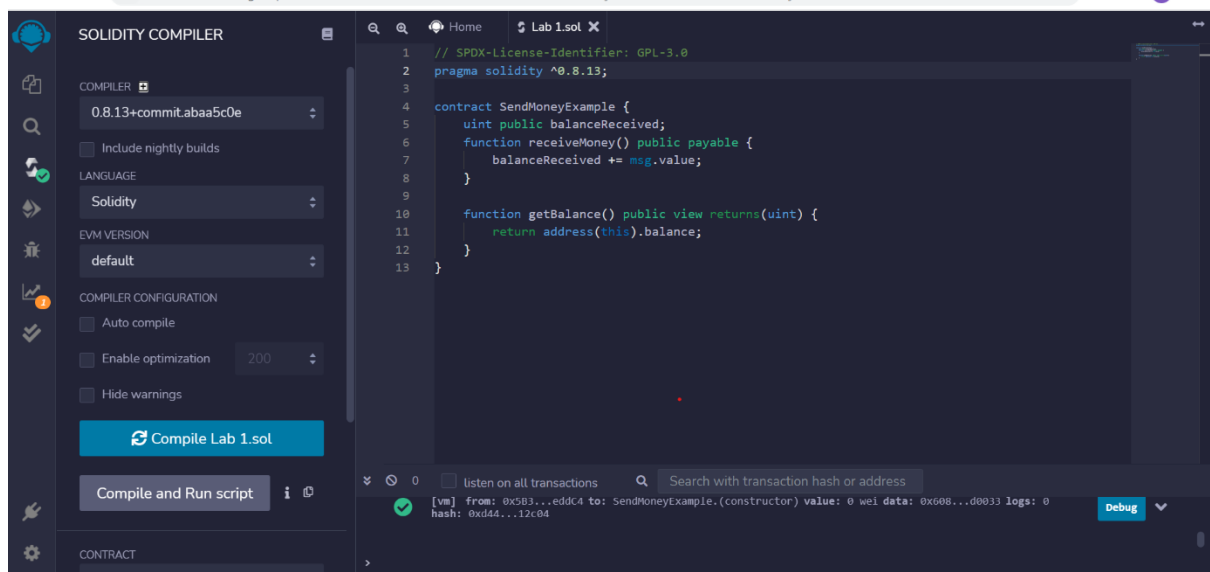


Nama : Weni Wilman Putri

NIM : 1103184134

UTS LAB 1



Penjelasan:

1. `uint public balanceReceived` yaitu variabel penyimpanan umum/publik. Variabel publik akan membuat fungsi pengambil secara otomatis di Solidity. Jadi Developer selalu bisa memonitor konten yang sedang ada pada variabel tersebut.
2. `BalanceReceived += msg.value` yaitu objek global yang slalu berisi tentang beberapa informasi tentang transaksi yang sedang berlangsung. Properti yang paling penting disini yaitu `value` dan `sender`.
3. `Function getBalance() public view returns(uint)` yaitu fungsi yang tidak dapat mengubah penyimpanan (read-only) dan dapat mengembalikan informasi.
4. `Address(this).balance` yaitu variabel tipe alamat yang slalu memiliki properti yang disebut `balance` yang memberi sejumlah Ether yang disimpan di alamat tersebut. Kita dapat mengakses dengan mudah, namun hanya memberitahu beberapa bayak yang disimpan disana. `Address(this)` ini mengonversi Smart Contract ke alamat.

Deploy & Run Transactions

The screenshot shows the 'DEPLOY & RUN TRANSACTIONS' panel in the Remix IDE. The 'ACCOUNT' field is set to '0x5B3...eddC4 (99.999999%)'. The 'GAS LIMIT' is set to '3000000'. The 'VALUE' is '0' Wei. The 'CONTRACT' dropdown shows 'SendMoneyExample - Lab 1.sol'. The 'Deploy' button is highlighted. Below it, there is a section for 'At Address' and 'Load contract from Address'. The 'Transactions recorded' section shows one transaction. The 'Deployed Contracts' section shows the contract 'SENDMONEYEXAMPLE AT 0xD91...3F'. The right panel shows the Solidity code for the 'SendMoneyExample' contract:

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.13;
3
4 contract SendMoneyExample {
5     uint public balanceReceived;
6     function receiveMoney() public payable {
7         balanceReceived += msg.value;
8     }
9
10    function getBalance() public view returns(uint) {
11        return address(this).balance;
12    }
13 }
```

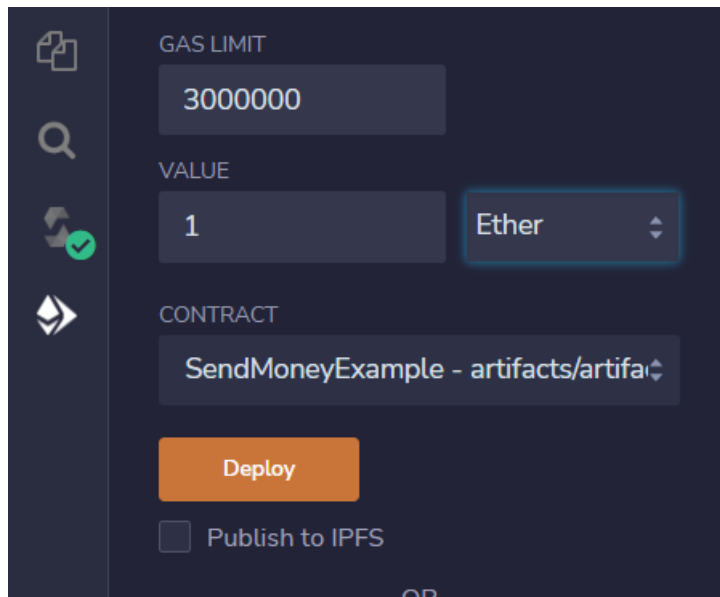
The bottom of the right panel shows the 'ContractDefinition SendMoneyExample' with 1 reference(s). The 'Debug' button is visible.

The screenshot shows the 'DEPLOY & RUN TRANSACTIONS' panel in the Remix IDE. The 'Publish to IPFS' checkbox is checked. The 'At Address' button is highlighted. The 'Transactions recorded' section shows one transaction. The 'Deployed Contracts' section shows the contract 'SENDMONEYEXAMPLE AT 0xD91...3F'. The 'receiveMoney' button is highlighted. The 'balanceReceived' button is also visible. The 'Low level interactions' section shows the 'CALLDATA' field and the 'Transact' button. The right panel shows the Solidity code for the 'SendMoneyExample' contract:

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.13;
3
4 contract SendMoneyExample {
5     uint public balanceReceived;
6     function receiveMoney() public payable {
7         balanceReceived += msg.value;
8     }
9
10    function getBalance() public view returns(uint) {
11        return address(this).balance;
12    }
13 }
```

The bottom of the right panel shows the 'ContractDefinition SendMoneyExample' with 1 reference(s). The 'Debug' button is visible.

Mengirim beberapa Ether ke Smart Contract



GAS LIMIT

3000000

VALUE

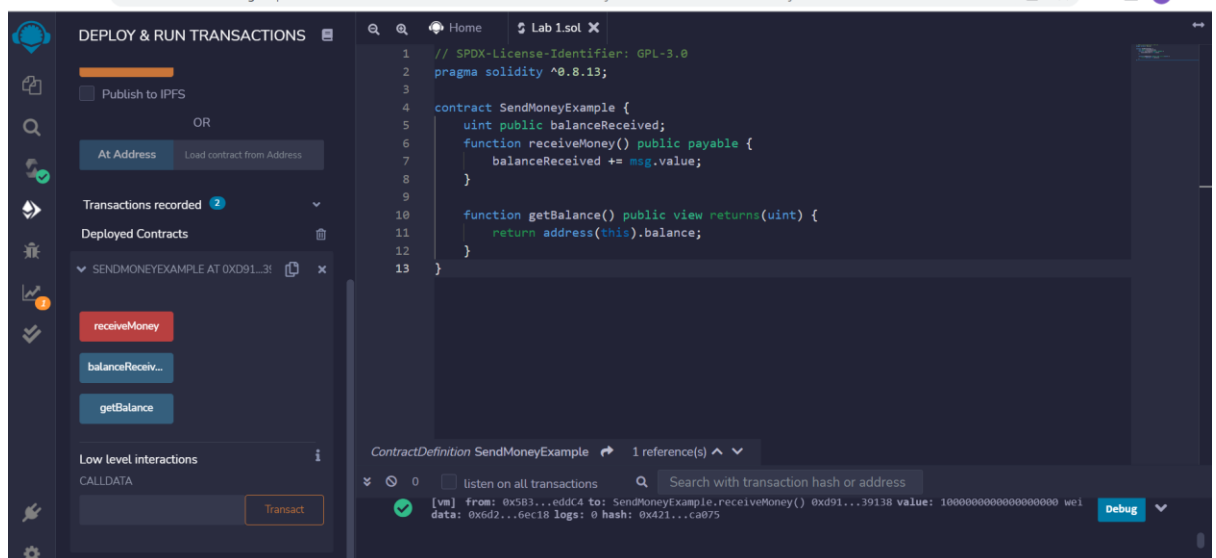
1 Ether

CONTRACT

SendMoneyExample - artifacts/artifa

Deploy

☐ Publish to IPFS



DEPLOY & RUN TRANSACTIONS

☐ Publish to IPFS

OR

At Address Load contract from Address

Transactions recorded 2

Deployed Contracts

SENDMONEYEXAMPLE AT 0XD91...3I

receiveMoney

balanceReceiv...

getBalance

Low level interactions

CALLDATA

Transact

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.13;
3
4 contract SendMoneyExample {
5     uint public balanceReceived;
6     function receiveMoney() public payable {
7         balanceReceived += msg.value;
8     }
9
10    function getBalance() public view returns(uint) {
11        return address(this).balance;
12    }
13 }
```

ContractDefinition: SendMoneyExample 1 reference(s)

[vm] from: 0x583...edd4 to: SendMoneyExample.receiveMoney() 0xd91...39138 value: 1000000000000000 wei data: 0x6d2...6ec18 logs: 0 hash: 0x421...ca075

Debug

The screenshot shows the 'DEPLOY & RUN TRANSACTIONS' window in a dark-themed IDE. The sidebar on the left contains icons for Explorer, File Explorer, Search, Run and Debug, Extensions, Testing, and Settings. The main panel has a top bar with a 'Deploy' button. Below it is a 'Publish to IPFS' checkbox. A section titled 'OR' offers two options: 'At Address' (selected) and 'Load contract from Address'. The 'Deployed Contracts' section shows a list with one entry: 'SENDMONEYEXAMPLE' at address '0XD91...3'. It has a dropdown arrow, a copy icon, and a close icon. Below the list are two function buttons: 'receiveMoney' and 'balanceReceive...'. Each button has a corresponding 'uint256' input field with a value of '1000000000000000000'. At the bottom, the 'Low level interactions' section is visible, showing a 'CALLDATA' input field and a 'Transact' button.