Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 6**	
Title of Lab Assignment: Election Contract		
DOP: 03-09-2024	**DOS: 01-10-2024**	
CO Mapped:	**PO Mapped:**	**Signature:**

Practical No. 6

Aim: Develop an Election contract using solidity programming. Create a struct called Candidate, the struct members are ID, name and the vote-count. The smart contract should have the functions like addcandidate, show-candidates, vote, candidatescount and the voters function to verify the status of the casted vote using the Ethereum account address. Further, compile the contract and deploy to the personal Blockchain network using Ganache.

Theory:

Ganache is a personal Ethereum blockchain used for testing and development. It allows developers to deploy and test smart contracts in a local environment without needing to interact with the real Ethereum network. This tool simulates the blockchain by providing test accounts with Ether, faster block times, and a simplified transaction and mining process. Ganache is part of the Truffle Suite, making it easier for developers to create decentralized applications (DApps), test smart contracts, and explore blockchain features before deploying them on a live network. Its key benefit is providing a controlled, local environment where blockchain development can take place without financial risks or delays due to real-world transaction processing.

MetaMask is a cryptocurrency wallet that allows users to interact with the Ethereum blockchain through a browser extension or mobile app. It manages Ethereum-based assets and provides a user-friendly interface for sending and receiving Ether (ETH) and other Ethereum-based tokens. MetaMask also functions as a bridge between decentralized applications (DApps) and the Ethereum network, enabling users to sign transactions, approve smart contract interactions, and manage multiple Ethereum accounts. It is commonly used in blockchain development because it connects easily to both live Ethereum networks and local blockchains, such as those set up by Ganache. MetaMask plays a key role in Web3 development by enabling users to interact securely with decentralized platforms through their private wallets.

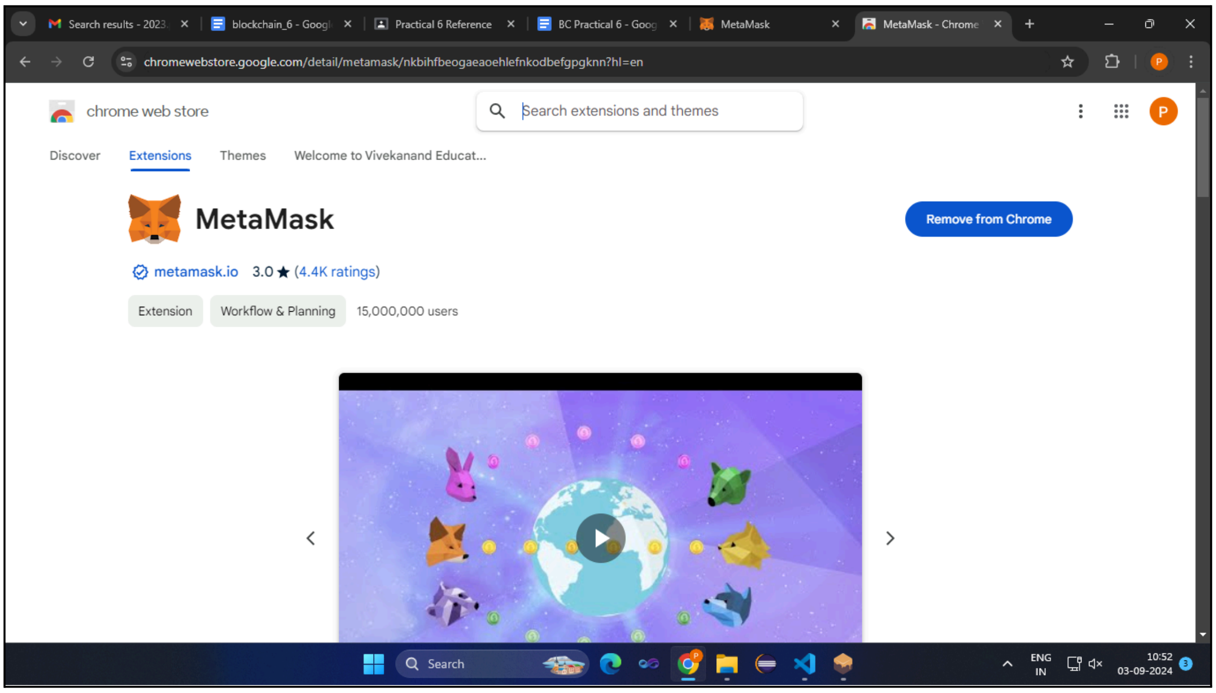
Smart contracts are self-executing programs on a blockchain that automatically enforce agreements or perform actions when predefined conditions are met. Written primarily in the Solidity programming language for Ethereum, smart contracts are used to handle transactions, manage digital assets, and perform automated functions without the need for intermediaries. These contracts are immutable once deployed, meaning that their code cannot be changed, ensuring the rules and logic will always operate as initially defined. For example, smart contracts can be used for a wide range of applications, such as voting systems, decentralized finance (DeFi), supply chain management, and more. They reduce the need for trust between parties by relying on the blockchain's transparent and secure infrastructure.

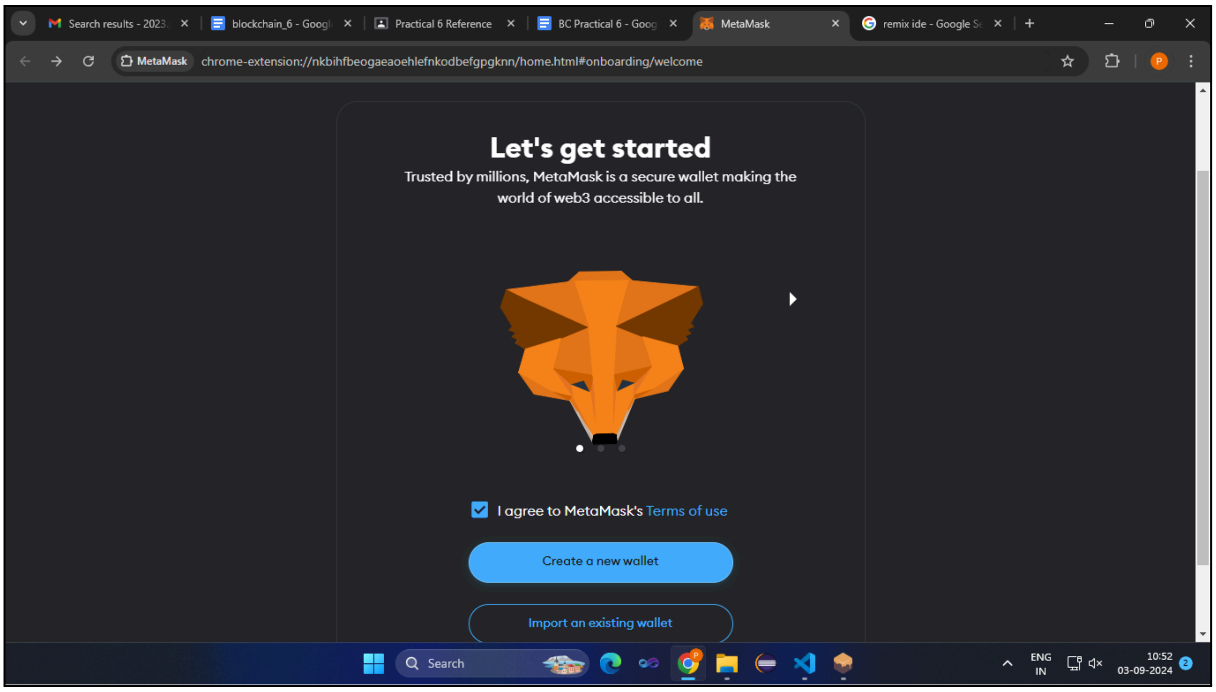
The **Web3 environment** refers to the decentralized internet where users interact directly with blockchain-based applications (DApps) and smart contracts. Unlike traditional web applications (Web2), which rely on centralized servers, Web3 enables peer-to-peer interactions on decentralized networks. MetaMask is a key component of this Web3 environment because it acts as a wallet and gateway to interact with these decentralized platforms. By injecting Ethereum accounts directly into the browser, MetaMask allows users to sign transactions, approve smart contracts, and manage their digital assets in a decentralized manner. It is widely used in the Ethereum ecosystem for connecting users to blockchain applications in a secure, non-custodial manner, meaning the user controls their private keys and assets.

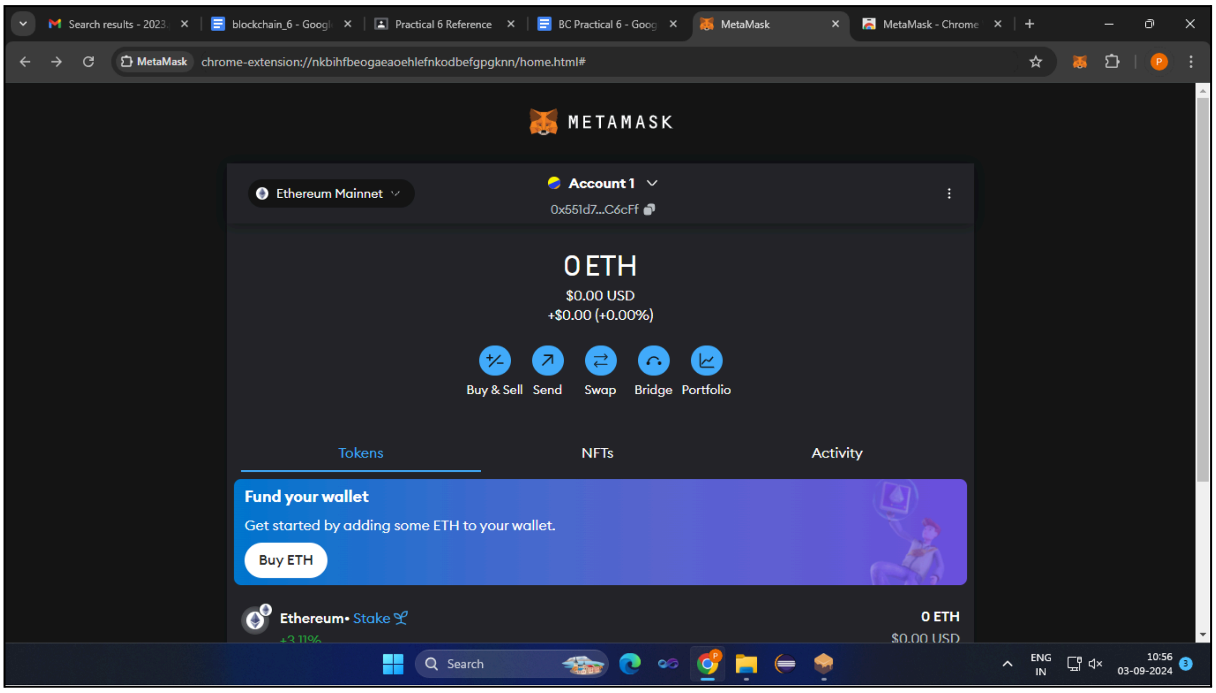
Remix IDE is a browser-based integrated development environment (IDE) for writing, compiling, deploying, and testing Ethereum smart contracts. It is one of the most popular tools for developing smart contracts using Solidity, Ethereum's programming language. Remix offers a user-friendly interface with a set of powerful features such as syntax highlighting, debugging, and contract analysis. One of its key advantages is that it does not require installation, making it accessible directly from the browser. Remix also integrates with tools like MetaMask and personal blockchains (like Ganache) for deploying and interacting with contracts in both local and real Ethereum environments. It is an essential tool for Ethereum developers, particularly for learning, prototyping, and testing smart contract functionality.

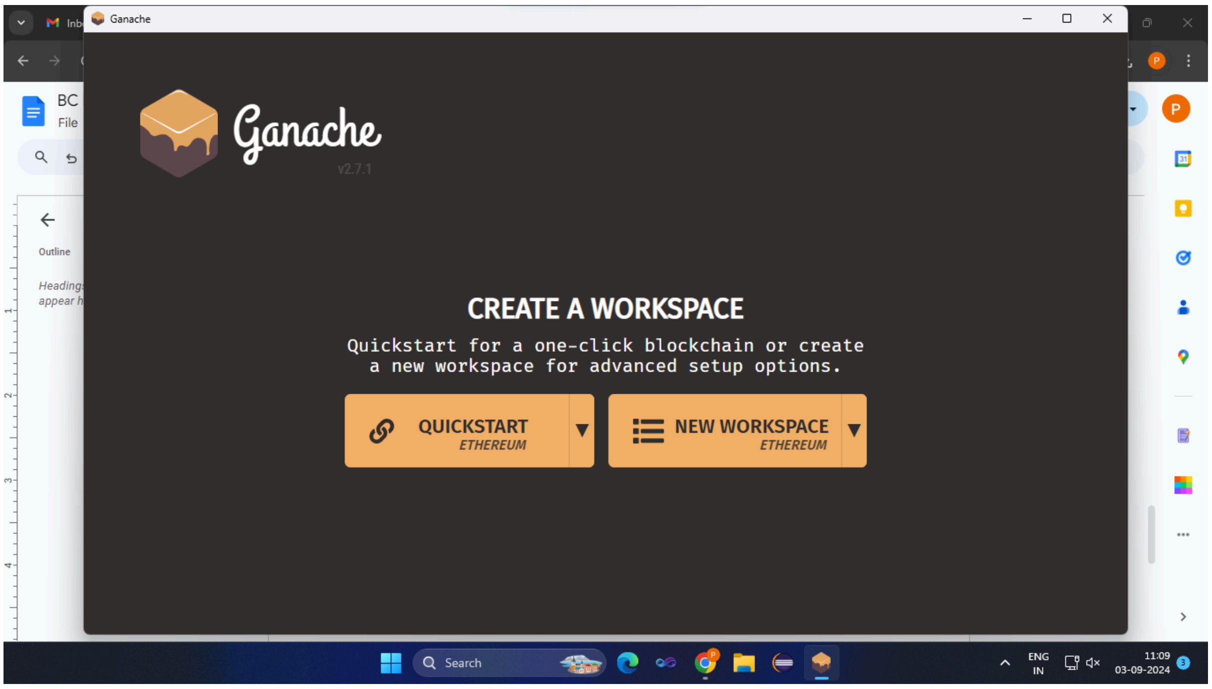
Code:
```
//SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Election {
    // Model a Candidate
    struct Candidate {
        //three propertiees for candidates
        uint id;
        string name;
        uint voteCount;
    }

    // Store accounts that have voted
    //voters list
     mapping(address => bool) public voters; //here address is account address, each user will
have unique and single address
    // Store Candidates, Fetch Candidate, candidate list
    mapping(uint => Candidate) public candidates;
    // Store Candidates Count
    uint public candidatesCount;

    // voted event, will fetch the candidate id whom you have voted
    event votedEvent (
        uint indexed _candidateId
    );
    constructor()  {
    addCandidate("N MODI, BJP");
    addCandidate("A kejriwal, AAP");
    addCandidate("Rahul G, Congress");
    addCandidate("Nikhil, JDS");
    }

    function addCandidate (string memory _name) private {
    candidatesCount ++;
    candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);
    }
    function vote (uint _candidateId) public {
    // require that they haven't voted before
    require(!voters[msg.sender]);
    // require a valid candidate
    require(_candidateId > 0 && _candidateId <= candidatesCount);
    // record that voter has voted
    voters[msg.sender] = true;
    // update candidate vote Count
    candidates[_candidateId].voteCount ++;
    // trigger voted event emit
    emit votedEvent(_candidateId);
    }
}
```

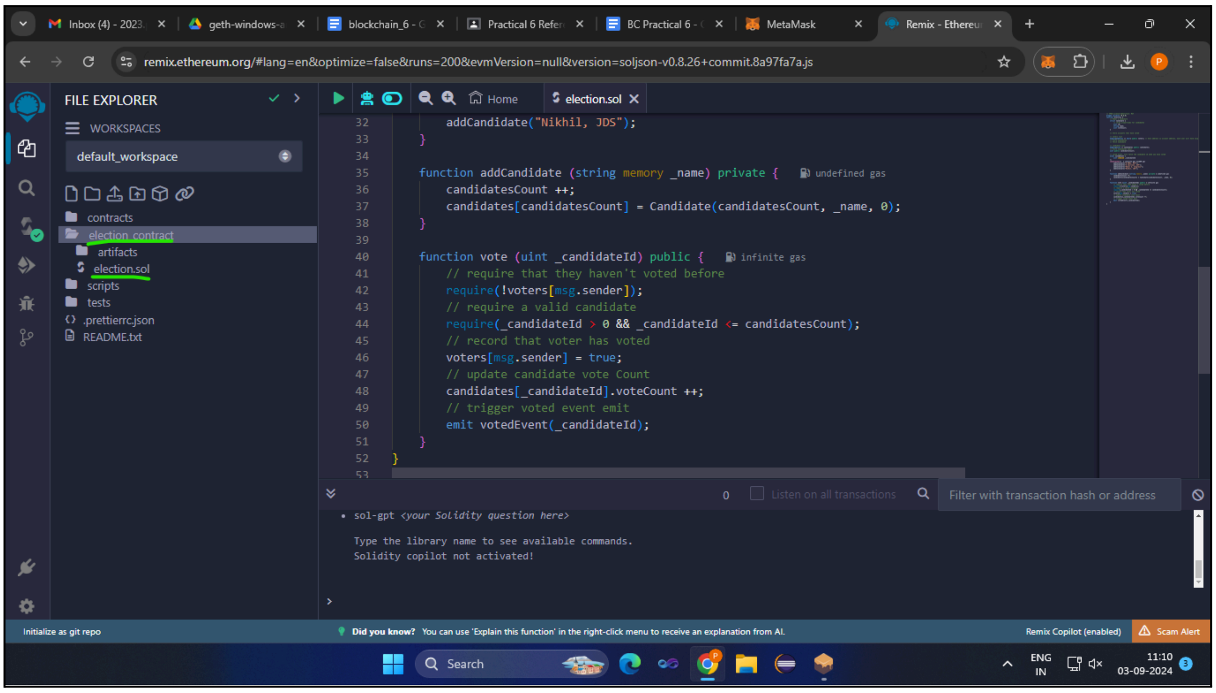
Output:


chromewebstore.google.com/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=en
chrome web store
Search extensions and themes
Discover
Extensions
Themes
Welcome to Vivekanand Educat...
MetaMask
Remove from Chrome
metamask.io 3.0 ★ (4.4K ratings)
Extension
Workflow & Planning
15,000,000 users
10:52
03-09-2024


chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#onboarding/welcome
Let's get started
Trusted by millions, MetaMask is a secure wallet making the world of web3 accessible to all.
I agree to MetaMask's Terms of use
Create a new wallet
Import an existing wallet
10:52
03-09-2024

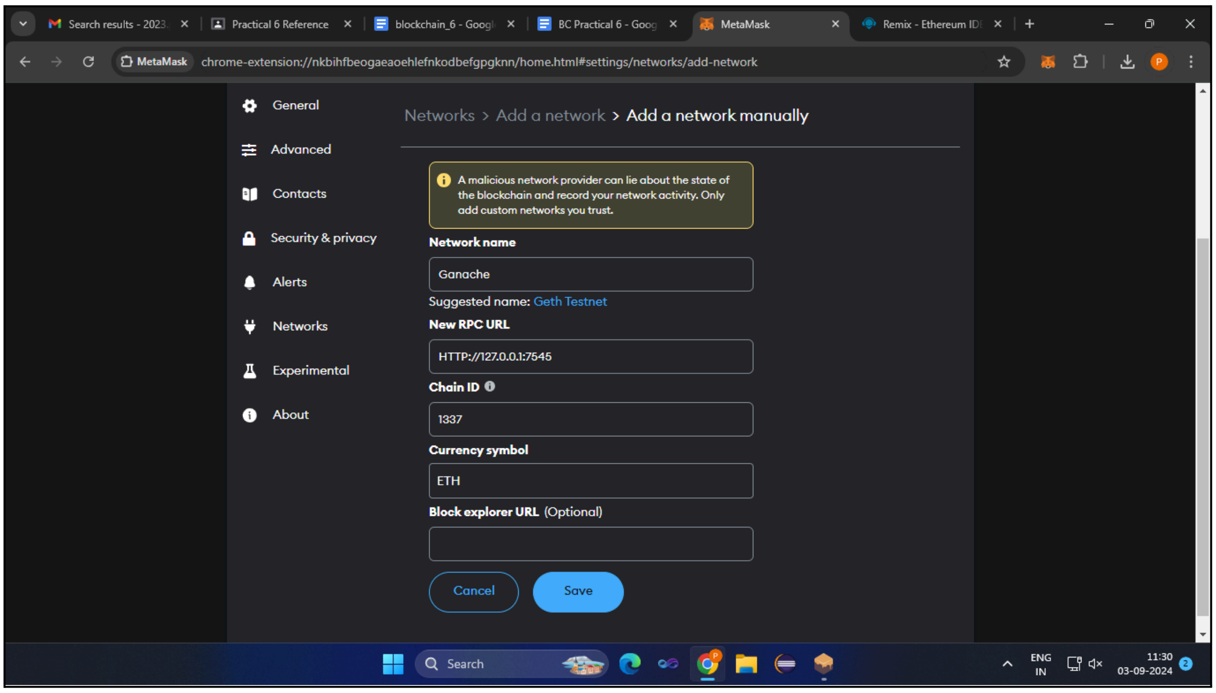

METAMASK
Ethereum Mainnet
Account 1
0x551d7...C6cFf
0 ETH
$0.00 USD
+$0.00 (+0.00%)
Buy & Sell
Send
Swap
Bridge
Portfolio
Tokens
NFTs
Activity
Fund your wallet
Get started by adding some ETH to your wallet.
Buy ETH
Ethereum• Stake
0 ETH
$0.00 USD


Ganache
v2.7.1
CREATE A WORKSPACE
Quickstart for a one-click blockchain or create a new workspace for advanced setup options.
QUICKSTART
ETHEREUM
NEW WORKSPACE
ETHEREUM



```
        addCandidate("Nikhil, JDS");
    }

    function addCandidate (string memory _name) private {   undefined gas
        candidatesCount ++;
        candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);
    }

    function vote (uint _candidateId) public {   infinite gas
        // require that they haven't voted before
        require(!voters[msg.sender]);
        // require a valid candidate
        require(_candidateId > 0 && _candidateId <= candidatesCount);
        // record that voter has voted
        voters[msg.sender] = true;
        // update candidate vote Count
        candidates[_candidateId].voteCount ++;
        // trigger voted event emit
        emit votedEvent(_candidateId);
    }
}
```

```
• sol-gpt <your Solidity question here>

  Type the library name to see available commands.
  Solidity copilot not activated!
```



Networks > Add a network > Add a network manually

A malicious network provider can lie about the state of the blockchain and record your network activity. Only add custom networks you trust.

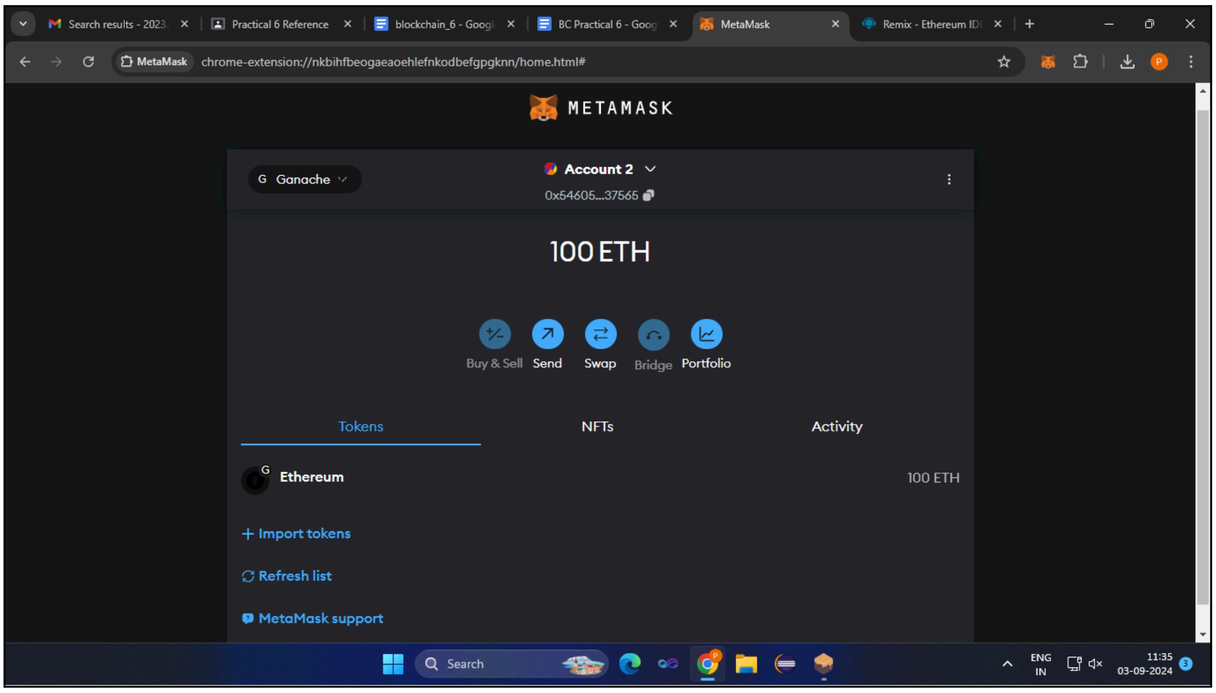
Network name
Ganache
Suggested name: Geth Testnet

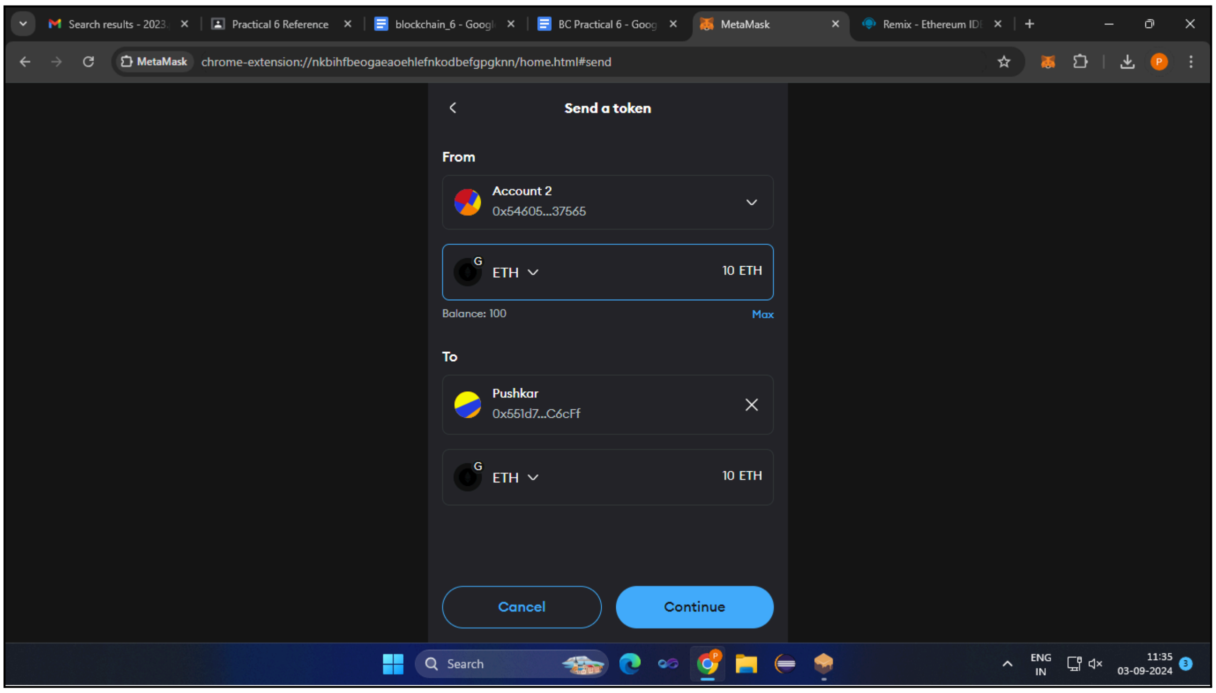
New RPC URL
HTTP://127.0.0.1:7545

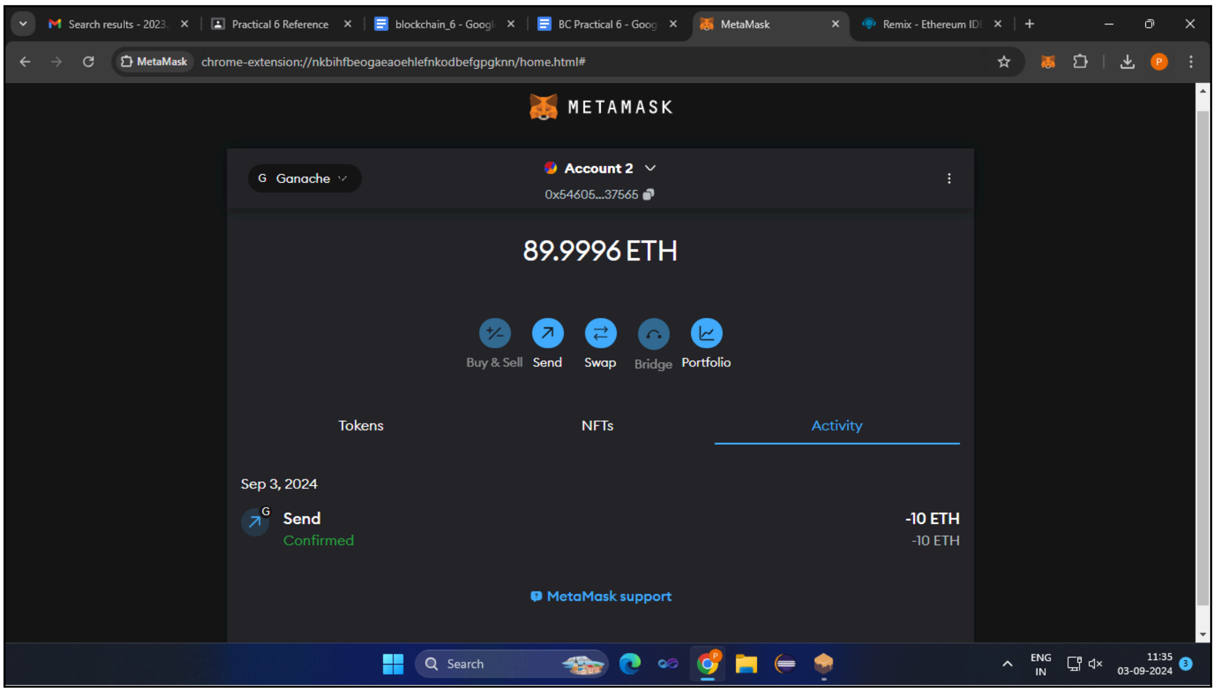
Chain ID
1337

Currency symbol
ETH

Block explorer URL (Optional)


chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#
METAMASK
Ganache
Account 2
0x54605...37565
100 ETH
Buy & Sell
Send
Swap
Bridge
Portfolio
Tokens
NFTs
Activity
Ethereum
100 ETH
+ Import tokens
Refresh list
MetaMask support

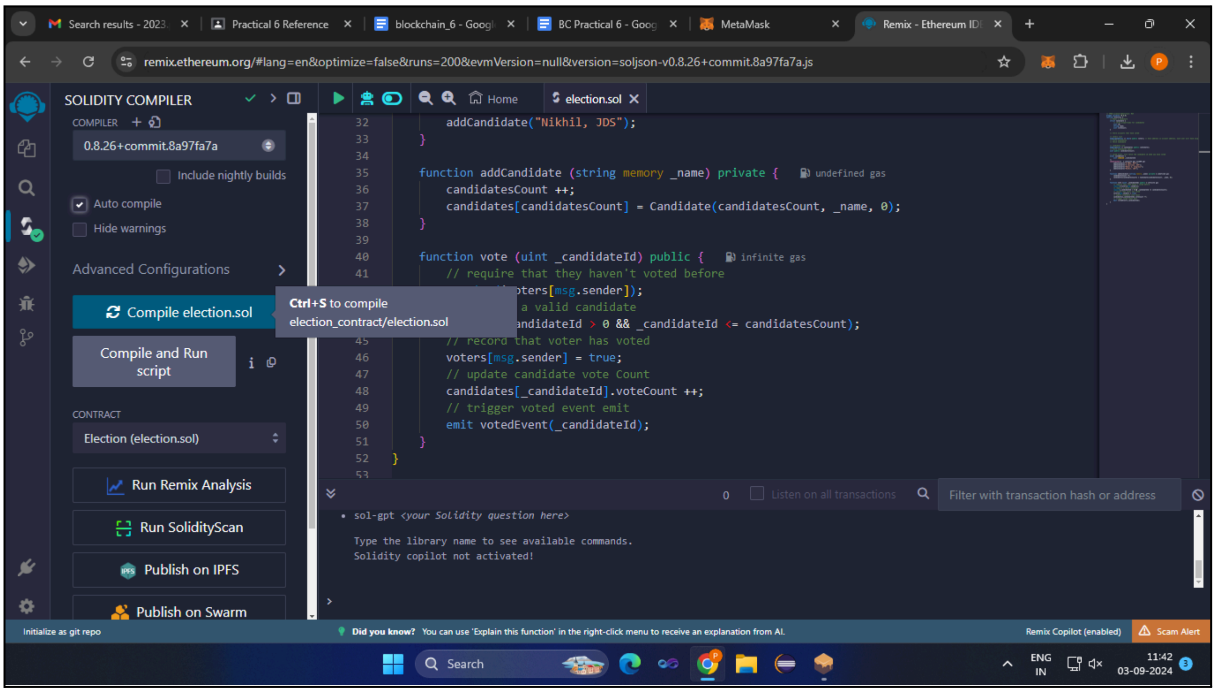

chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#send
Send a token
From
Account 2
0x54605...37565
ETH
10 ETH
Balance: 100
Max
To
Pushkar
0x551d7...C6cFf
ETH
10 ETH
Cancel
Continue



89.9996 ETH

Sep 3, 2024

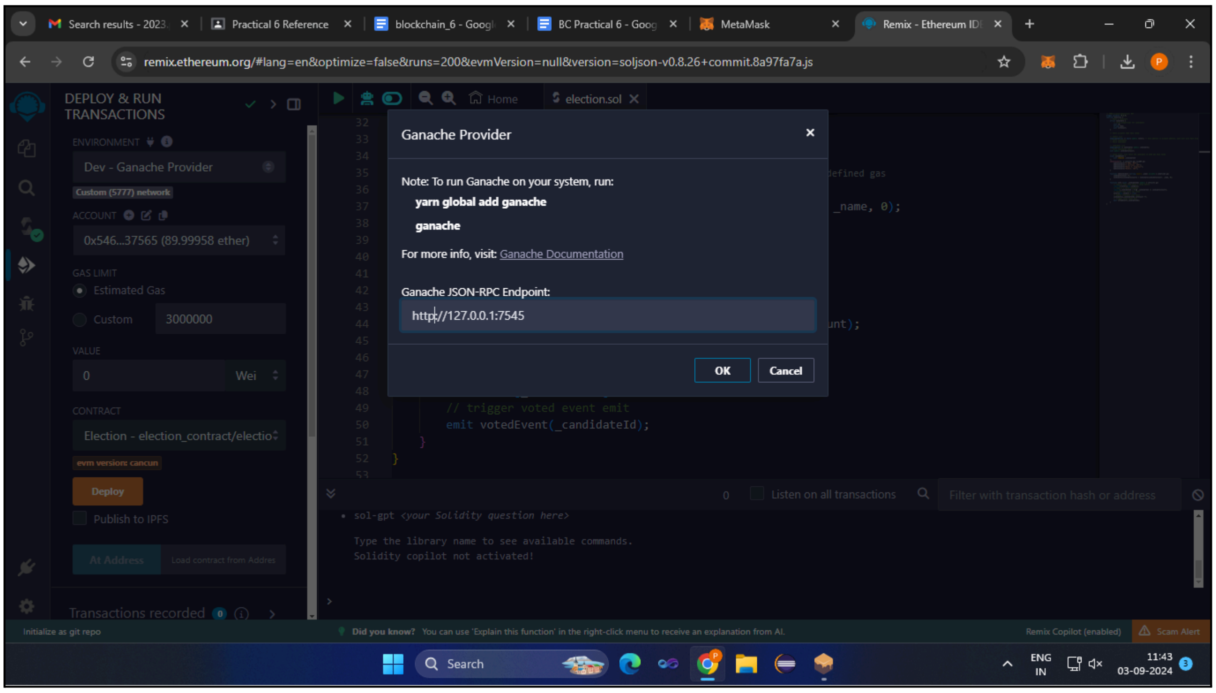
Send — Confirmed — -10 ETH — -10 ETH



```
        addCandidate("Nikhil, JDS");
    }

    function addCandidate (string memory _name) private {
        candidatesCount ++;
        candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);
    }

    function vote (uint _candidateId) public {
        // require that they haven't voted before
                 oters[msg.sender]);
              a valid candidate
                andidateId > 0 && _candidateId <= candidatesCount);
        // record that voter has voted
        voters[msg.sender] = true;
        // update candidate vote Count
        candidates[_candidateId].voteCount ++;
        // trigger voted event emit
        emit votedEvent(_candidateId);
    }
}
```

```
• sol-gpt <your Solidity question here>

  Type the library name to see available commands.
  Solidity copilot not activated!
```

Did you know? You can use 'Explain this function' in the right-click menu to receive an explanation from AI.



Ganache Provider

Note: To run Ganache on your system, run:

yarn global add ganache

ganache

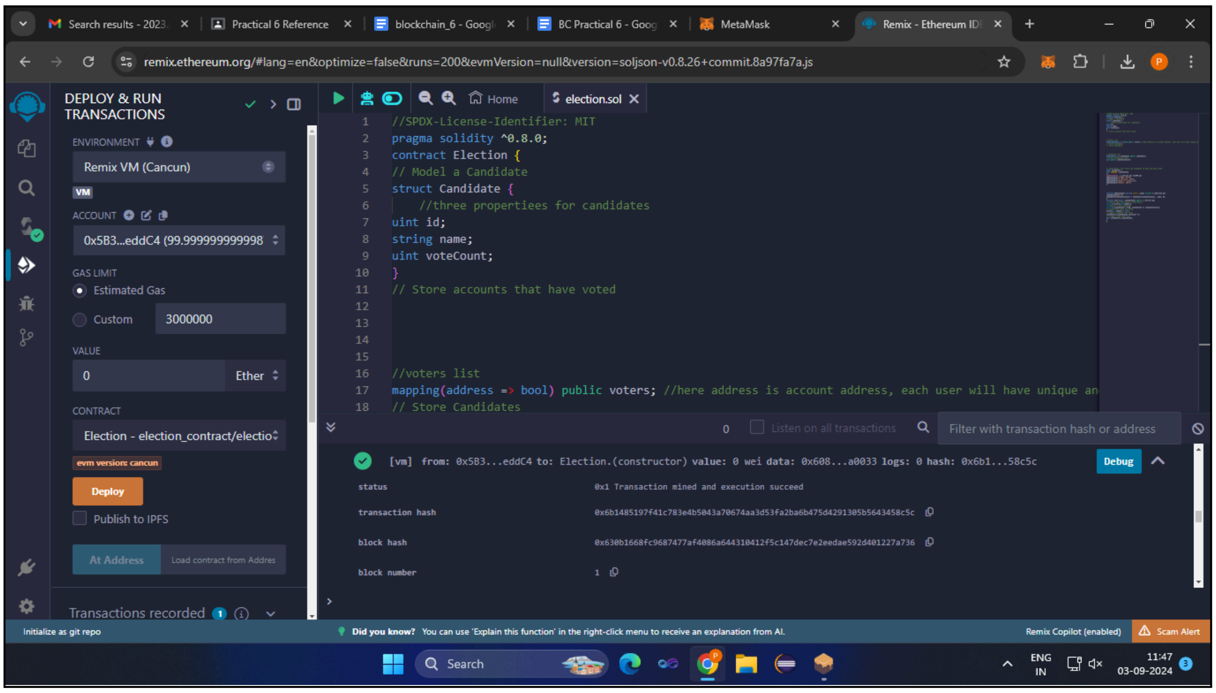
For more info, visit: Ganache Documentation

Ganache JSON-RPC Endpoint:

http://127.0.0.1:7545

OK Cancel

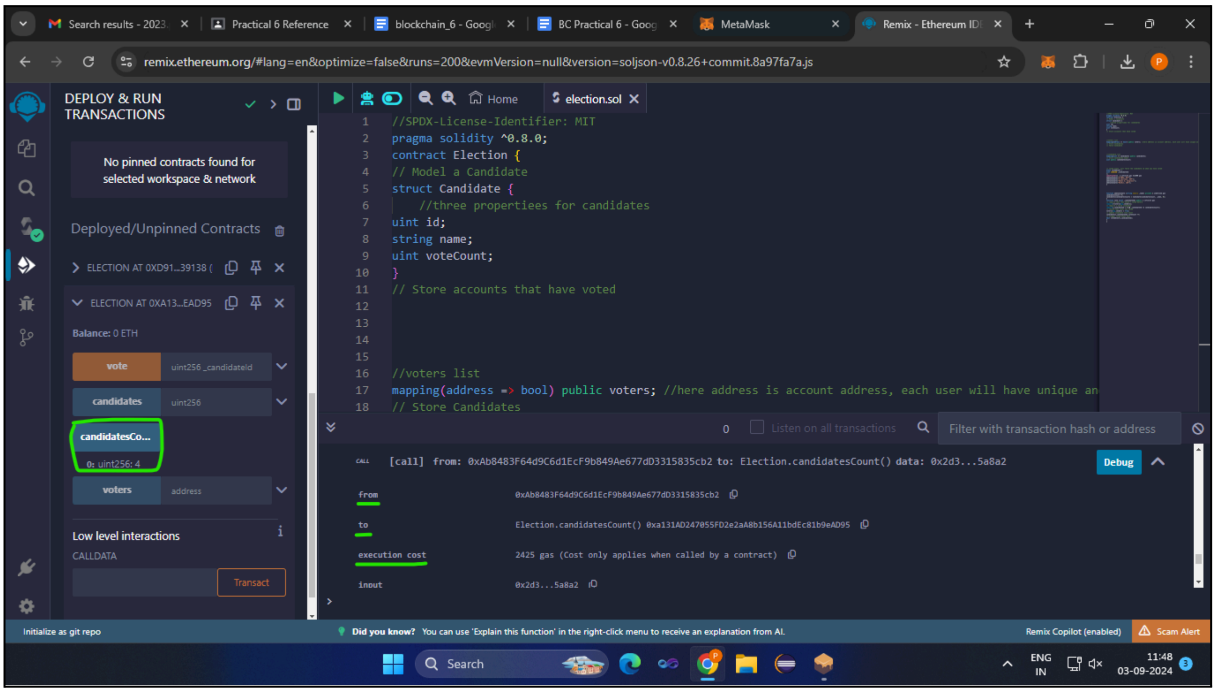
ENVIRONMENT: Dev - Ganache Provider (Custom (5777) network)
ACCOUNT: 0x546...37565 (89.99958 ether)
GAS LIMIT: Estimated Gas / Custom 3000000
VALUE: 0 Wei
CONTRACT: Election - election_contract/electio



ENVIRONMENT: Remix VM (Cancun)
ACCOUNT: 0x5B3...eddC4 (99.999999999998
GAS LIMIT: Estimated Gas / Custom 3000000
VALUE: 0 Ether
CONTRACT: Election - election_contract/electio

```
//SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Election {
// Model a Candidate
struct Candidate {
    //three propertiees for candidates
uint id;
string name;
uint voteCount;
}
// Store accounts that have voted




//voters list
mapping(address => bool) public voters; //here address is account address, each user will have unique an
// Store Candidates
```

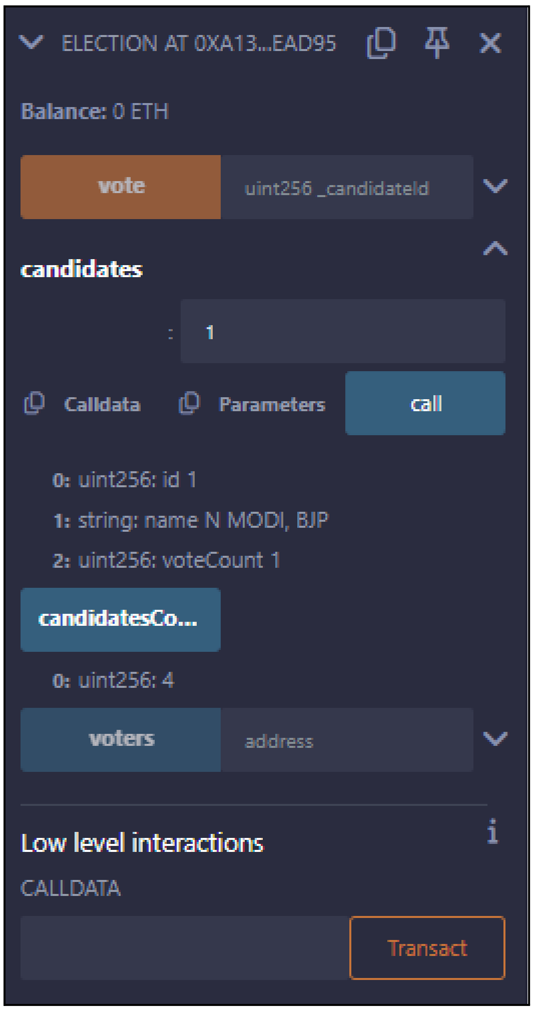
```
[vm] from: 0x5B3...eddC4 to: Election.(constructor) value: 0 wei data: 0x608...a0033 logs: 0 hash: 0x6b1...58c5c
status               0x1 Transaction mined and execution succeed
transaction hash     0x6b1485197f41c783e4b5043a70674aa3d53fa2ba6b475d4291305b5643458c5c
block hash           0x630b1668fc9687477af4086a644310412f5c147dec7e2eedae592d401227a736
block number         1
```



```
//SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Election {
// Model a Candidate
struct Candidate {
    //three propertiees for candidates
uint id;
string name;
uint voteCount;
}
// Store accounts that have voted




//voters list
mapping(address => bool) public voters; //here address is account address, each user will have unique an
// Store Candidates
```

```
[call] from: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 to: Election.candidatesCount() data: 0x2d3...5a8a2
from            0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2
to              Election.candidatesCount() 0xa131AD247055FD2e2aA8b156A11bdEc81b9eAD95
execution cost  2425 gas (Cost only applies when called by a contract)
input           0x2d3...5a8a2
```

DEPLOY & RUN TRANSACTIONS
No pinned contracts found for selected workspace & network
Deployed/Unpinned Contracts
ELECTION AT 0XD91...39138 (
ELECTION AT 0XA13...EAD95
Balance: 0 ETH
vote
uint256 _candidateId
candidates
uint256
candidatesCo...
0: uint256: 4
voters
address
Low level interactions
CALLDATA
Transact
Did you know? You can use 'Explain this function' in the right-click menu to receive an explanation from AI.
Remix Copilot (enabled)
Scam Alert



ELECTION AT 0XA13...EAD95

Balance: 0 ETH

vote uint256 _candidateId

candidates

: 1

Calldata Parameters call

0: uint256: id 1
1: string: name N MODI, BJP
2: uint256: voteCount 1

candidatesCo...

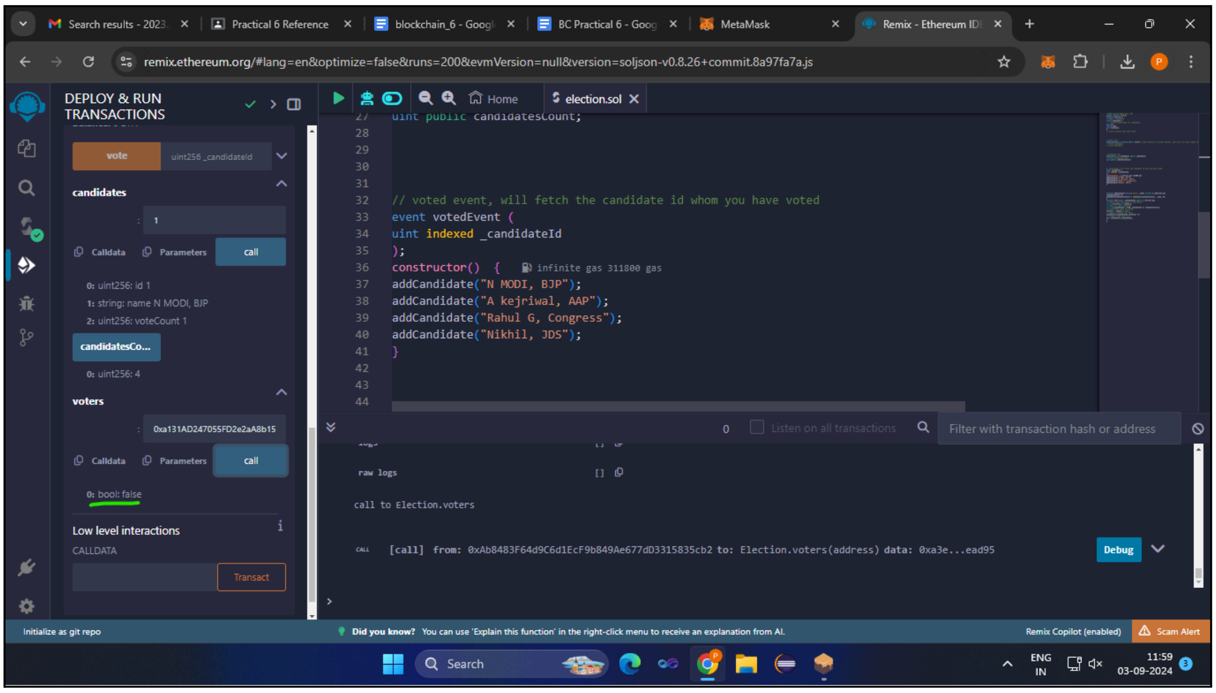
0: uint256: 4

voters address

Low level interactions

CALLDATA

Transact



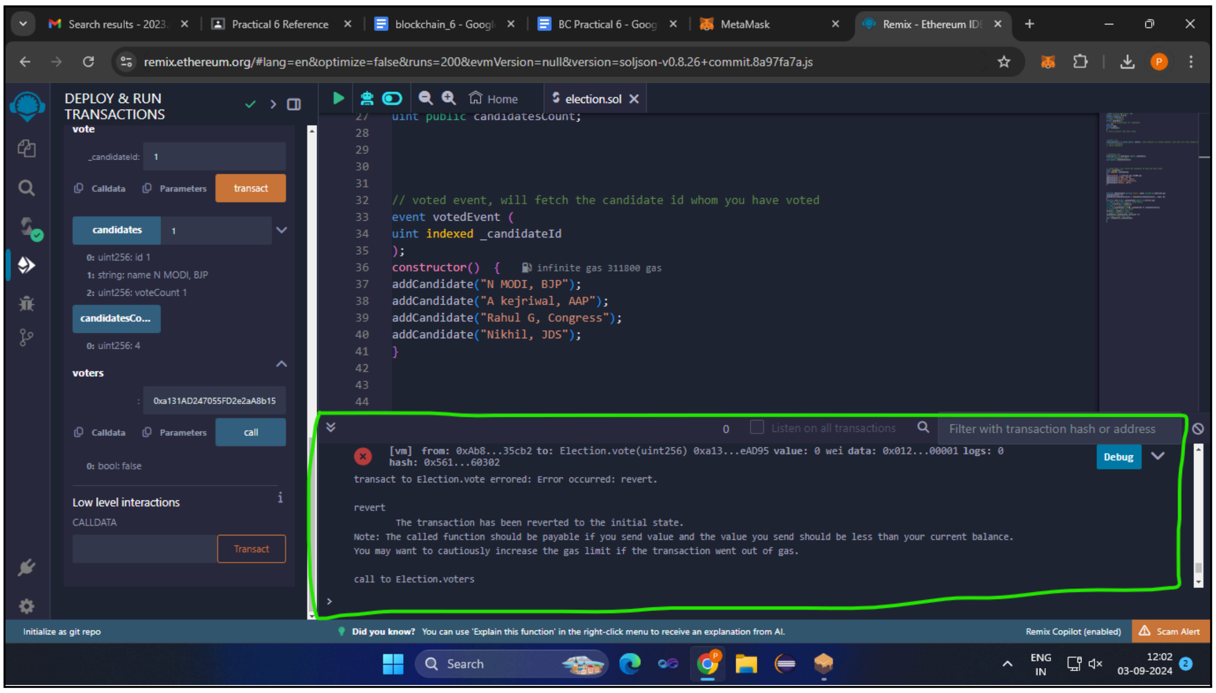
```
uint public candidatesCount;



// voted event, will fetch the candidate id whom you have voted
event votedEvent (
uint indexed _candidateId
);
constructor() {
addCandidate("N MODI, BJP");
addCandidate("A kejriwal, AAP");
addCandidate("Rahul G, Congress");
addCandidate("Nikhil, JDS");
}
```

candidates: 1
0: uint256: id 1
1: string: name N MODI, BJP
2: uint256: voteCount 1

candidatesCount
0: uint256: 4

voters: 0xa131AD247055FD2e2aA8b15
0: bool: false

```
raw logs                 []

call to Election.voters

CALL  [call] from: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 to: Election.voters(address) data: 0xa3e...ead95
```



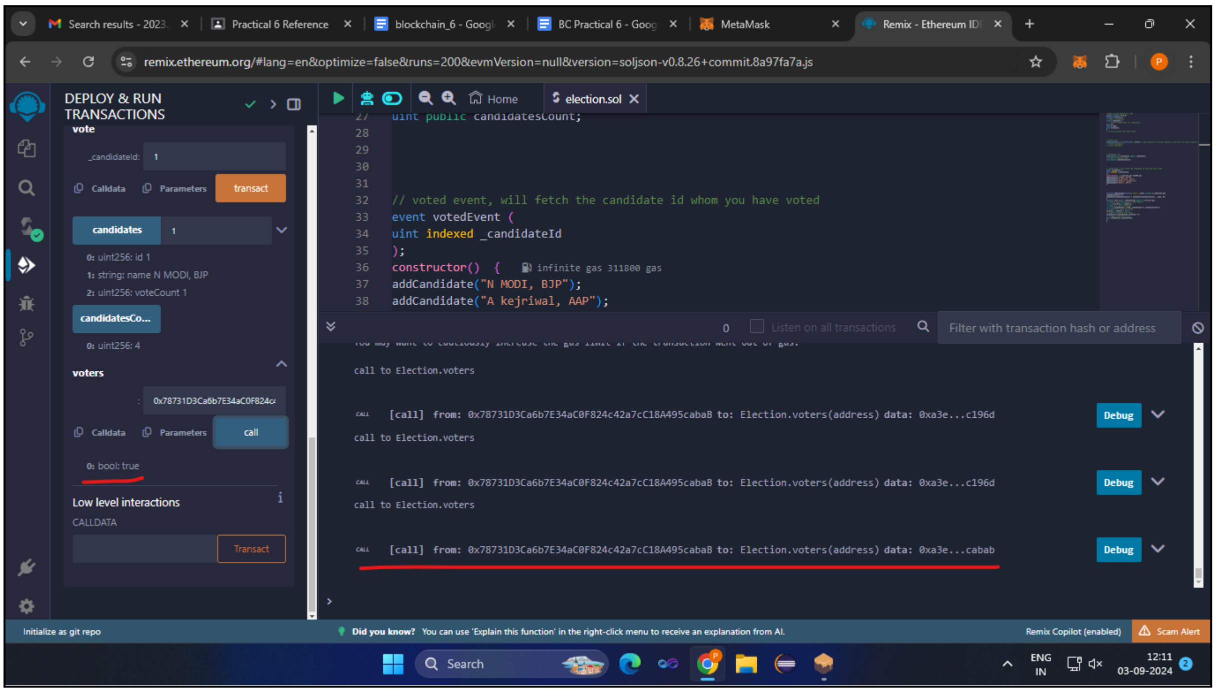
vote
_candidateId: 1

candidates: 1
0: uint256: id 1
1: string: name N MODI, BJP
2: uint256: voteCount 1

candidatesCount
0: uint256: 4

voters: 0xa131AD247055FD2e2aA8b15
0: bool: false

```
[vm] from: 0xAb8...35cb2 to: Election.vote(uint256) 0xa13...eAD95 value: 0 wei data: 0x012...00001 logs: 0
hash: 0x561...60302
transact to Election.vote errored: Error occurred: revert.

revert
        The transaction has been reverted to the initial state.
Note: The called function should be payable if you send value and the value you send should be less than your current balance.
You may want to cautiously increase the gas limit if the transaction went out of gas.

call to Election.voters
```



DEPLOY & RUN TRANSACTIONS

vote
_candidateId: 1
transact

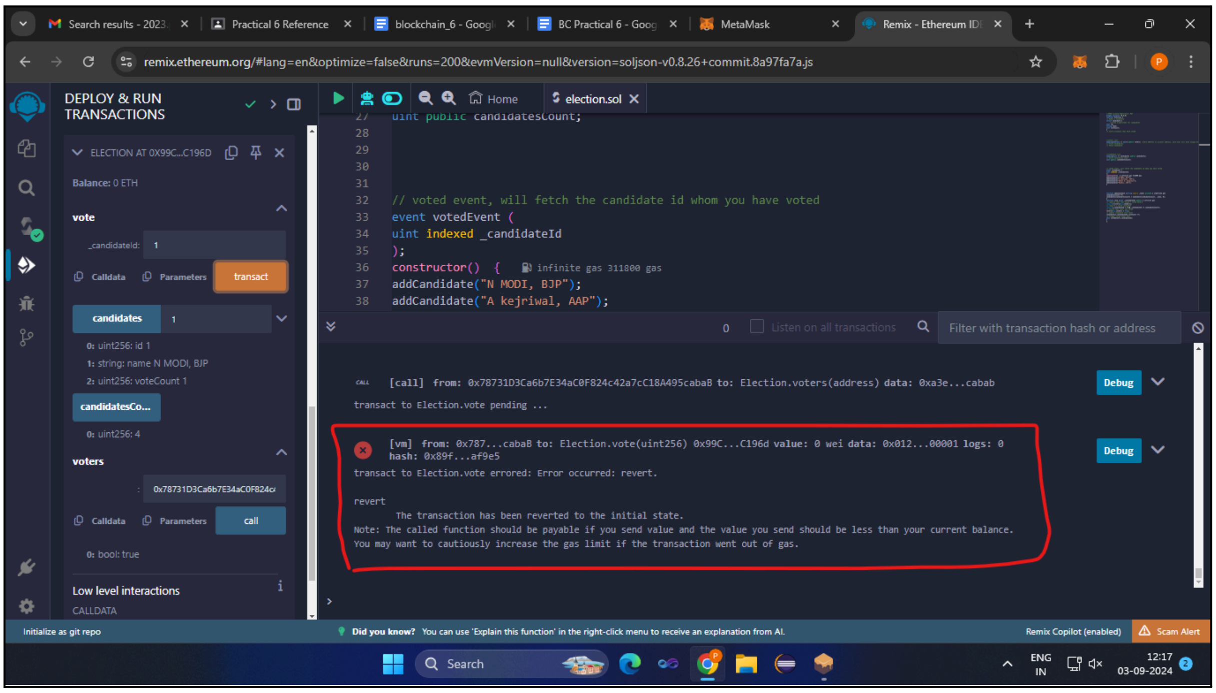
candidates 1
0: uint256: id 1
1: string: name N MODI, BJP
2: uint256: voteCount 1

candidatesCount
0: uint256: 4

voters
0x78731D3Ca6b7E34aC0F824c
call
0: bool: true

```
27  uint public candidatesCount;
28
29
30
31
32  // voted event, will fetch the candidate id whom you have voted
33  event votedEvent (
34  uint indexed _candidateId
35  );
36  constructor()  {    infinite gas 311800 gas
37  addCandidate("N MODI, BJP");
38  addCandidate("A kejriwal, AAP");
```

```
call to Election.voters

[call] from: 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB to: Election.voters(address) data: 0xa3e...c196d
call to Election.voters

[call] from: 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB to: Election.voters(address) data: 0xa3e...c196d
call to Election.voters

[call] from: 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB to: Election.voters(address) data: 0xa3e...cabab
```



ELECTION AT 0X99C...C196D
Balance: 0 ETH

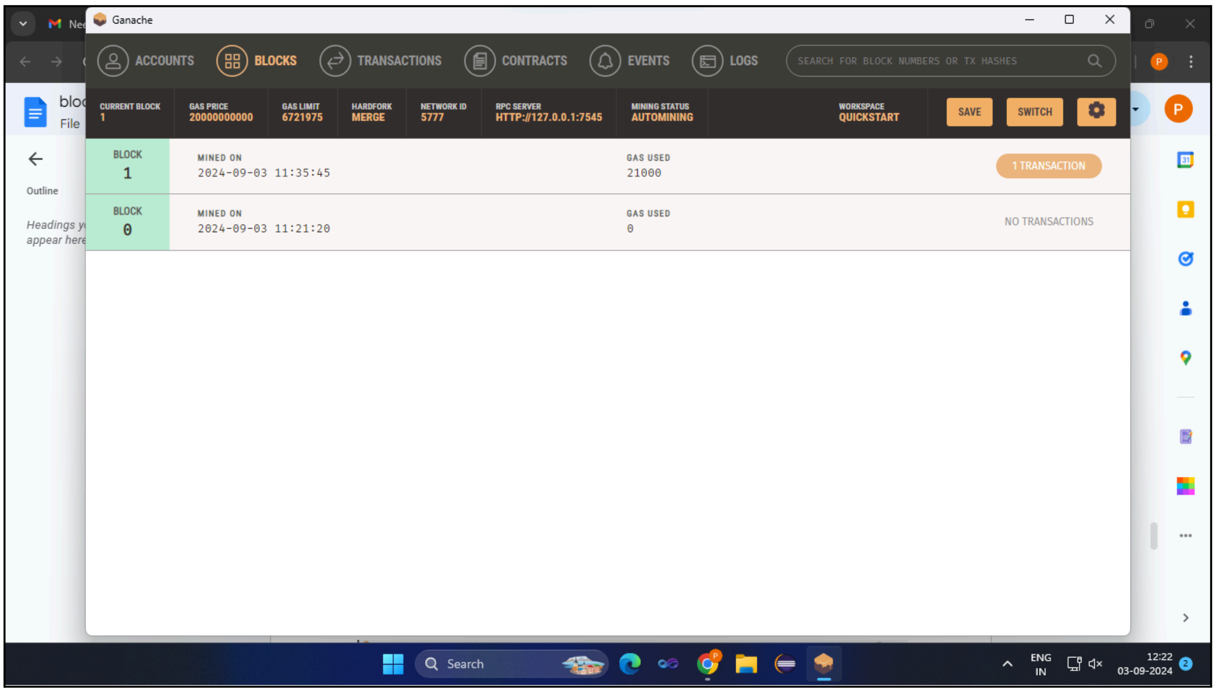
vote
_candidateId: 1
transact

candidates 1
0: uint256: id 1
1: string: name N MODI, BJP
2: uint256: voteCount 1

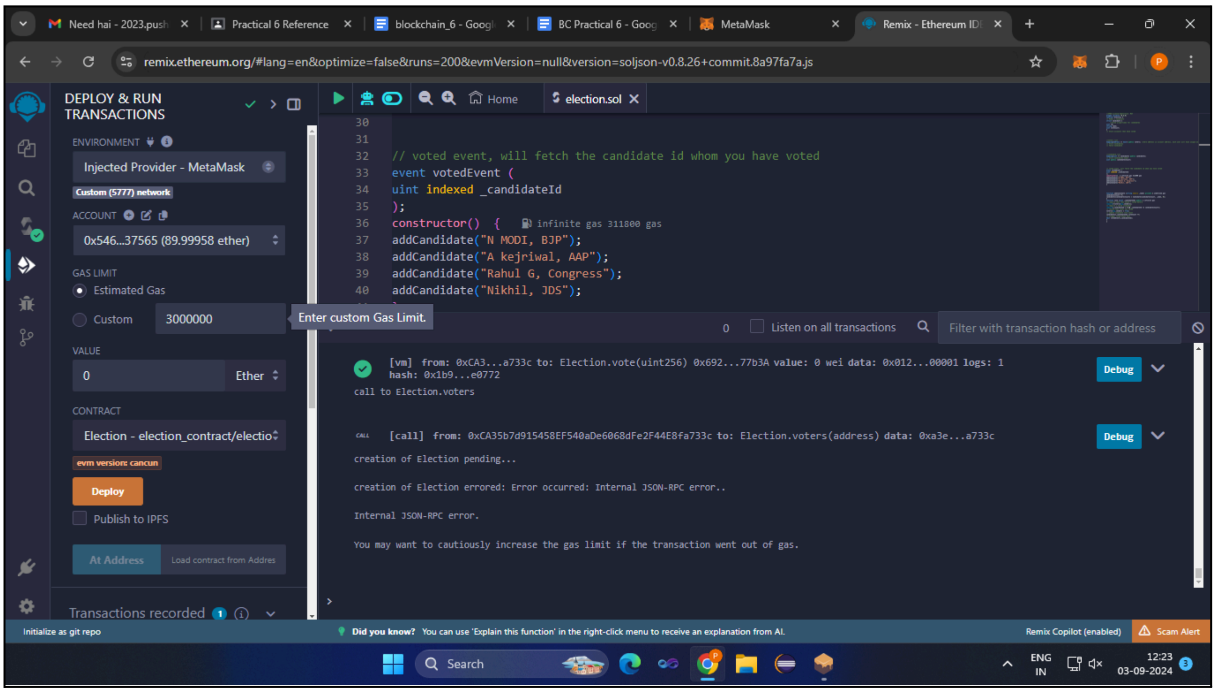
candidatesCount
0: uint256: 4

voters
0x78731D3Ca6b7E34aC0F824c
call
0: bool: true

```
[call] from: 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB to: Election.voters(address) data: 0xa3e...cabab
transact to Election.vote pending ...

[vm] from: 0x787...cabaB to: Election.vote(uint256) 0x99C...C196d value: 0 wei data: 0x012...00001 logs: 0
hash: 0x89f...af9e5
transact to Election.vote errored: Error occurred: revert.

revert
        The transaction has been reverted to the initial state.
Note: The called function should be payable if you send value and the value you send should be less than your current balance.
You may want to cautiously increase the gas limit if the transaction went out of gas.
```



```
CURRENT BLOCK  GAS PRICE    GAS LIMIT  HARDFORK  NETWORK ID  RPC SERVER              MINING STATUS  WORKSPACE
1              20000000000  6721975    MERGE     5777        HTTP://127.0.0.1:7545   AUTOMINING     QUICKSTART

BLOCK 1   MINED ON 2024-09-03 11:35:45   GAS USED 21000   1 TRANSACTION
BLOCK 0   MINED ON 2024-09-03 11:21:20   GAS USED 0       NO TRANSACTIONS
```



remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.26+commit.8a97fa7a.js

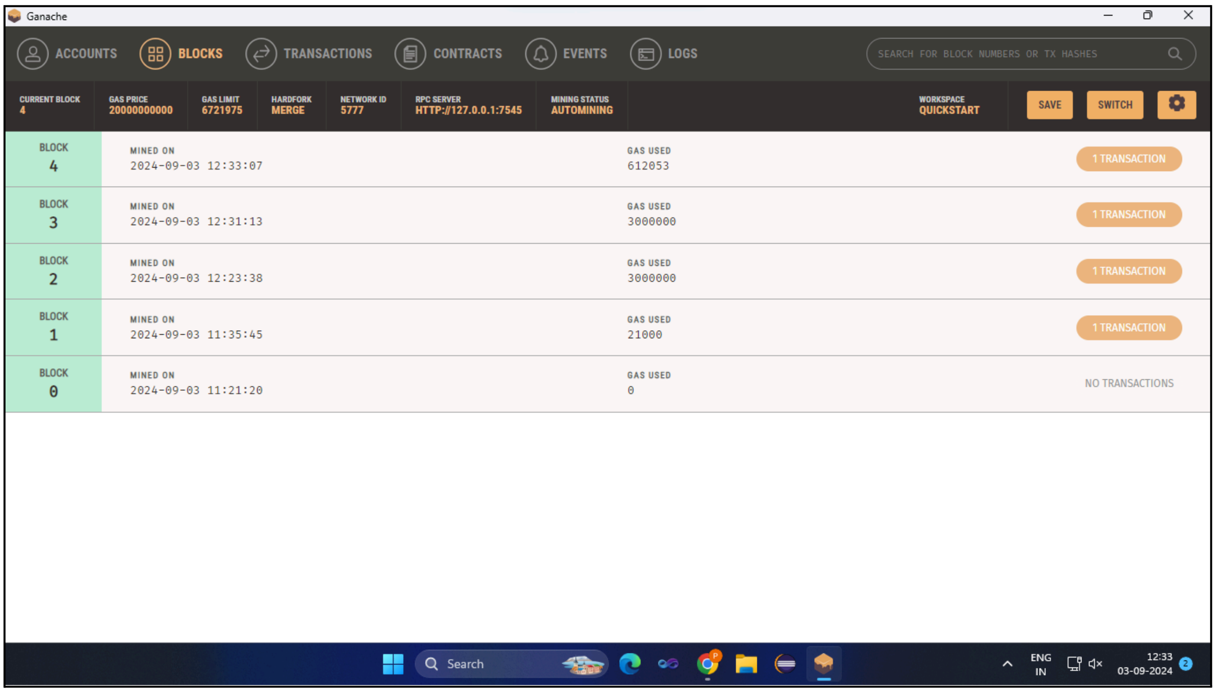
DEPLOY & RUN TRANSACTIONS

ENVIRONMENT: Injected Provider - MetaMask (Custom (5777) network)
ACCOUNT: 0x546...37565 (89.99958 ether)
GAS LIMIT: Estimated Gas / Custom 3000000
VALUE: 0 Ether
CONTRACT: Election - election_contract/electio
evm version: cancun

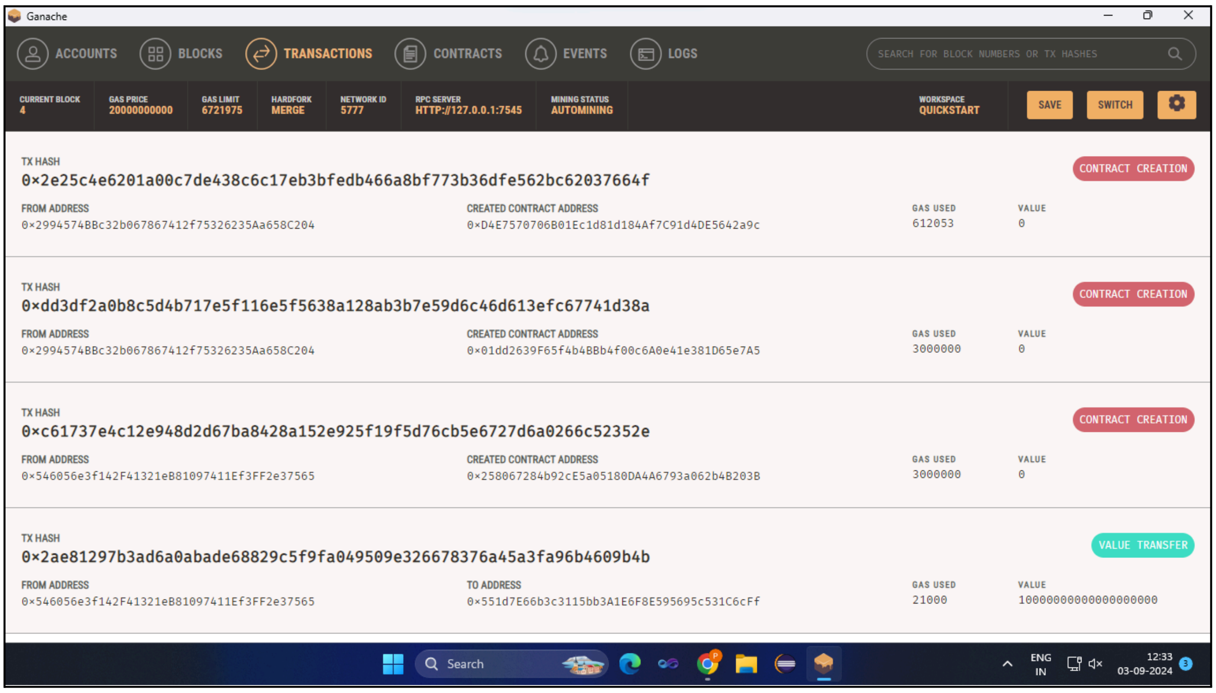
```
// voted event, will fetch the candidate id whom you have voted
event votedEvent (
uint indexed _candidateId
);
constructor()  {     infinite gas 311800 gas
addCandidate("N MODI, BJP");
addCandidate("A kejriwal, AAP");
addCandidate("Rahul G, Congress");
addCandidate("Nikhil, JDS");
```

```
[vm]  from: 0xCA3...a733c to: Election.vote(uint256) 0x692...77b3A value: 0 wei data: 0x012...00001 logs: 1
hash: 0x1b9...e0772
call to Election.voters

[call]  from: 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c to: Election.voters(address) data: 0xa3e...a733c
creation of Election pending...

creation of Election errored: Error occurred: Internal JSON-RPC error..

Internal JSON-RPC error.

You may want to cautiously increase the gas limit if the transaction went out of gas.
```

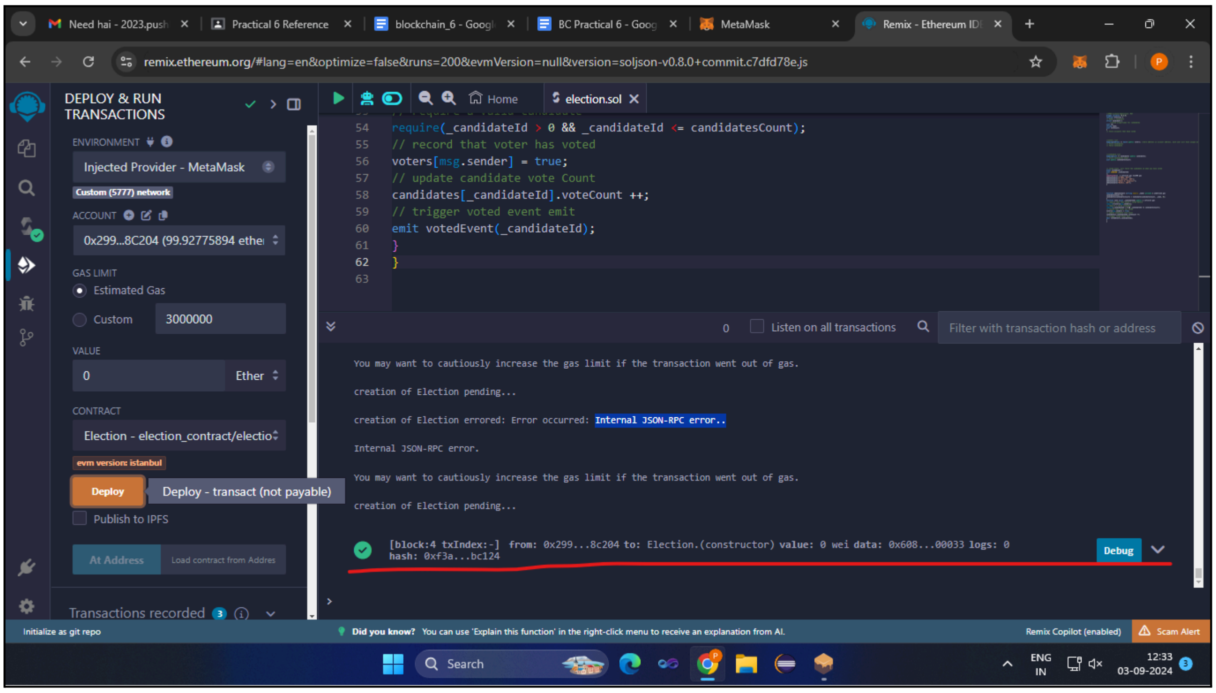
Enter custom Gas Limit.
Deploy
Debug
Did you know? You can use 'Explain this function' in the right-click menu to receive an explanation from AI.



```
CURRENT BLOCK  GAS PRICE    GAS LIMIT  HARDFORK  NETWORK ID  RPC SERVER              MINING STATUS   WORKSPACE
4              20000000000  6721975    MERGE     5777        HTTP://127.0.0.1:7545   AUTOMINING      QUICKSTART

BLOCK 4   MINED ON 2024-09-03 12:33:07   GAS USED 612053    1 TRANSACTION
BLOCK 3   MINED ON 2024-09-03 12:31:13   GAS USED 3000000   1 TRANSACTION
BLOCK 2   MINED ON 2024-09-03 12:23:38   GAS USED 3000000   1 TRANSACTION
BLOCK 1   MINED ON 2024-09-03 11:35:45   GAS USED 21000     1 TRANSACTION
BLOCK 0   MINED ON 2024-09-03 11:21:20   GAS USED 0         NO TRANSACTIONS
```



```
CURRENT BLOCK  GAS PRICE    GAS LIMIT  HARDFORK  NETWORK ID  RPC SERVER              MINING STATUS   WORKSPACE
4              20000000000  6721975    MERGE     5777        HTTP://127.0.0.1:7545   AUTOMINING      QUICKSTART

TX HASH                                                                                   CONTRACT CREATION
0×2e25c4e6201a00c7de438c6c17eb3bfedb466a8bf773b36dfe562bc62037664f
FROM ADDRESS                                 CREATED CONTRACT ADDRESS                     GAS USED  VALUE
0×2994574BBc32b067867412f75326235Aa658C204   0×D4E7570706B01Ec1d81d184Af7C91d4DE5642a9c   612053    0

TX HASH                                                                                   CONTRACT CREATION
0×dd3df2a0b8c5d4b717e5f116e5f5638a128ab3b7e59d6c46d613efc67741d38a
FROM ADDRESS                                 CREATED CONTRACT ADDRESS                     GAS USED  VALUE
0×2994574BBc32b067867412f75326235Aa658C204   0×01dd2639F65f4b4BBb4f00c6A0e41e381D65e7A5   3000000   0

TX HASH                                                                                   CONTRACT CREATION
0×c61737e4c12e948d2d67ba8428a152e925f19f5d76cb5e6727d6a0266c52352e
FROM ADDRESS                                 CREATED CONTRACT ADDRESS                     GAS USED  VALUE
0×546056e3f142F41321eB81097411Ef3FF2e37565   0×258067284b92cE5a05180DA4A6793a062b4B203B   3000000   0

TX HASH                                                                                   VALUE TRANSFER
0×2ae81297b3ad6a0abade68829c5f9fa049509e326678376a45a3fa96b4609b4b
FROM ADDRESS                                 TO ADDRESS                                   GAS USED  VALUE
0×546056e3f142F41321eB81097411Ef3FF2e37565   0×551d7E66b3c3115bb3A1E6F8E595695c531C6cFf   21000     10000000000000000000
```



```
require(_candidateId > 0 && _candidateId <= candidatesCount);
// record that voter has voted
voters[msg.sender] = true;
// update candidate vote Count
candidates[_candidateId].voteCount ++;
// trigger voted event emit
emit votedEvent(_candidateId);
}
}
```

```
You may want to cautiously increase the gas limit if the transaction went out of gas.

creation of Election pending...

creation of Election errored: Error occurred: Internal JSON-RPC error..

Internal JSON-RPC error.

You may want to cautiously increase the gas limit if the transaction went out of gas.

creation of Election pending...

[block:4 txIndex:-] from: 0x299...8c204 to: Election.(constructor) value: 0 wei data: 0x608...00033 logs: 0
hash: 0xf3a...bc124
```

Conclusion:

In this way , the installation of Ganache(Personal block chain) and MetaMask, Compilation and deployment of an election smart contract in the personal blockchain using injected web3 environment(MetaMask wallet) has been done successfully.