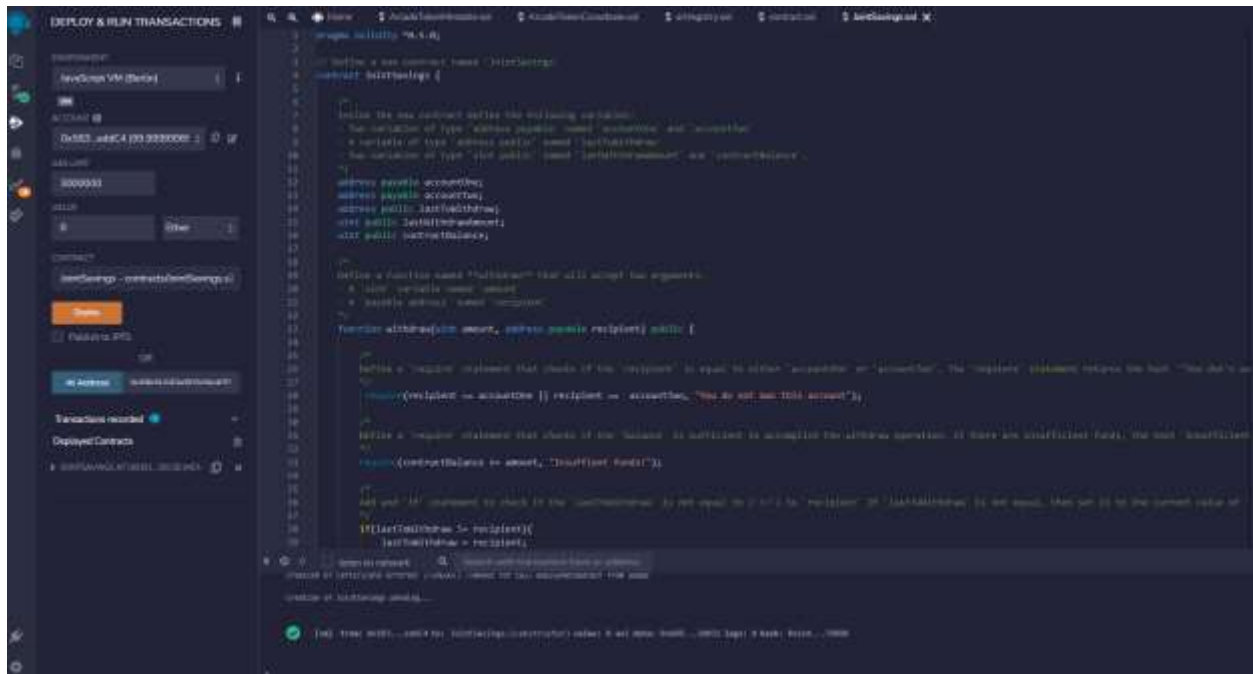


Compiled ^



Deployed ^

The screenshot displays the Remix IDE interface, which is used for developing and deploying smart contracts. The interface is divided into several panels:








- Left Panel (Deploy & Run Transactions):** This panel contains options to deploy a contract to the Etherscan testnet or a local environment. It also shows a list of transactions recorded and a section for deployed contracts. The 'contractBalance' variable is visible in the deployed contracts section.
- Center Panel (Solidity Code Editor):** This panel shows the Solidity code for a contract. The code includes a 'contractBalance' variable, a 'deposit' function, and a 'setAccounts' function. The code is as follows:


```

11 contractBalance = address(this).balance;
12
13
14
15 // Define a 'public payable' function named 'deposit'.
16 function deposit() public payable {
17
18     /*
19      * Call the 'contractBalance' variable and set it equal to the balance of the contract by using 'address(this)'.
20      */
21     contractBalance = address(this).balance;
22 }
23
24
25 // Define a 'public' function named 'setAccounts' that receives two 'address payable' arguments named 'account1' and 'account2'.
26 function setAccounts(address payable account1, address payable account2) public {
27
28     /*
29      * Set the values of 'accountOne' and 'accountTwo' to 'account1' and 'account2' respectively.
30      */
31     accountOne = account1;
32     accountTwo = account2;
33 }
34
35
36 // Finally, add the "Default Fallback Function" so that your contract can store ether sent from outside the contract.
37 function() external payable {
38 }
39
40
41
42

```
- Right Panel (Transaction Log):** This panel shows the transaction log, which includes the transaction hash, the function being called, and the arguments passed. The log shows a transaction with the function 'setAccounts' and the arguments '0x00' and '0x00'.

Transaction 1: Send 1 ether as wei. ^



DEPLOY & RUN TRANSACTIONS

ENVIRONMENT

JavaScript VM (Berlin)

VM

ACCOUNT

0xAb8...35cb2 (89.999999%)

GAS LIMIT

3000000

VALUE

10 Ether

CONTRACT

JointSavings - contracts/JointSavings

Deploy

☐ Publish to IPFS

OR

At Address 0x306c5131E2e3970A5e167

Transactions recorded 8

All transactions (deployed contracts and function executions) in this environment can be saved and replayed in another environment. e.g Transactions created in Javascript VM can be replayed in the Injected Web3.








Deployed Contracts

JOINTSAVINGS AT 0XD91...39138 (MI)

deposit

setAccounts



account1: 0x0c0669Cd5e60a6F4b8ce4



DEPLOY & RUN TRANSACTIONS

Transactions recorded 9

All transactions (deployed contracts and function executions) in this environment can be saved and replayed in another environment. e.g Transactions created in Javascript VM can be replayed in the Injected Web3.



Deployed Contracts

JOINTSAVINGS AT 0XD91...39138 (MI)

deposit

setAccounts

withdraw

contractBalance

lastToWithdraw

lastWithdraw...

account1: 0x0c0669Cd5e60a6F4b8ce4

account2: 0x7A1f3dFAa0a4a19844B60

transact








amount: uint256

recipient: address

transact

o: uint256: 100000000000000000001

Transaction 2: Send 10 ether as wei ^



DEPLOY & RUN TRANSACTIONS

ENVIRONMENT

JavaScript VM (Berlin) ⓘ

VM

ACCOUNT ⓘ

0xAb8...35cb2 (89.999999%) ⓘ ⓘ

GAS LIMIT

3000000

VALUE

5 ⓘ Ether ⓘ

CONTRACT

JointSavings - contracts/JointSavings ⓘ

Deploy

☐ Publish to IPFS

OR

At Address 0x306c5131E2e3970A5e167

Transactions recorded 9 ⓘ

All transactions (deployed contracts and function executions) in this environment can be saved and replayed in another environment. e.g Transactions created in JavaScript VM can be replayed in the

DEPLOY & RUN TRANSACTIONS

Transactions recorded **11**

All transactions (deployed contracts and function executions) in this environment can be saved and replayed in another environment. e.g Transactions created in Javascript VM can be replayed in the Injected Web3.



Deployed Contracts



JOINTSAVINGS AT 0XD91...39138 (MI)



deposit

setAccounts



account1: 0x0c0669Cd5e60a6F4b8ce4'

account2: 0x7A1f3dFAa0a4a19844B60



transact

withdraw



amount: uint256

recipient: address



transact

contractBalance

0: uint256: 150000000000000000001

lastToWithdraw

lastWithdraw...

- Transaction 3: Send 5 ether.^