**Assignment 4**

**Title:-**

Write a program in solidity to create student data.use the following constructs:

1. Structure
2. Arrays
3. Fallback

Deploy this as a smart contract on Ethereum and observe the transaction fee and gas value.

**Program :-**

//SPDX-License-Identifier: UNLICENSED

pragma solidity >=0.7.0 <0.9.0;

// Build the Contract

contract MarksManagmtSys {

    // Create a structure for

    // student details

    struct StudentStruct {

        uint256 ID;

        string fName;

        string lName;

        uint256 marks;

    }

    address owner;

    uint256 public stdCount = 0;

    //Create Array to store Student data

    StudentStruct[] stdRecords;

    constructor() {

        owner = msg.sender;

    }

    // Create a function to add

    // the new records

    function addNewRecords(

        uint256 \_ID,

        string memory \_fName,

        string memory \_lName,

        uint256 \_marks

    ) public payable {

        // Increase the count by 1

        stdCount = stdCount + 1;

        //Adding data into array

        stdRecords.push(StudentStruct(\_ID, \_fName, \_lName, \_marks));

    }

    function getAllRecords() public view returns (StudentStruct[] memory) {

        return stdRecords;

    }

}

Output :-

**[vm]**

**from:** 0x5B3...eddC4

**to:** MarksManagmtSys.addNewRecords(uint256,string,string,uint256) 0xd91...39138

**value:** 0 wei

**data:** 0xc7c...00000

**logs:** 0

**hash:** 0x31e...8222f

**Debug**

|  |  |
| --- | --- |
| **status** | true Transaction mined and execution succeed |
| **transaction hash** | 0x31e9bf92fde0d7e50ee3ca03e54961dfd631ab6a7fd695e84ec59eb85de8222f |
| from | 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 |
| **to** | MarksManagmtSys.addNewRecords(uint256,string,string,uint256) 0xd9145CCE52D386f254917e481eB44e9943F39138 |
| **gas** | 142641 gas |
| **transaction cost** | 124035 gas |
| **execution cost** | 124035 gas |
| **input** | 0xc7c...00000 |
| **decoded input** | { "uint256 \_ID": "2", "string \_fName": "rohit", "string \_lName": "kharade", "uint256 \_marks": "100" } |
| **decoded output** | {} |
| **logs** | [] |
| **val** | 0 wei |

call to MarksManagmtSys.getAllRecords

***CALL*[call]**

**from:** 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

**to:** MarksManagmtSys.getAllRecords()

**data:** 0xa7f...9fe72

**Debug**

|  |  |
| --- | --- |
| from | 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 |
| **to** | MarksManagmtSys.getAllRecords() 0xd9145CCE52D386f254917e481eB44e9943F39138 |
| **execution cost** | 46616 gas (Cost only applies when called by a contract) |
| **input** | 0xa7f...9fe72 |
| **decoded input** | {} |
| **decoded output** | { "0": "tuple(uint256,string,string,uint256)[]: 1,ayush,shinde,100,2,rohit,kharade,100" } |
| **logs** | [] |

call to MarksManagmtSys.stdCount

***CALL*[call]**

**from:** 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

**to:** MarksManagmtSys.stdCount()

**data:** 0xfe6...15f2a

**Debug**

|  |  |
| --- | --- |
| from | 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 |
| **to** | MarksManagmtSys.stdCount() 0xd9145CCE52D386f254917e481eB44e9943F39138 |
| **execution cost** | 23515 gas (Cost only applies when called by a contract) |
| **input** | 0xfe6...15f2a |
| **decoded input** | {} |
| **decoded output** | { "0": "uint256: 2" } |

