**Name**: Vemburaj Konar **Class**: D16AD **Roll**: 28

Blockchain Lab 6

**Aim:** To develop a web-based gateway that serves as an entry point for users to access and interact with Ethereum Mainnet.

# Theory:

Etherscan is a popular blockchain explorer speciﬁcally designed for the Ethereum blockchain. It serves as a search engine, analytics platform, and API provider for Ethereum transactions, addresses, tokens, decentralized applications (DApps), and smart contracts.

# Working of Etherscan:

1. Data Retrieval: Etherscan continuously synchronizes with the Ethereum blockchain, collecting and indexing data like transactions, contract interactions, and token transfers.
2. User Interface: Users can search for speciﬁc transactions, addresses, or smart contracts using Etherscan's user-friendly interface. They can view real-time transaction details, contract source code, and token balances.
3. Veriﬁcation: Etherscan provides a platform for developers to verify their smart contracts' source code, ensuring transparency and security. Veriﬁed contracts show that the deployed code matches the published source.
4. API Services: Etherscan oﬀers APIs that developers can use to access Ethereum data programmatically. These APIs allow applications to interact with the Ethereum blockchain without running a full node.

# Why Do We Need Etherscan:

1. Transparency: Etherscan enhances the transparency of the Ethereum network. Anyone can inspect transactions, ensuring accountability and trust in the ecosystem.
2. Veriﬁcation: Developers can validate and showcase their smart contract source code, assuring users that the code has not been tampered with and operates as intended.
3. Research and Analysis: Researchers and analysts use Etherscan's data to gain insights into market trends, user behavior, and the overall health of the Ethereum network.
4. Monitoring: Users and businesses can monitor their Ethereum addresses, transactions, and contracts, ensuring the security of their assets and verifying the success of transactions.

In summary, Etherscan is a crucial tool for Ethereum users, developers, and researchers, oﬀering a transparent view of the blockchain, allowing veriﬁcation of smart contracts, and providing essential data and insights to the Ethereum community.

# Program:

**import requests import datetime**

**now = datetime.datetime.now() def get\_latest\_block(api\_key):**

**url = "https://api.etherscan.io/api" params = {**

**"module": "proxy",**

**"action": "eth\_getBlockByNumber", "tag": "latest",**

**"boolean": "true", "apikey": api\_key,**

**}**

**try:**

**response = requests.get(url, params=params) if response.status\_code == 200:**

**data = response.json() return data["result"]**

**else:**

**print("Request failed with status code:", response.status\_code)**

**except requests.RequestException as e: print("Request failed:", str(e))**

**return None**

**# Replace "YOUR\_API\_KEY" with your actual API key**

**api\_key = "361N7FBI6W9BSCSDSJWPIM6ZI9BB5N989V" # Paste your API**

**Key**

**print ("Current date and time : ", now.strftime("%d-%B-%Y")) latest\_block = get\_latest\_block(api\_key) #print(latest\_block)**

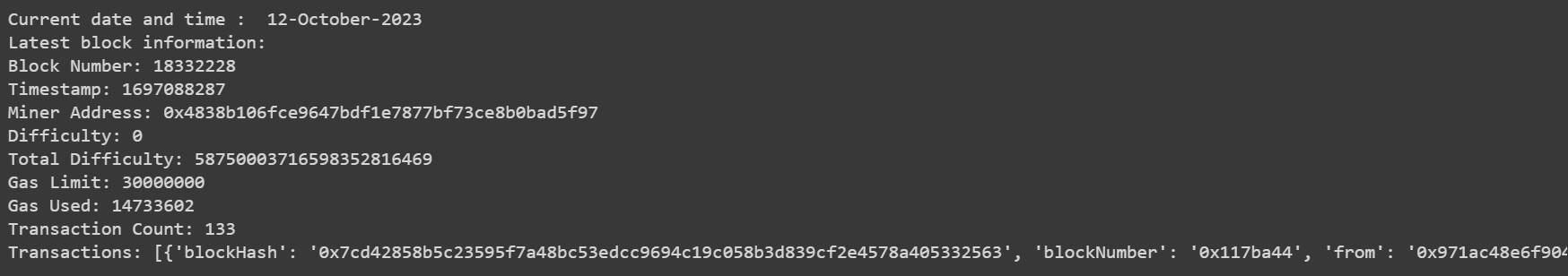
**if latest\_block is not None: print("Latest block information:")**

**print("Block Number:", int(latest\_block["number"], 16)) print("Timestamp:", int(latest\_block["timestamp"], 16)) print("Miner Address:", latest\_block["miner"]) print("Difficulty:", int(latest\_block["difficulty"], 16))**

**print("Total Difficulty:", int(latest\_block["totalDifficulty"], 16))**

**print("Gas Limit:", int(latest\_block["gasLimit"], 16)) print("Gas Used:", int(latest\_block["gasUsed"], 16)) print("Transaction Count:", len(latest\_block["transactions"]))**

**print("Transactions:", latest\_block["transactions"])**



**import requests**

**# Infura HTTP endpoint infura\_url =**

**'https://mainnet.infura.io/v3/7329beded7a74ad085a6144b63645314' # Make a request to retrieve the latest block number**

**response = requests.post( infura\_url,**

**json={**

**"jsonrpc": "2.0",**

**"method": "eth\_blockNumber", "params": [],**

**"id": 1**

**}**

**)**

**if response.status\_code == 200: result = response.json()**

**latest\_block\_number = int(result["result"], 16) # Convert hexadecimal to decimal**

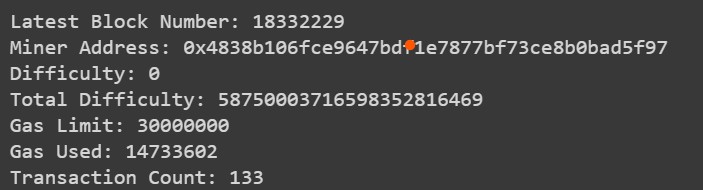
**print("Latest Block Number:",latest\_block\_number) # Print the desired information**

**print("Miner Address:", latest\_block["miner"]) # Error print("Difficulty:", int(latest\_block["difficulty"], 16)) print("Total Difficulty:", int(latest\_block["totalDifficulty"], 16)) print("Gas Limit:", int(latest\_block["gasLimit"], 16))**

**print("Gas Used:", int(latest\_block["gasUsed"], 16)) print("Transaction Count:", len(latest\_block["transactions"]))**

**else:**

**print("Failed to retrieve the latest block. Error:", response.text)**



**Conclusion:** Thus we have successfully implemented program to develop a web-based gateway that serves as an entry point for users to access and interact with Ethereum Mainnet.