EX.NO: 1 **Basic Programs Using Solidity**

Date:27/06/2023

Aim:

To implement the basic programs using solidity smart control in remix IDE

Procedure:

Step 1: Open the Remix IDE
Step 2: Create a new file by using .sol Extension

Step 3: Rename the file and give the basic coding for the basic programs
Step 4: Select the Compiler version using compilation section
Step 5: Also select the Environment to compile the program
Step 6: Compile the program by using Compile option of the file
Step 7: Then Deploy the compiled program for transaction
Step 8: Get the Input of the program by using the get and set method
Step 9: Then transact or call the input
Step 10: Output will be printed then stop the transaction

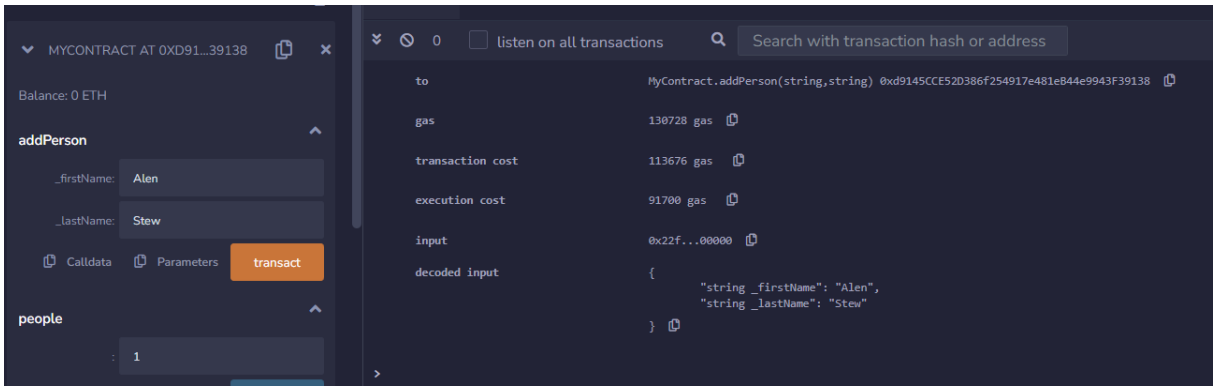
Programs:

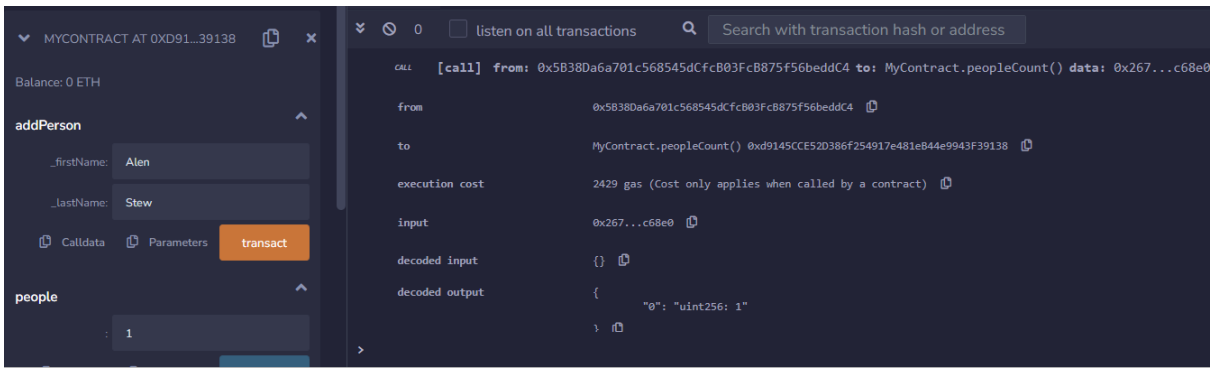
1.pragma solidity >= 0.5.1< 0.9.0;

```
contract MyContract {
  Person[] public people;
  uint256 public peopleCount;
  struct Person {
     string _firstName;
     string _lastName;
  }
  function addPerson(string memory _firstName, string memory _lastName) public {
     people.push(Person(_firstName, _lastName));
     peopleCount += 1;
  }
```

}

Output:

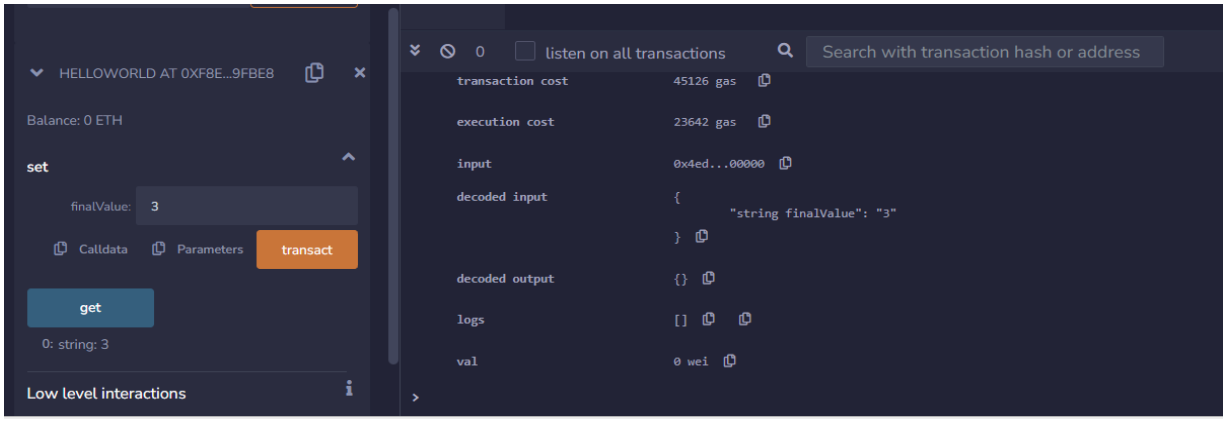


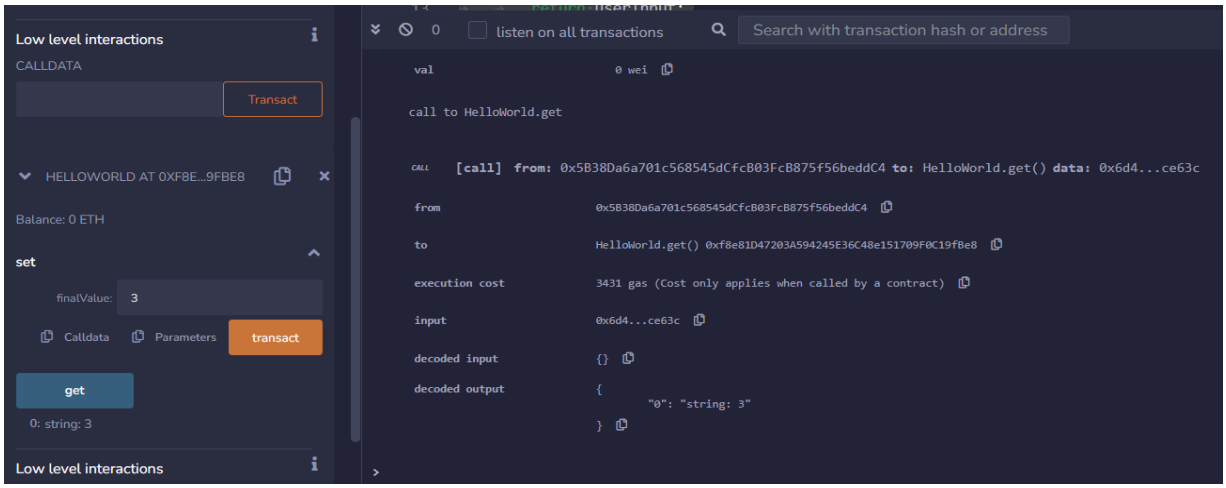


2. pragma solidity >= 0.5.0 <0.9.0;

```
  contract HelloWorld{
   string userInput;
   function set(string memory finalValue) public
   {
      userInput = finalValue;
   }
   function get() public view returns(string memory){
      return userInput;
   }
}
```

Output:

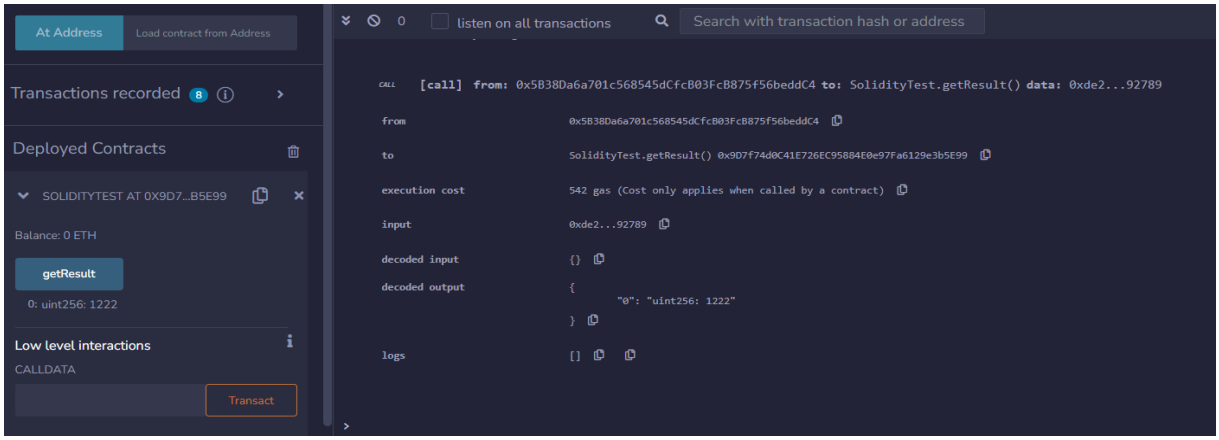




3. pragma solidity >=0.5.1 < 0.9.0;

```
contract SolidityTest {
        constructor() public{
        }
        function getResult() public view returns(uint){
                uint a = 1000;
                uint b = 222;
                uint result = a + b;
                return result;
        }
}
```

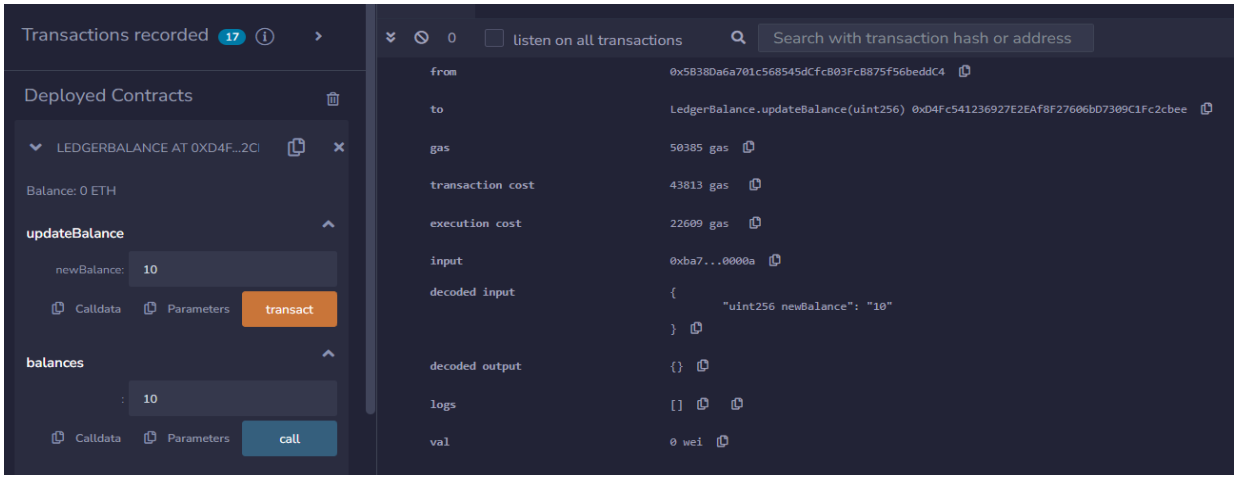
Output:



4. pragma solidity >=0.5.0<0.9.1;

```
 contract LedgerBalance {
        mapping(address => uint) public balances;
        function updateBalance(uint newBalance) public {
                balances[msg.sender] = newBalance;
        }
}
 contract Updater {
        function updateBalance() public returns (uint) {
                LedgerBalance ledgerBalance = new LedgerBalance();
                ledgerBalance.updateBalance(10);
                return ledgerBalance.balances(address(this));
        }
}
```

Output:



Result:

- Thus all the basic programs has been implemented and executed out successfully using remix IDE.