

# BLOCKCHAIN TECHNOLOGY

## EXPERIMENT NO.06

Experiment 6

Shahrukh Shah  
60004220126  
Be comp.

Aim: Program to compute transaction details using  
etherbase API.

Theory: Ethereum is a widely used blockchain  
explore. Specifically designed for the ethereum network  
to provide a user friendly interface to access and  
analyze data from the ethereum blockchain, including  
transaction details, Smart contract & token information.

The Ethereum API allows for querying specific  
transaction details, virtual transaction hashes, wallet  
addresses, or block numbers. The API provides  
various endpoints to retrieve comprehensive information  
such as transaction history, status, fees, block  
and the ether token.

This section develops to create that can monitor,  
analyze and respond to blockchain events in  
real time. In the context of computing transaction  
details, the Ethereum API is particularly useful  
for retrieving information such as:

1) Transaction hash - A virtual identifier for each transaction  
on the ethereum network.

2) Sender & Recipient Address - The public key of the sender  
FOR EDUCATIONAL USE

28/11/2024 18:04

## CODE & OUTPUT:-

```
import requests
import json

class EtherscanAPI:
    def __init__(self, api_key):
        self.base_url = "https://api.etherscan.io/api"
        self.api_key = api_key

    def get_transaction_details(self, tx_hash):
        params = {
            'module': 'proxy',
            'action': 'eth_getTransactionByHash',
            'txhash': tx_hash,
            'apikey': self.api_key
        }

        response = requests.get(self.base_url, params=params)
        if response.status_code == 200:
            return response.json()['result']
        else:
            return None

    def get_transaction_receipt(self, tx_hash):
        params = {
            'module': 'proxy',
            'action': 'eth_getTransactionReceipt',
            'txhash': tx_hash,
            'apikey': self.api_key
        }

        response = requests.get(self.base_url, params=params)
        if response.status_code == 200:
            return response.json()['result']
        else:
            return None

def print_transaction_details(tx_details, tx_receipt):
    print("Transaction Details:")
    print(f"From: {tx_details['from']}")
    print(f"To: {tx_details['to']}")
    print(f"Value: {int(tx_details['value'], 16) / 1e18} ETH")
    print(f"Gas Price: {int(tx_details['gasPrice'], 16)} Wei")
    print(f"Gas Limit: {int(tx_details['gas'], 16)}")
    print(f"Nonce: {int(tx_details['nonce'], 16)}")
    print(f"Block Number: {int(tx_details['blockNumber'], 16)}")
    print(f"Transaction Index: {int(tx_details['transactionIndex'], 16)}")
```

```

    if tx_receipt:
        print("\nTransaction Receipt:")
        print(f"Status: {'Success' if tx_receipt['status'] == '0x1' else 'Failure'}")
        print(f"Gas Used: {int(tx_receipt['gasUsed'], 16)}")
        print(f"Cumulative Gas Used: {int(tx_receipt['cumulativeGasUsed'], 16)}")
        if 'contractAddress' in tx_receipt and tx_receipt['contractAddress']:
            print(f"Contract Address: {tx_receipt['contractAddress']}")

# Usage
if __name__ == "__main__":
    # Replace with your Etherscan API key
    api_key = "*****"
    etherscan = EtherscanAPI(api_key)

    # Replace with the transaction hash you want to look up
    tx_hash = "0x7e02ff81f98b1f2e4c7be063836bae54f8bb67716dcc01e54ec7b0c5ff487094"

    tx_details = etherscan.get_transaction_details(tx_hash)
    tx_receipt = etherscan.get_transaction_receipt(tx_hash)

    if tx_details and tx_receipt:
        print_transaction_details(tx_details, tx_receipt)
    else:
        print("Failed to retrieve transaction details.")

```

```
print(f"Contract Address: {tx_receipt['contractAddress']}")

# Usage
if __name__ == "__main__":
    # Replace with your Etherscan API key
    api_key = "3C8E8AMNEZR2F36C84NUSB7HIPJGDISGY4"
    etherscan = EtherscanAPI(api_key)

    # Replace with the transaction hash you want to look up
    tx_hash = "0x7e02ff81f98b1f2e4c7be063836bae54f8bb67716dcc01e54ec7b0c5ff487094"

    tx_details = etherscan.get_transaction_details(tx_hash)
    tx_receipt = etherscan.get_transaction_receipt(tx_hash)

    if tx_details and tx_receipt:
        print_transaction_details(tx_details, tx_receipt)
    else:
        print("Failed to retrieve transaction details.")
```

[8]

```
... Transaction Details:
From: 0xa119e544569cba0daefea99b44c4097f4a2825ab
To: 0x6571e50e8769d236414f3fb9e9b1d05341f6f79a
Value: 0.0 ETH
Gas Price: 1272308272 Wei
Gas Limit: 297221
Nonce: 123
Block Number: 20424012
Transaction Index: 136

Transaction Receipt:
Status: Success
Gas Used: 195429
Cumulative Gas Used: 12161380
```