root, text="ERC20 Token Generator", font=("Arial", 14, "bold"))
10 title_label.pack(pady=10)
11
12 name_label = tk.Label(root, text="Token Name: ", font=("Arial", 11))
13 name_label.pack()
14
15 symbol_label = tk.Label(root, text="Token Symbol: ", font=("Arial", 11))
16 symbol_label.pack()
17
18 supply_label = tk.Label(root, text="Total Supply: ", font=("Arial", 11))
19 supply_label.pack()
20
21 # Generate function
22 def generate_token():
23     token_names = ["MyToken", "AlphaCoin", "BlockCoin", "CryptoToken", "SmartToken"]
24     symbols = ["BTC", "ALP", "BLK", "CTK", "SMT"]
25
26     name = random.choice(token_names)
27     symbol = random.choice(symbols)
28     supply = random.randint(1000, 1000000)
29
30     name_label.config(text=f"Token Name: {name}")
```

Terminal Output:

```
PS E:\User\Jashwanth\desktop\BLOCK CHAIN ASSIGNMENTS> & C:\Users\Jashwanth\AppData\Local\Microsoft\WindowsApps\python3.12.exe "e:\User\Jashwanth\desktop\BLOCK CHAIN ASSIGNMENTS\Assignment-4.py"
PS E:\User\Jashwanth\desktop\BLOCK CHAIN ASSIGNMENTS>
```

