

Name: Vaneet Singla

Assignment-3

The screenshot shows the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel displays the 'VOTINGSYSTEM' contract at address 0x7EF...8CB4. The contract's state is shown with a balance of 0 ETH and a list of candidates: Vaneet (ID 0, 20 votes), Girish (ID 1, 21 votes), and Sumit (ID 2, 54 votes). The 'vote' button is highlighted. The main editor shows the Solidity code for the 'VotingSystem' contract, which includes a 'Candidate' struct, a 'candidates' array, and functions for 'getCandidates', 'vote', and 'getResult'.

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity >=0.7.0 <0.9.0;
3 contract VotingSystem {
4     struct Candidate {
5         uint id;
6         string name;
7         uint age;
8         string sex;
9         uint income;
10        uint voteCount;
11    }
12    Candidate[] public candidates;
13    constructor() {
14        // Add 3 to 4 candidates
15        candidates.push(Candidate(0, "Vaneet", 20, "Male", 50000, 1));
16        candidates.push(Candidate(1, "Girish", 21, "Female", 75000, 1));
17        candidates.push(Candidate(2, "Sumit", 54, "Male", 35000, 0));
18        // You can add more candidates as needed
19    }
20
21    function getCandidates() public view returns (Candidate[] memory) {
22        // Return the candidate array
23        return candidates;
24    }
25
26    function vote(uint candidateId) public {
27        // Find the candidate with the given ID
28        for (uint i = 0; i < candidates.length; i++) {
```

The screenshot shows the Remix IDE interface with the 'VOTINGSYSTEM' contract. The 'getResult' function is being executed, and the output is displayed in the 'Low level interactions' panel. The output shows the candidate names and their vote counts: '0: string Vaneet: 1 Girish: 2 Sumit: 0'. The main editor shows the Solidity code for the 'VotingSystem' contract, including the 'getResult' function and a helper function 'toString'.

```
28 for (uint i = 0; i < candidates.length; i++) {
29     if (candidates[i].id == candidateId) {
30         // Increment the candidate's vote count
31         candidates[i].voteCount++;
32         break;
33     }
34 }
35
36 function getResult() public view returns (string memory) {
37     string memory result = "";
38     for (uint i = 0; i < candidates.length; i++) {
39         // Concatenate the candidate name and vote count
40         result = string(abi.encodePacked(result, candidates[i].name, ": ", toString(candidates[i].voteCount), "\n"));
41     }
42     return result;
43 }
44
45 // Helper function to convert uint to string
46 function toString(uint value) internal pure returns (string memory) {
47     if (value == 0) {
48         return "0";
49     }
50     uint temp = value;
51     uint digits;
52     while (temp != 0) {
53         digits++;
54         temp /= 10;
55     }
```

DEPLOY & RUN TRANSACTIONS

OR

At Address Load contract from Address

Transactions recorded 6 1

Deployed Contracts

VOTINGSYSTEM AT 0x7EF...8CB4

Balance: 0 ETH

vote 1

candidates uint256

getCandidates

0: tuple(uint256,string,uint256,string,uint256,uint256)[]: 0.Vaneet,20,Male,50000,1,1, Girish,21,Female,75000,1,2,Sumit,54,Male,35000,0

getResult getResult - call

0: string: Vaneet: 1 Girish: 2 Sumit: 0

Low level interactions

CALLDATA

Transact

Home

ass4.sol

```
49 if (value == 0) {
50     return "0";
51 }
52 uint temp = value;
53 uint digits;
54 while (temp != 0) {
55     digits++;
56     temp /= 10;
57 }
58 bytes memory buffer = new bytes(digits);
59 while (value != 0) {
60     digits -= 1;
61     buffer[digits] = bytes1(uint8(48 + uint(value % 10)));
62     value /= 10;
63 }
64 return string(buffer);
65 }
```

listen on all transactions

Search with transaction hash or address

Debug

[call] from: 0x58380a6a701c56854dCfc803fc8875f56beddC4 to: VotingSystem.getResult() data: 0xde2...92789

Balance: 0 ETH

vote 1

candidates uint256

getCandidates

0: tuple(uint256,string,uint256,string,uint256,uint256)[]: 0.Vaneet,20,Male,50000,1,1, Girish,21,Female,75000,1,2,Sumit,54,Male,35000,0

getResult getResult - call

0: string: Vaneet: 1 Girish: 2 Sumit: 0

Low level interactions

CALLDATA

Transact