

Code:

```
pragma solidity 0.8.11;
// SPDX-License-Identifier: UNLICENSED

contract StudentInfo {
    struct Student {
        string id;
        string name;
        string contactNo;
    }
    Student[20] studInfoArray;
    uint i=0;
    function registerStudent(string memory _stud_id, string memory _name, string memory _contact) public returns(string memory) {
        Student memory student = Student(_stud_id, _name, _contact);

        if(i > 20) {
            return "Memory limit exhausted";
        }
        else {
            studInfoArray[i] = student;
            i += 1;
            return "Student registered successfully!";
        }
    }

    function getStudent(uint idx) public view returns(string memory){
        Student memory student = studInfoArray[idx];
        return string(bytes.concat("Name: ", bytes(student.name), ", Contact No.: ", bytes(student.contactNo), ", Seat No: ", bytes(student.id)));
    }
}

function getStudent(uint idx) public view returns(string memory){
    Student memory student = studInfoArray[idx];
    return string(bytes.concat("Name: ", bytes(student.name), ", Contact No.: ", bytes(student.contactNo), ", Seat No: ", bytes(student.id)));
}
// The solidity fallback function is executed if none of the other functions match the function identifier
// or no data was provided with the function call. Only one unnamed function can be assigned to a contract

fallback(bytes calldata) external payable returns (bytes memory){
    return "Fallback function called";
}
receive() external payable{
}
```

Output:

Balance: 0 ETH

registerStudent

_stud_id: 1

_name: Abhijeet

_contact: 9765456445



Calldata



Parameters

transact

getStudent

1

0: string: Name: Abhijeet, Contact No.: 9765456445, Seat No: 1

✓

[vm] from: 0x93e...00000 data: 0x93e...00000 logs: 0 hash: 0xbf0...eacc3

Debug

^

status

true Transaction mined and execution succeed

transaction hash

0xbf0ae2fd4f69407fd37c5d736bea7a12250fbc9d04b766d4df8b2f06da5eacc3

from

0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to

StudentInfo.registerStudent(string,string,string) 0x358AA13c52544ECCEf6B0ADD0f801012ADAD5eE3

gas

113488 gas

transaction cost

98685 gas

execution cost

98685 gas

input

0x93e...00000

decoded input

{

"string _stud_id": "1",

"string _name": "Abhijeet",

"string _contact": "9765456445"

}

decoded output

{

"0": "string: Student registered successfully!"

}

logs

[]

val

0 wei

> 1000

CALL

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: StudentInfo.getStudent(uint256) data: 0x642...00001

Debug

^

from

0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to

StudentInfo.getStudent(uint256) 0x358AA13c52544ECCEf6B0ADD0f801012ADAD5eE3

execution cost

32034 gas (Cost only applies when called by a contract)

input

0x642...00001

decoded input

{

"uint256 idx": "1"

}

decoded output

{

"0": "string: Name: Abhijeet, Contact No.: 9765456445, Seat No: 1"

}

logs

[]