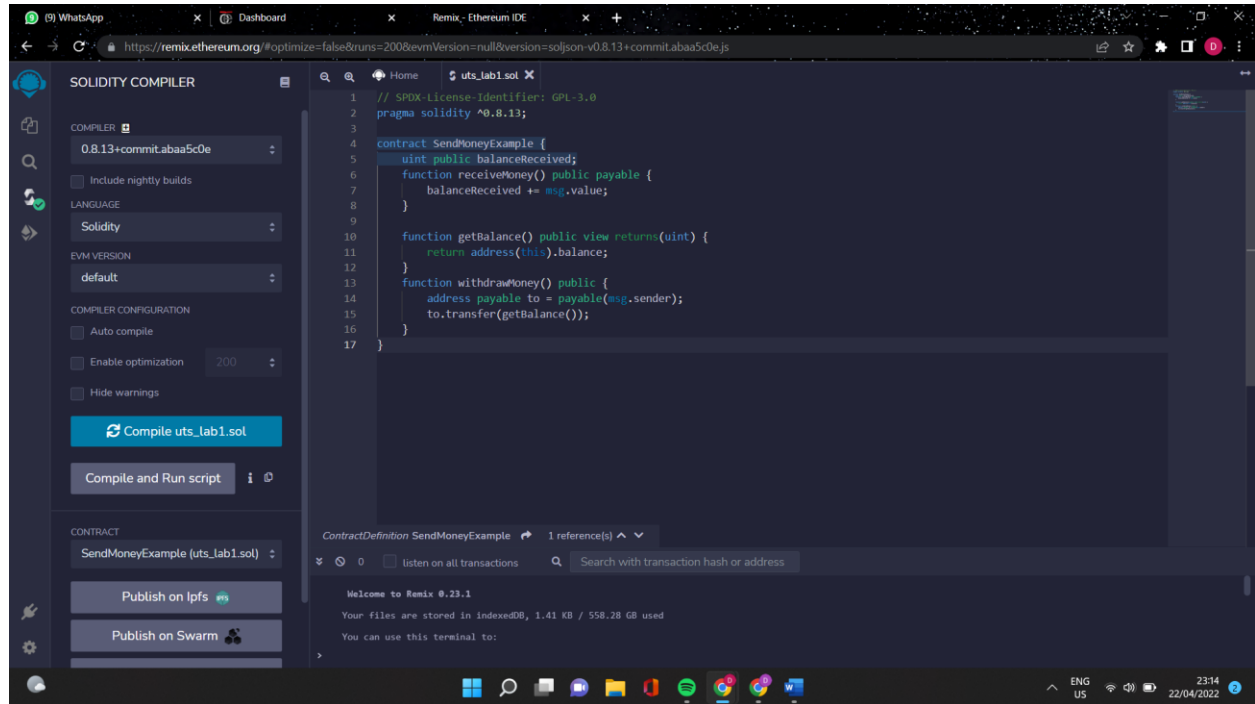


Nama : Dira Mawarni

Nim : 1103180053



1. `uint public balanceReceived` merupakan sebuah variabel penyimpanan yang bisa bersifat umum dan publik yang akan membuat fungsi pengambil secara otomatis di solidity. Sehingga developer bisa memonitor konten pada variabel tersebut.
2. `balanceReceived += msg.value` adalah objek global yang berisi tentang sekumpulan informasi transaksi yang sedang berlangsung. Value dan sender merupakan properti yang paling penting.
3. `function getBalance() public view returns(uint)` adalah fungsi yang tidak bisa mengubah penyimpanan dan dapat mengembalikan informasi.
4. `Address(this).balance` adalah variabel tipe alamat yang selalu memiliki properti yang disebut dengan balance, memberikan sejumlah ether yang disimpan pada alamat tersebut. Dapat diakses dengan mudah tetapi hanya menampilkan beberapa yang tersimpan disana. `Address(this)` mengonversi Smart Contract ke alamat.

## Deploy and Run Transactions

The screenshot shows the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' sidebar is visible. It includes an 'ENVIRONMENT' section set to 'JavaScript VM (London)', an 'ACCOUNT' section with address '0x5B3...eddC4', a 'GAS LIMIT' of '3000000', a 'VALUE' of '0' in 'Wei', and a 'CONTRACT' dropdown set to 'SendMoneyExample - uts\_lab1.sol'. A 'Deploy' button is present. Below it, 'Transactions recorded' and 'Deployed Contracts' sections are shown. The main editor displays the Solidity code for 'SendMoneyExample'.

```
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    function withdrawMoney() public {
14        address payable to = payable(msg.sender);
15        to.transfer(getBalance());
16    }
17 }
```

The bottom status bar shows a successful deployment message: '[vm] from: 0x5B3...eddC4 to: SendMoneyExample.(constructor) value: 0 wei data: 0x608...d0033 logs: 0 hash: 0xc9d...dab7e'.

This screenshot shows the same Remix IDE interface after the contract has been deployed. The 'CONTRACT' dropdown in the sidebar is still 'SendMoneyExample - uts\_lab1.sol'. Below the 'Deploy' button, there are buttons for 'receiveMoney' (red), 'withdrawMoney' (orange), 'balanceReceiv...' (blue), and 'getBalance' (blue). The 'Low level interactions' section shows 'CALLDATA' and a 'Transact' button. The main editor shows the same Solidity code. The bottom status bar displays the same deployment success message as the previous screenshot.

## Mengirim beberapa Ether ke Smart Contract dengan menukar nilai value dengan 1

The screenshot shows the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel is active, displaying the 'SendMoneyExample - uts\_lab1.sol' contract. The 'Deploy' button is highlighted. Below it, there are buttons for 'receiveMoney', 'withdrawMoney', 'balanceReceive...', and 'getBalance'. The 'Low level interactions' section shows a 'CALLDATA' field with a 'Transact' button. The main editor displays the Solidity code for the 'SendMoneyExample' contract, which includes functions for receiving money, getting balance, and withdrawing money. The bottom panel shows a successful transaction log: '[vm] from: 0x5B3...eddC4 to: SendMoneyExample.(constructor) value: 0 wei data: 0x608...d0033 logs: 0 hash: 0x6c2...bb4e8'.

The screenshot shows the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel is active, displaying the 'SendMoneyExample - uts\_lab1.sol' contract. The 'Deploy' button is highlighted. Below it, there are buttons for 'receiveMoney', 'withdrawMoney', 'balanceReceive...', and 'getBalance'. The 'Low level interactions' section shows a 'CALLDATA' field with a 'Transact' button. The main editor displays the Solidity code for the 'SendMoneyExample' contract, which includes functions for receiving money, getting balance, and withdrawing money. The bottom panel shows a successful transaction log: '[vm] from: 0x5B3...eddC4 to: SendMoneyExample.(constructor) value: 1 wei data: 0x608...d0033 logs: 0 hash: 0xd7a...8bc8f'.