

Name: Krishnaprasad Awala

Roll No.: D2215004

Enrollment No.: MITU21BTITD006

Class: LY IT Core

Assignment No. 6: Solidity (Part 2)

Execute the following code in remix ide (<https://remix.ethereum.org/>)

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity >= 0.6.0 < 0.9.0;

/**
 * @title Storage
 * @dev Store & retrieve value in a variable
 * @custom:dev-run-script contracts/test2.sol
 */

contract pay {
    address payable user = payable
(0x5B38Da6a701c568545dCfcB03FcB875f56beddC4);

    function payEther() public payable {

    }

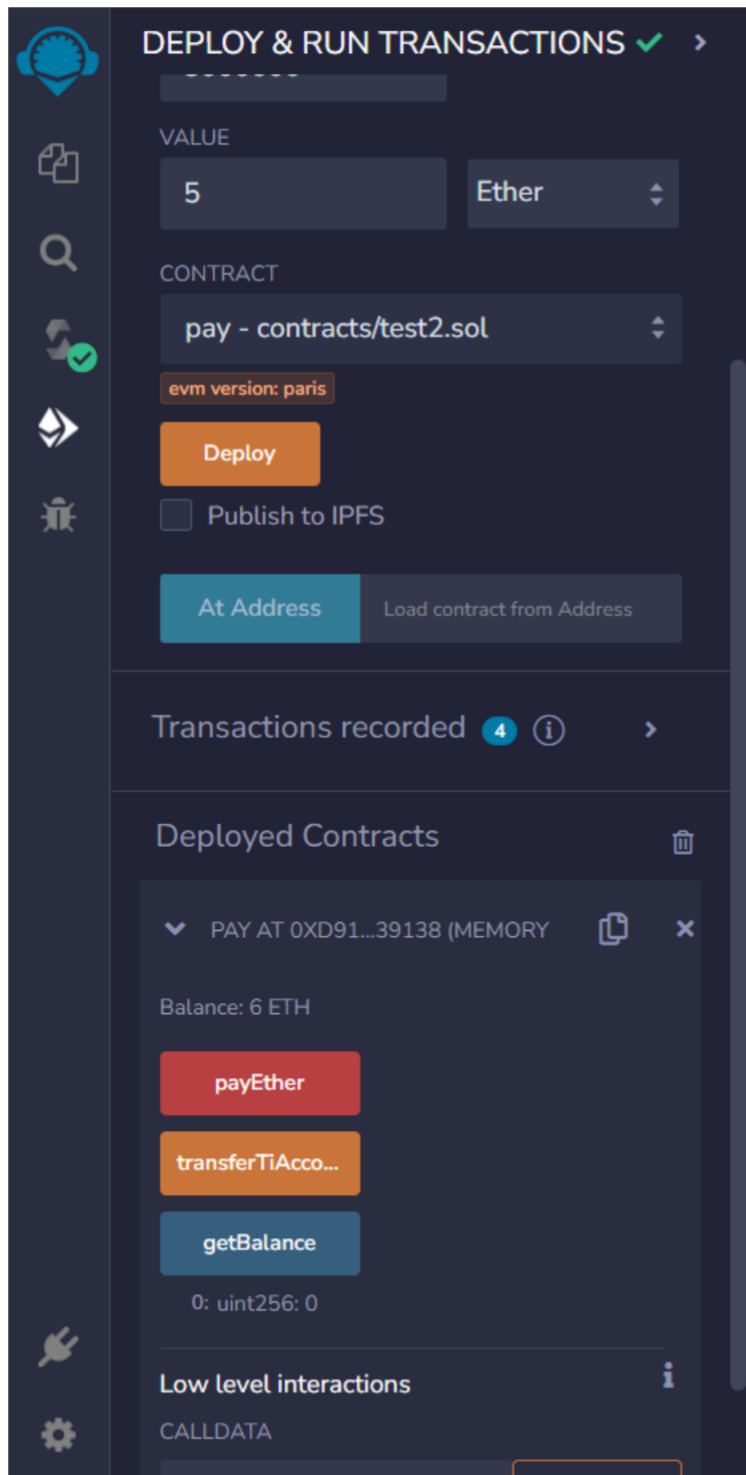
    function getBalance() public view returns (uint){
        return address(this).balance;
    }

    function transferTiAccount() public {
        user.transfer(2 ether);
    }
}
```

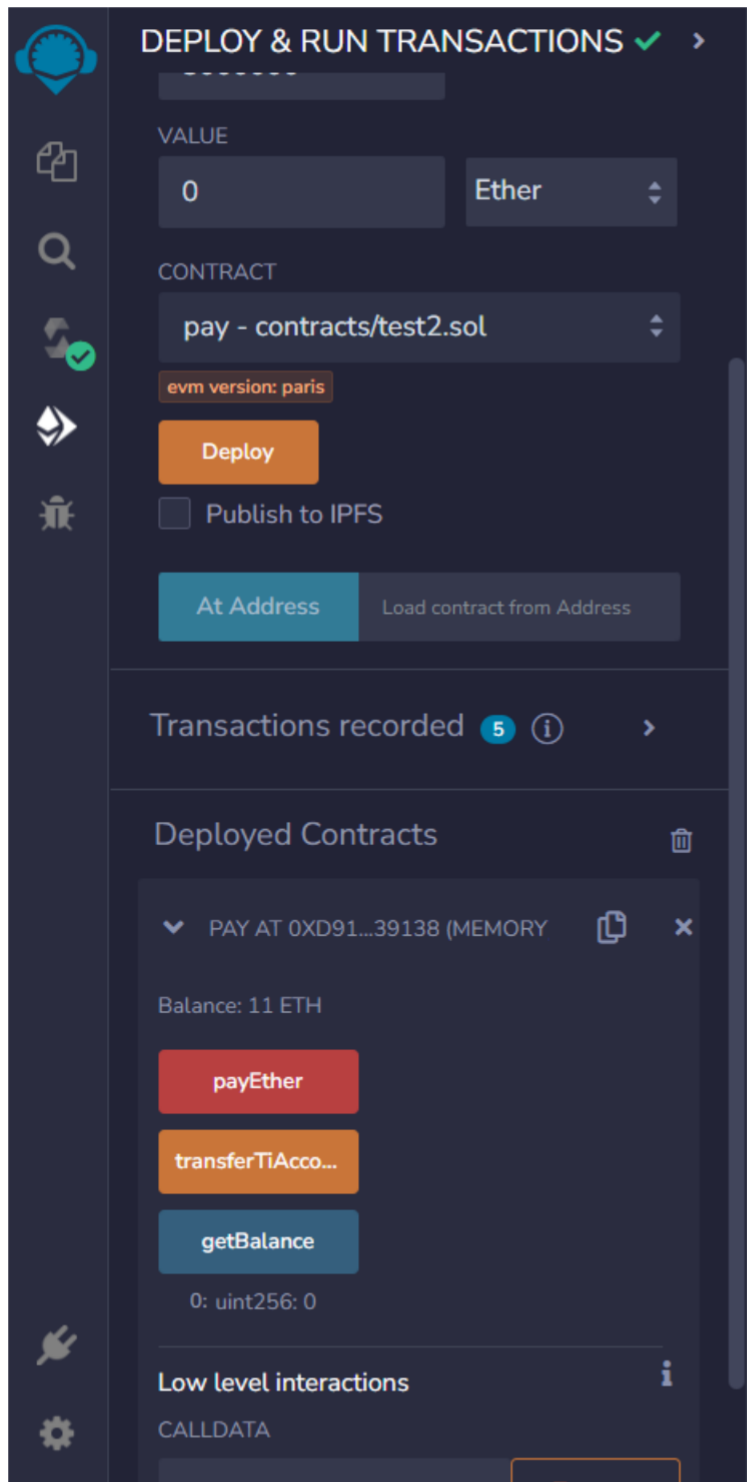
Output:

The screenshot displays the Remix Ethereum IDE interface. The top bar shows the browser address: `remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js`. The left sidebar contains the 'DEPLOY & RUN TRANSACTIONS' panel, which includes a 'CONTRACT' section with a dropdown menu set to 'pay - contracts/test2.sol', a 'Deploy' button, and a 'Publish to IPFS' checkbox. Below this, the 'Transactions recorded' section shows a list of transactions, and the 'Deployed Contracts' section shows a contract named 'PAY AT 0xD91...39138 (MEMORY)' with a balance of 2 ETH. The 'Low level interactions' section shows a 'CALLDATA' field and a 'Transact' button. The main editor area displays the Solidity code for the 'test2.sol' contract, which includes a pragma statement, a license identifier, and three functions: 'payEther', 'getBalance', and 'transferTiAccount'. The bottom status bar shows a successful transaction: '[vm] from: 0x583...edc4 to: pay.payEther() 0xd91...39138 value: 2000000000000000 wei data: 0xf74...74d00 logs: 0 hash: 0xe56...58c87'.

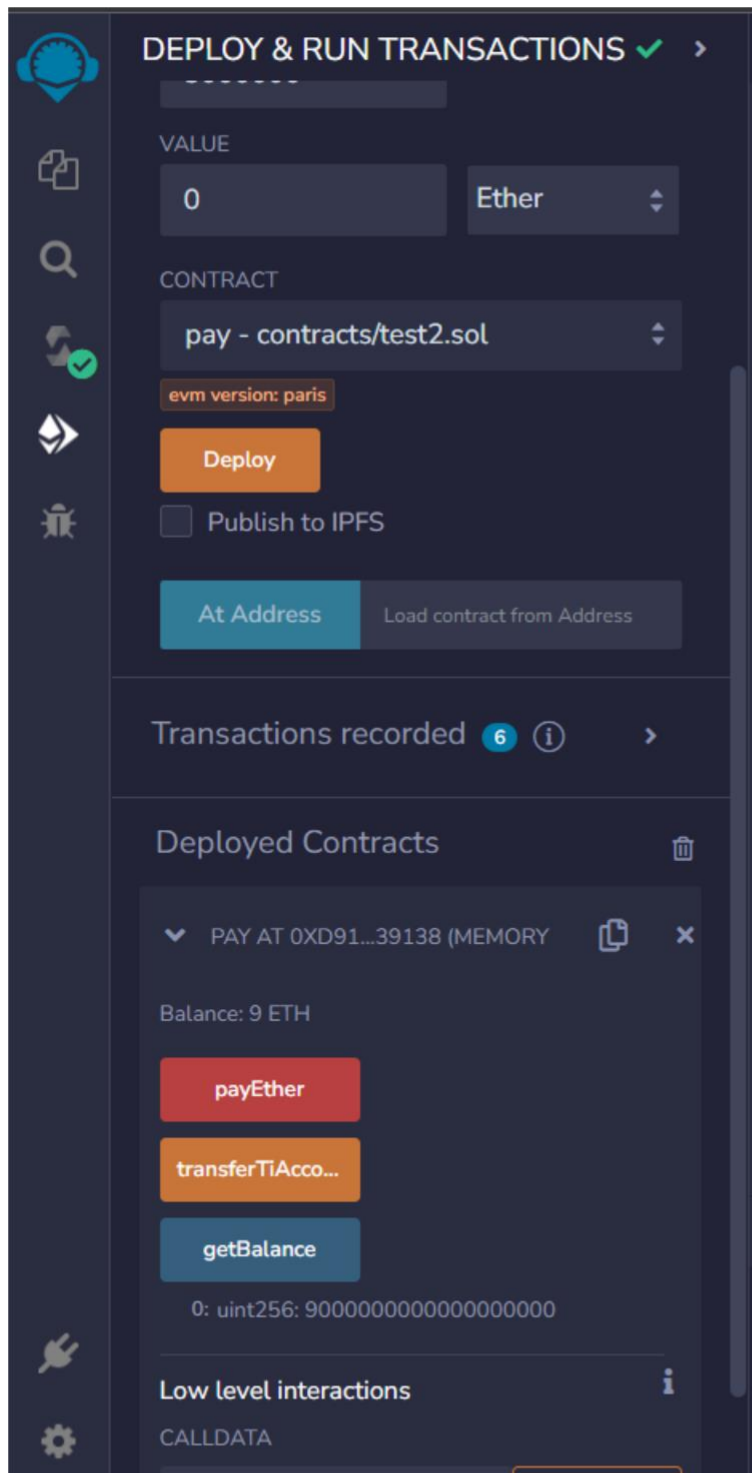
```
1 // SPDX-License-Identifier: GPL-3.0
2
3 pragma solidity >= 0.6.0 < 0.9.0;
4
5 /**
6  * @title Storage
7  * @dev Store & retrieve value in a variable
8  * @custom:dev-run-script contracts/test2.sol
9  */
10
11 contract pay {
12     address payable user = payable (0x58380a6a701c568545dcfc803Fc8875f56beddc4);
13
14     function payEther() public payable { 142 gas
15
16     }
17
18     function getBalance() public view returns (uint){ 339 gas
19         return address(this).balance;
20     }
21
22     function transferTiAccount() public { Infinite gas
23         user.transfer(2 ether);
24     }
25 }
```



Get Balance using getBalance function (current balance 6 ETH)



Add Ether to account (adding of 5 ETH) (current balance 11 ETH)



Transfer ether to account (2 ETH Transferred) (current balance 9 ETH)