Assignment 4

Rajdhani
DATE / /

Aim:-

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

1. structures.
2. Arrays.
3. fallback

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

Objectives:-

1. Understand the working of blockchain.
2. Learn about smart contract.
3. Implement smart contracts on a live network.

Requirements:-

- Any browser.
- Remix IDE.
- Metamask wallet.

Theory:-

Smart Contract:

Smart contracts are immutable programs stored on a blockchain. They automate the execution of transactions based on predetermined conditions being met, and they are widely used to execute agreements in a decentralized manner without middlemen.

Smart contracts have particular outcomes, which are governed by immutable code, so the participants in the contract can be confident in the contract execution. No third-party involvement, no time-lost aggrements are executed immediately when the conditions are met.

Smart contracts can be deployed on the blockchain for use. Ethereum supports smart contracts written in Solidity.

Solidity's code is encapsulated in contracts which means a contract in solidity is a collection of code and data that resides at a specific address on the Ethereum blockchain.
A contract is a fundamental block of building an application on Ethereum.

Approach :-

1. The first step is to deploy the smart contract using the Remix IDE. After writing the code, compile the code. When it is successfully compiled, then deploy it. After deploying the contract, a deployed contract is obtained and then add the student details one by one.
2. If bonus marks need to be added then add in the bonusMarks selection after that click on stdCount and fetch the student details to call the stdRecords.

Rajdhani
DATE / /

3. Add one or more new student details in this smart contract by the increment of stdCount.

Creating an array:-

To declare an array in solidity, the data type of the elements and the number of elements should be specified. The size of the array must be a positive integer and data type should be a valid solidity type.

Syntax:-

<data type><array name>[size] = <initialization>

Fixed-size arrays:-

The size of the array should be pre-defined. The total number of elements should not exceed the size of the array.

If the size of the array is not specified then the array of enough size is created which is enough to hold the initialization.

```
function array_example() public returns(
  int[5] memory, uint[6] memory) {
    int[5] memory data
    = [int(50), -63, 77, -28, 90];
    data1
    = [uint(10), 20, 30, 40, 50, 60];
    return (data, data1);
}
```

Rajdhani
DATE / /

Fall Back Function:

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 and it is executed whenever the contract receives plain Ether without any data.

Properties of a fallback function:

1) Has no name or arguments.
2) If it is not marked payable, the contract will throw an exception if it receives plain Ether without data.
3) Cannot return anything.
4) Can be defined once per contract.
5) It is also executed if the caller meant to call a function that is not available.
6) It is mandatory to mark it external.
7) It is limited to 2300 gas when called by another function. It is so for as to make this function.
8) Call as cheap as possible.

```
function setX(uint _x) public returns (bool) {
    x = _x;
    return true;
  }
function () public payable
  {
    balance[msg.sender] += msg.value;
  }
```

Rajdhani
DATE / /

```
}
 contract sender
{
 function transfer () public payable
{
 address receiver = ------;
- receiver.transfer (100);
}
```

ner

Step 1: Open Remix IDE.

Step 2: Click on file explorers and select solidity in the environment and create a new file. StudentMarksMangmtSys.sol by clicking on the New file section.

Step 3: Build a smart contract that contains all the details of the student with the help of Remix IDE by clicking on the file name.

Step 4: After building the contract, compile it. Select the compiler version before clicking on Compile button.

Step 5: If the contract is successfully deployed, then deployed contract is obtained. Open the deployed contract and add the student details and transact it.

Step 6: Add the bonus marks if you want to give them to the student and transact it after that click on the stdCount. One can see the student details after calling the stdRecords by entering the stdCount

Conclusion:-

Hence, we have studied about Solidity program and learnt how to develop test, and deploy smart contract on Ethereum.