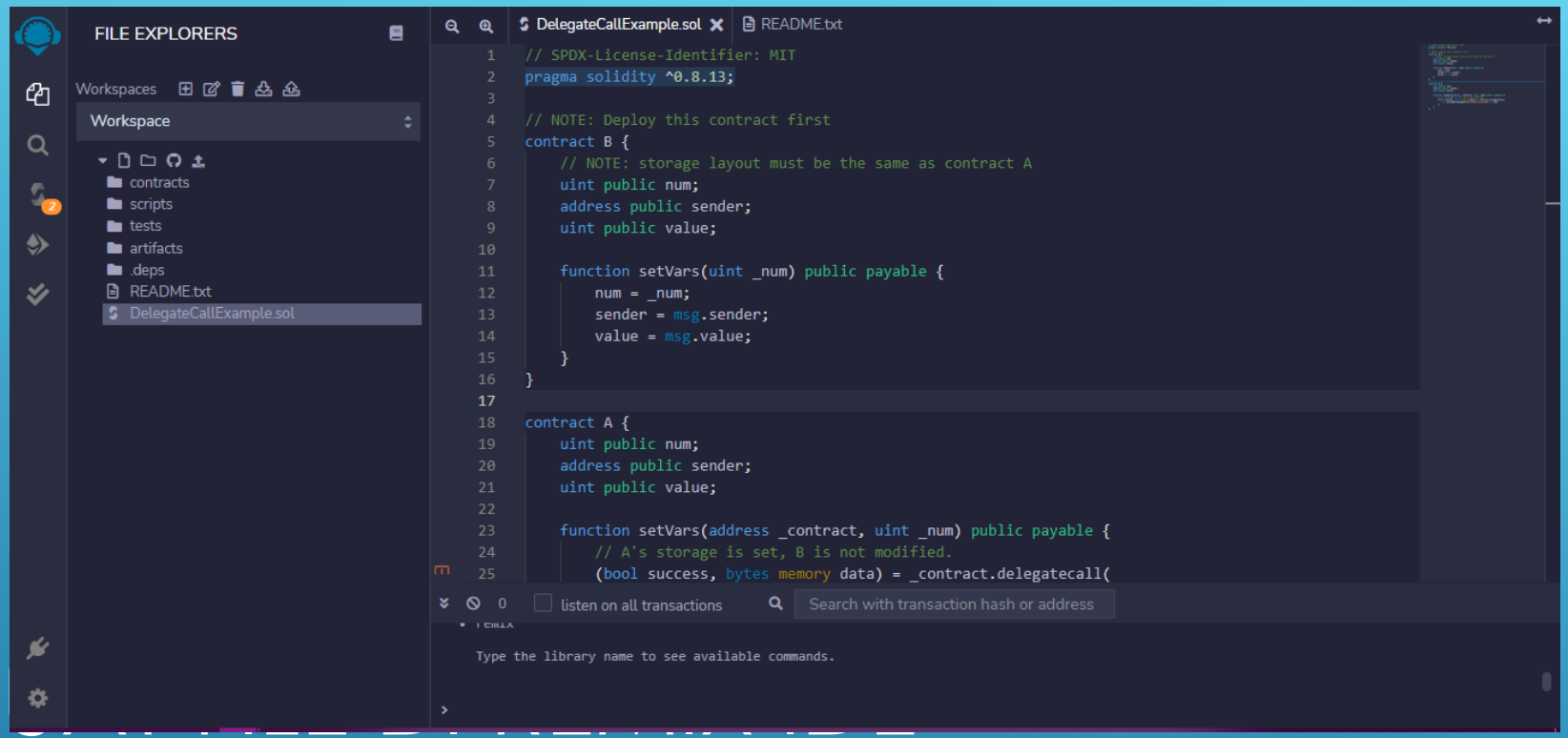
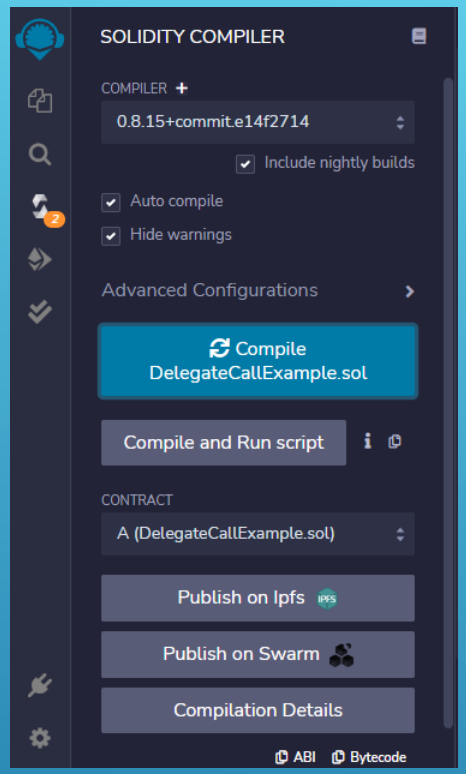
LESSON 16: HARDHAT UPGRADES

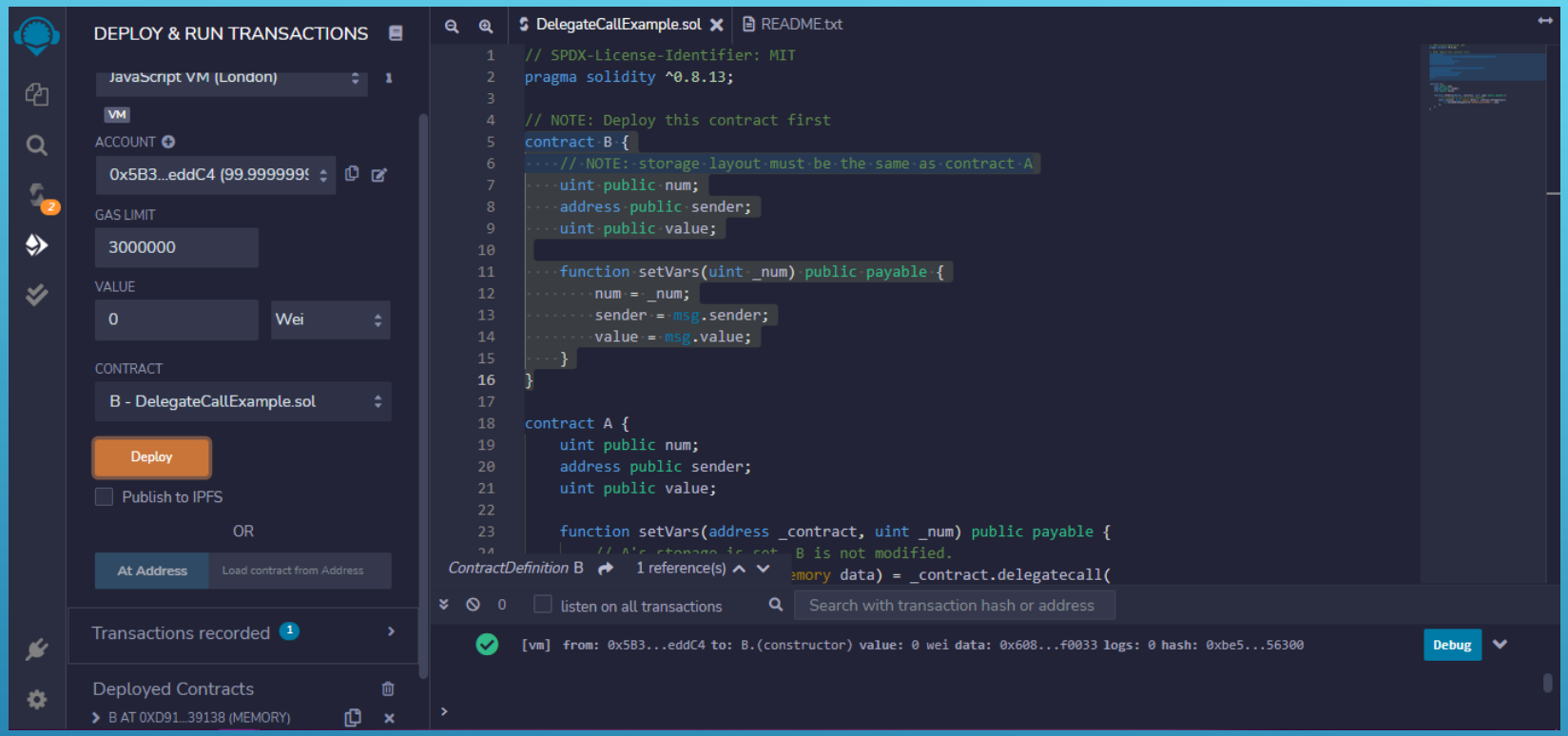
Delegate Call



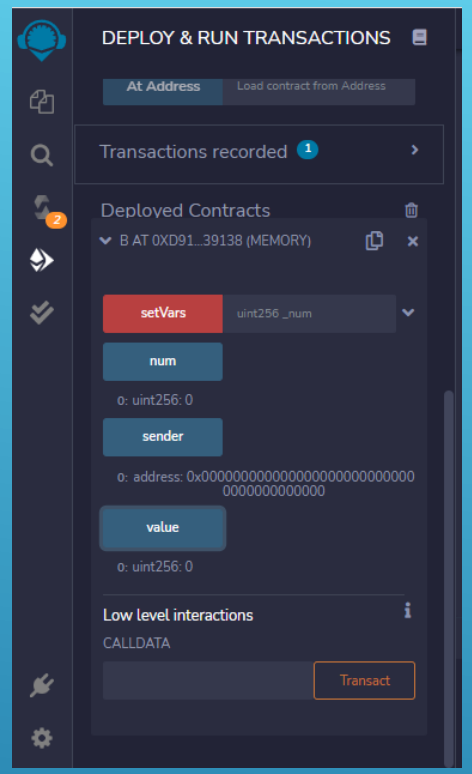
B DELEGATECALLEXAMPLE.SOL DENGAN ISI SEPERTI DI LINK HTTPS://SOLIDITY-BY-EXAMPLE.ORG/DELEGATECALL



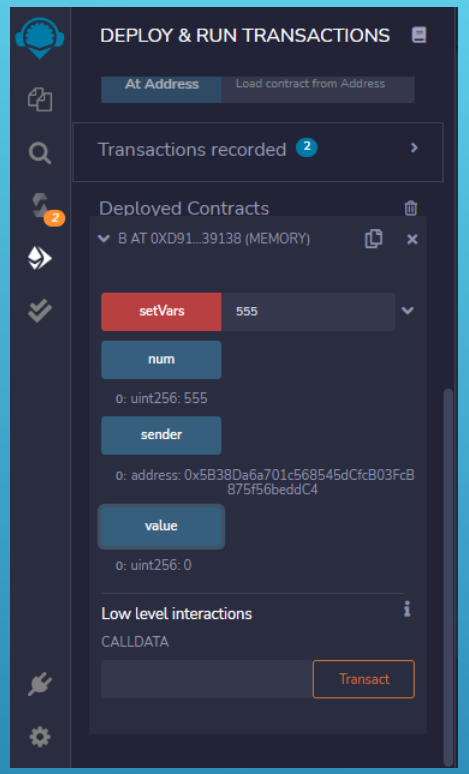
COMPILE FILE DELEGATECALLEXAMPLE.SOL DENGAN VERSI 0.8.13 KEATAS



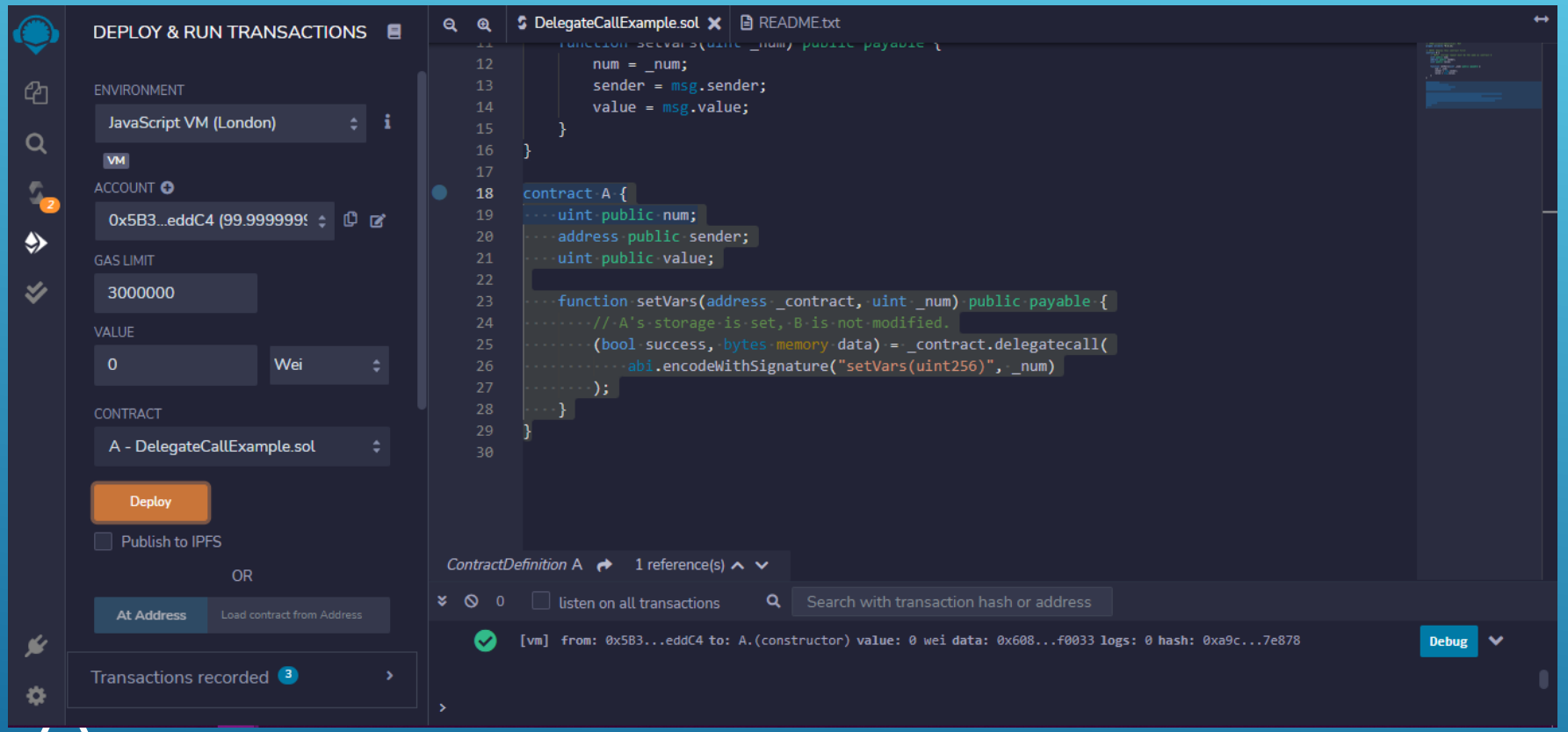
LALU KITA DEPLOY CONTRACT B, MAKA AKAN MUNCUL TAMPILAN SEPERTI INI

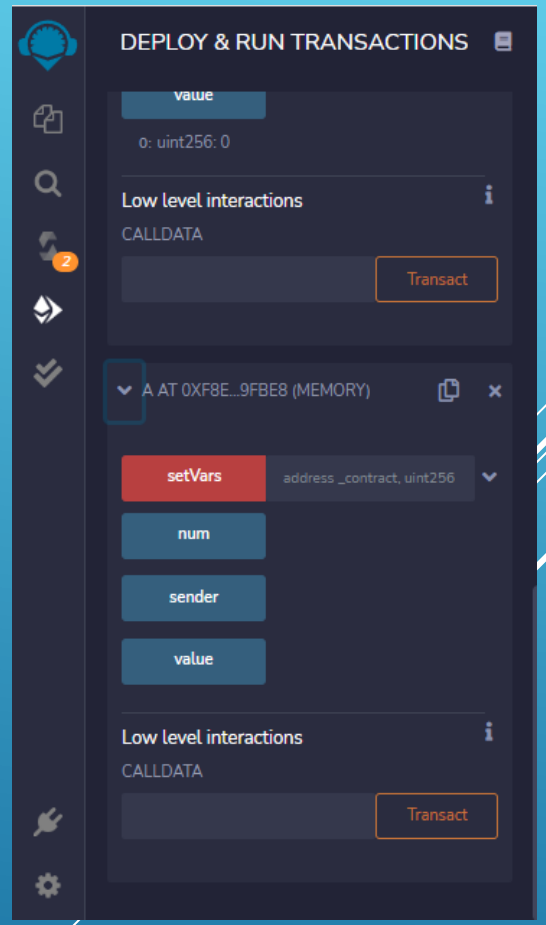


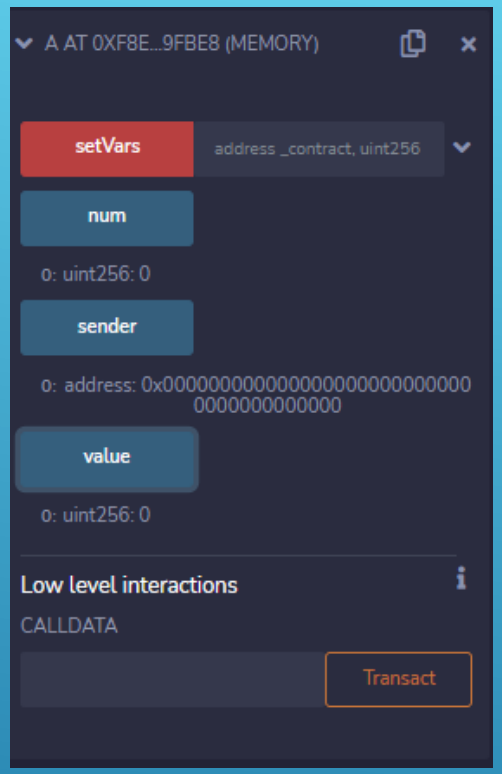
KLIK NUM, SENDER, DAN VALUE MAKA ISINYA AKAN KOSONG SEPERTI DIBAWAH INI PADA CONTRACT B.



LALU KITA SETVARS 555 MAKA IA AKAN MENGISI NUM 555 DAN AKAN MENGIRIM SENDER.

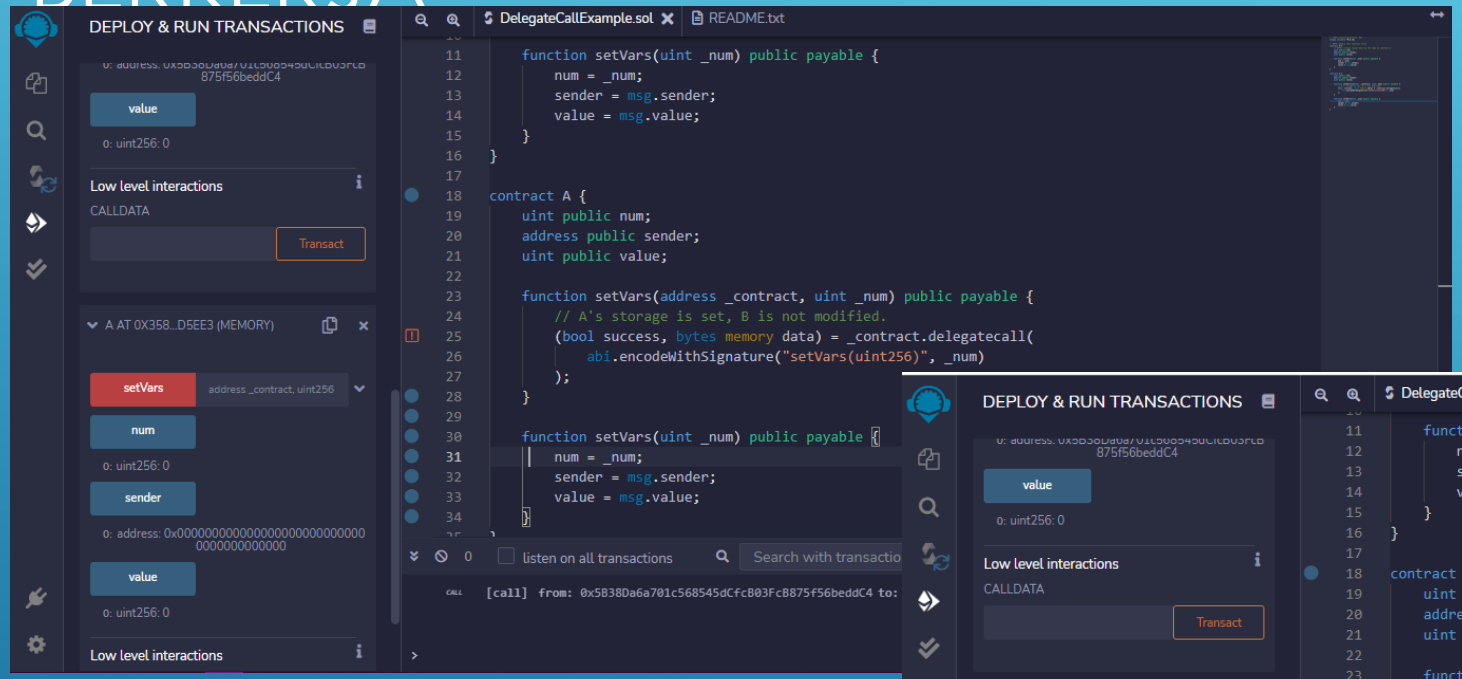

DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
JavaScript VM (London)
VM
ACCOUNT
0x5B3...eddC4 (99.999999
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
A - DelegateCallExample.sol
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 3
DelegateCallExample.sol
README.txt
num = _num;
sender = msg.sender;
value = msg.value;
contract A {
uint public num;
address public sender;
uint public value;
function setVars(address _contract, uint _num) public payable {
//A's storage is set, B is not modified.
(bool success, bytes memory data) = _contract.delegatecall(
abi.encodeWithSignature("setVars(uint256)", _num)
);
}
}
ContractDefinition A
1 reference(s)
listen on all transactions
Search with transaction hash or address
[vm] from: 0x5B3...eddC4 to: A.(constructor) value: 0 wei data: 0x608...f0033 logs: 0 hash: 0xa9c...7e878
Debug

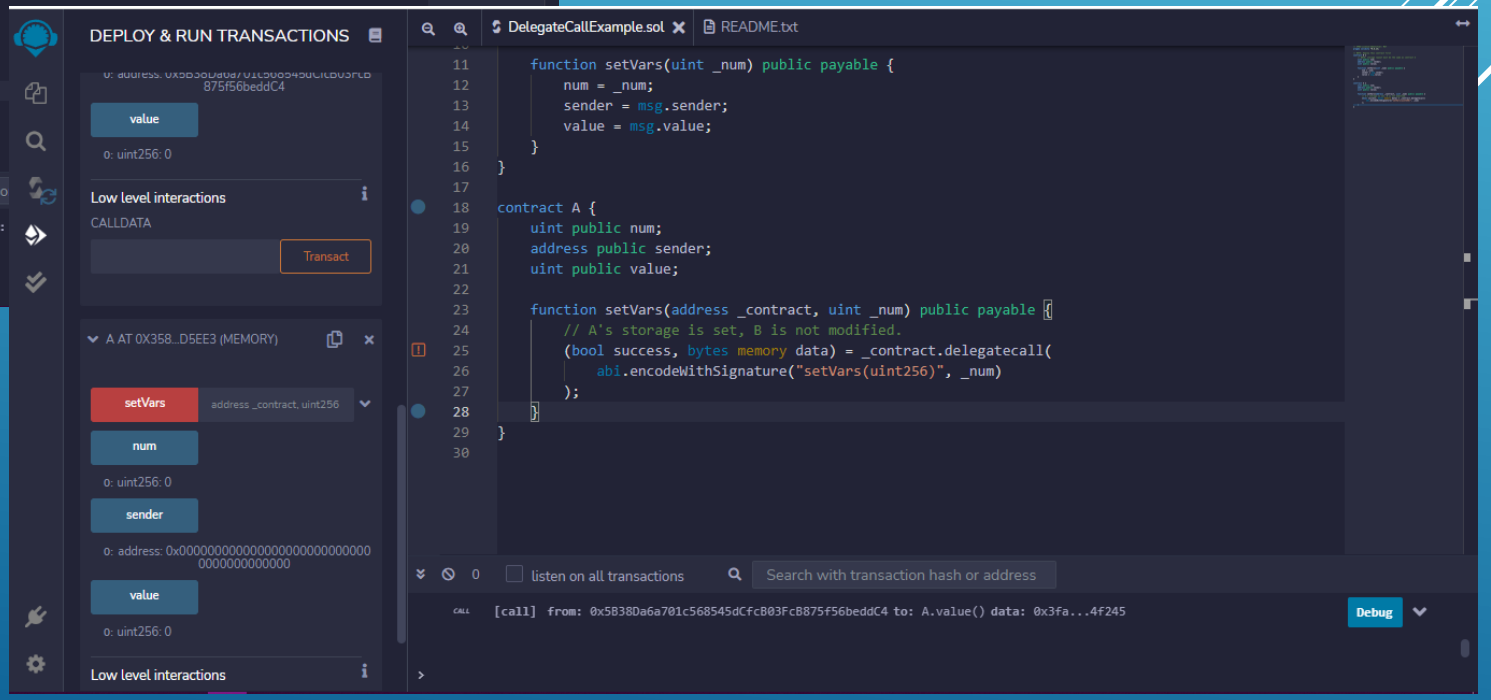

DEPLOY & RUN TRANSACTIONS
value
0: uint256: 0
Low level interactions
CALLDATA
Transact
A AT 0XF8E...9FBE8 (MEMORY)
setVars
address _contract, uint256
num
sender
value
Low level interactions
CALLDATA
Transact

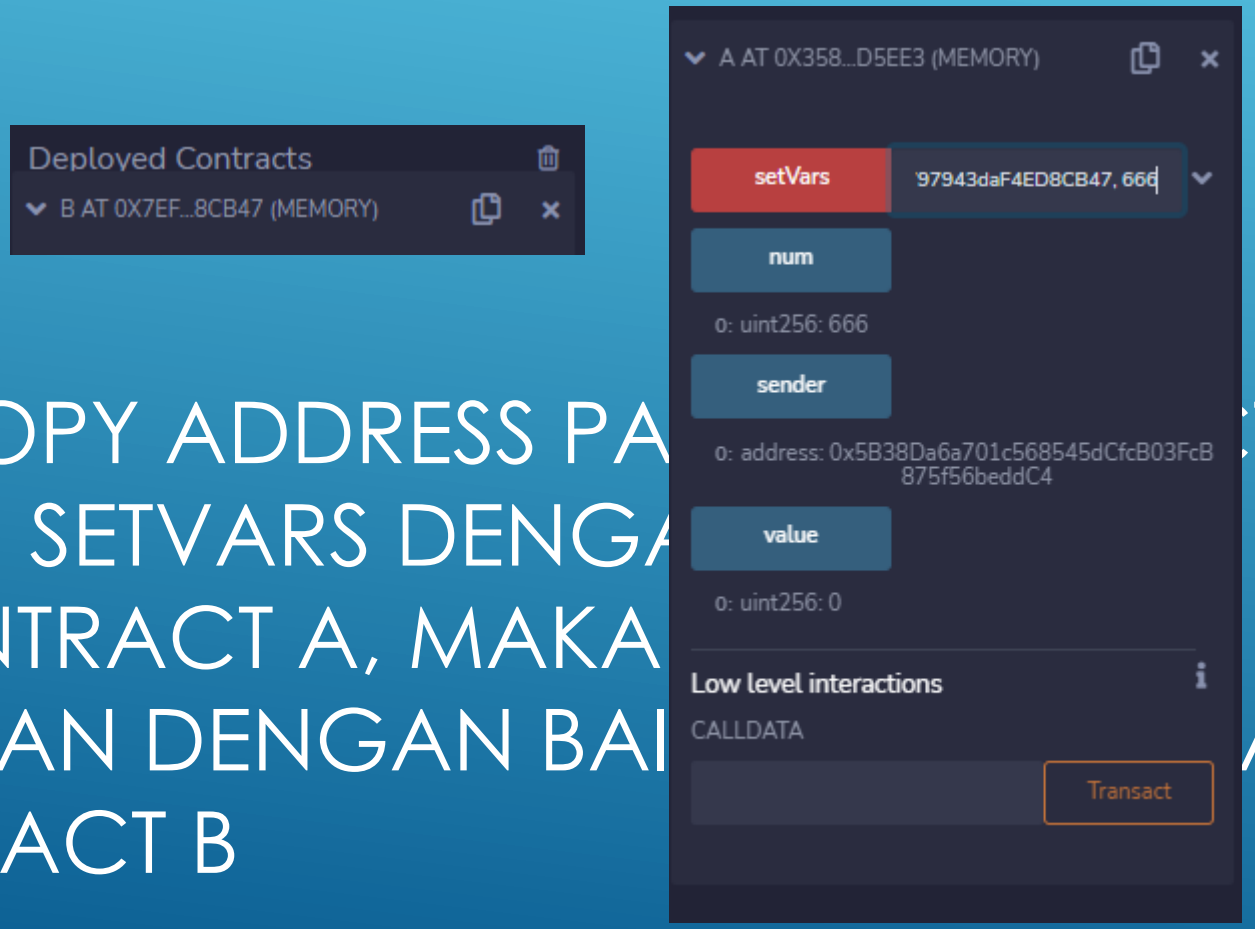


KLIK NUM, SENDER, DAN VALUE MAKA ISINYA AKAN KOSONG SEPERTI DIBAWAH INI, PADA CONTRACT A.

LALU KITA PINJAM FUNCTION DARI CONTRACT B RUN SEKALI, LALU HAPUS KEMBALI FUNCTION TERSEBUT, DAN INI ADALAH PERUMPAMAAN SISTEM DELEGATE BERKERJA







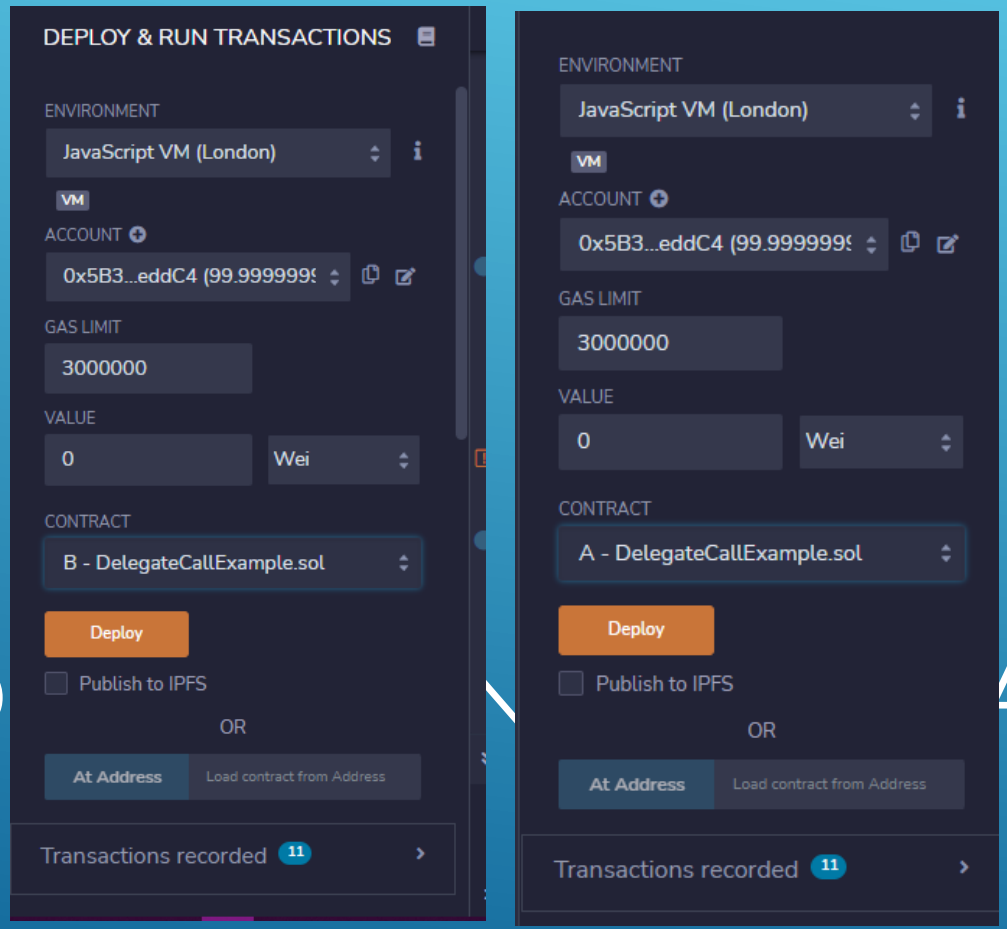
KITA COPY ADDRESS PA T B LALU ISI SETVARS DENGA MAU DI CONTRACT A, MAKA BERJALAN DENGAN BAI A CONTRACT B

JIKA KITA MENGUBAH NAMA VARIABLE PADA CONTRACT A, MAKA TIDAK AKAN BERPENGARUH DE ION YANG DI ON DI CO AH

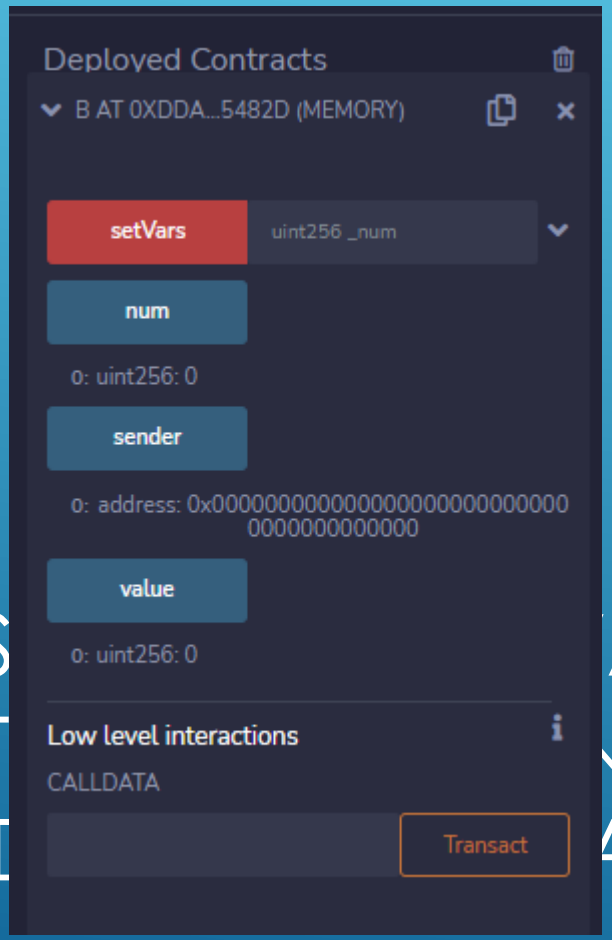
DelegateCallExample.sol | README.txt

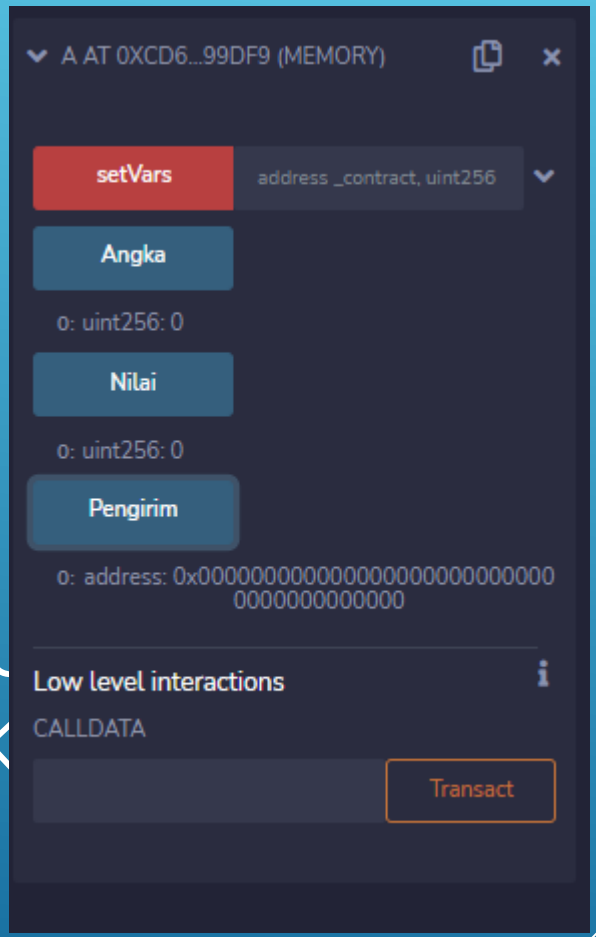
```
11      function setVars(uint _num) public payable {
12          num = _num;
13          sender = msg.sender;
14          value = msg.value;
15      }
16  }
17
18  contract A {
19      uint public Angka;
20      address public Pengirim;
21      uint public Nilai;
22
23      function setVars(address _contract, uint _num) public payable {
24          // A's storage is set, B is not modified.
25          (bool success, bytes memory data) = _contract.delegatecall(
26              abi.encodeWithSignature("setVars(uint256)", _num)
27          );
28      }
29  }
30
```

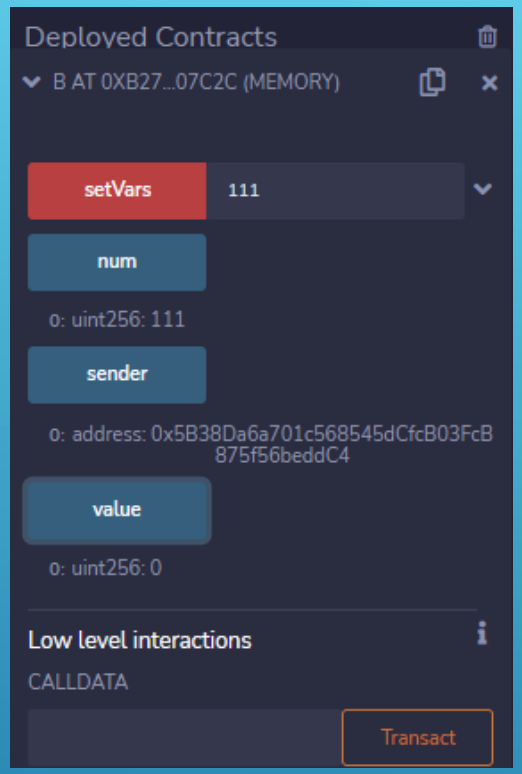
LALU KITA D ACT A DAN B



KLIK NUM, S ALU CONTRACT NGK DAN NILAI D A.

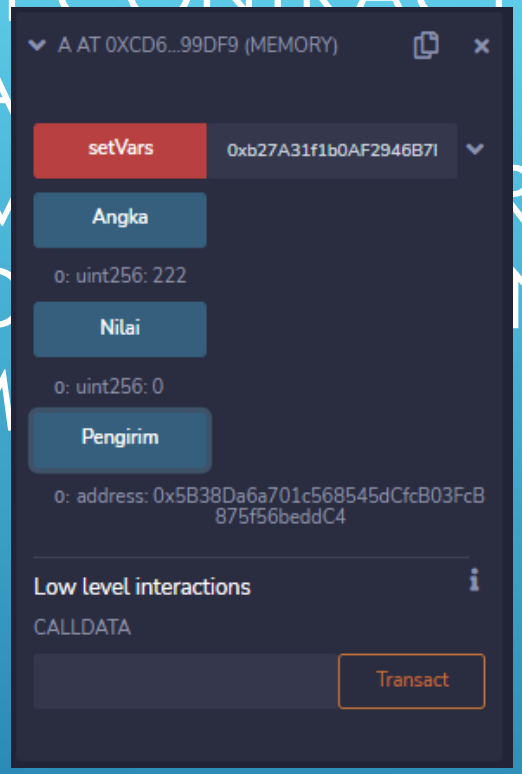


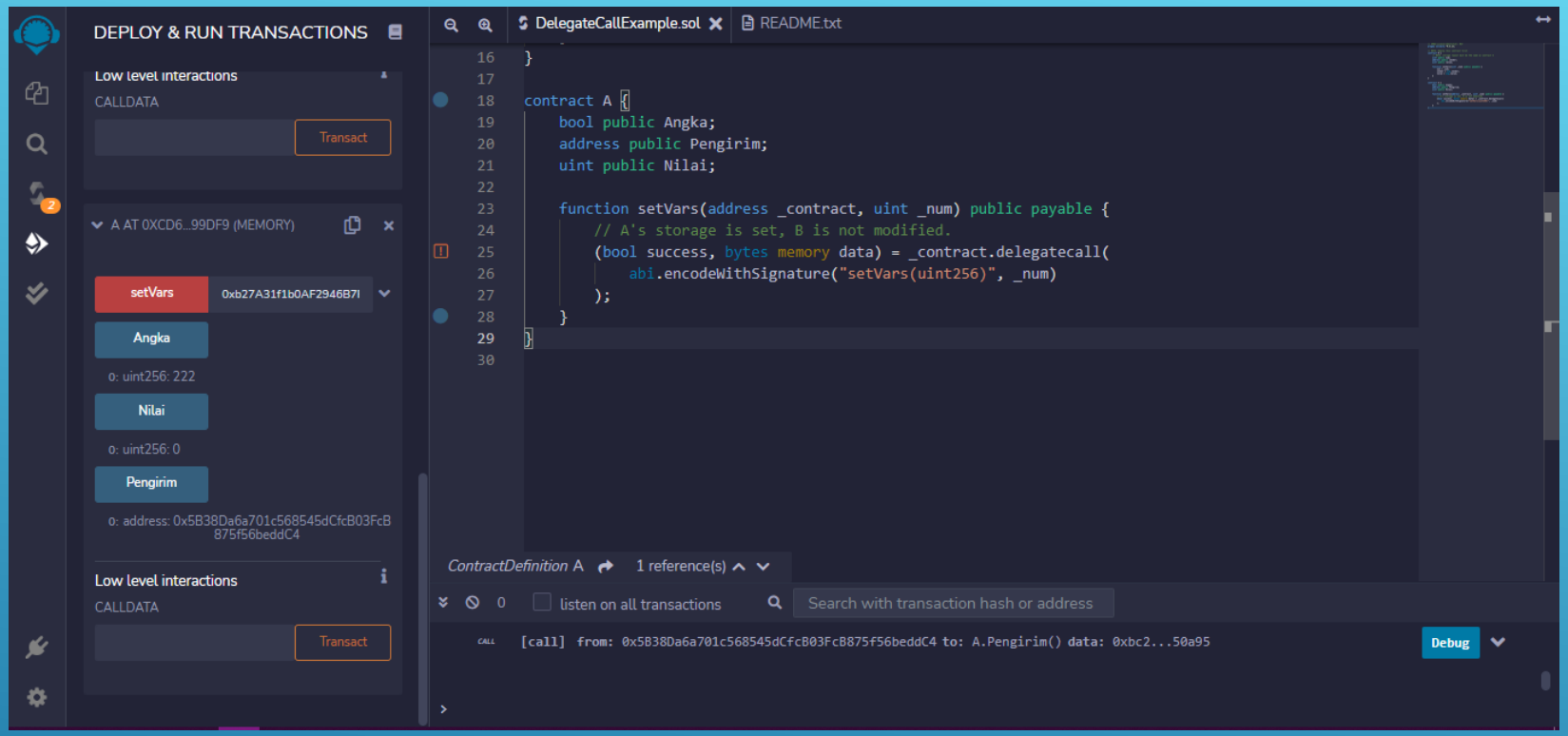




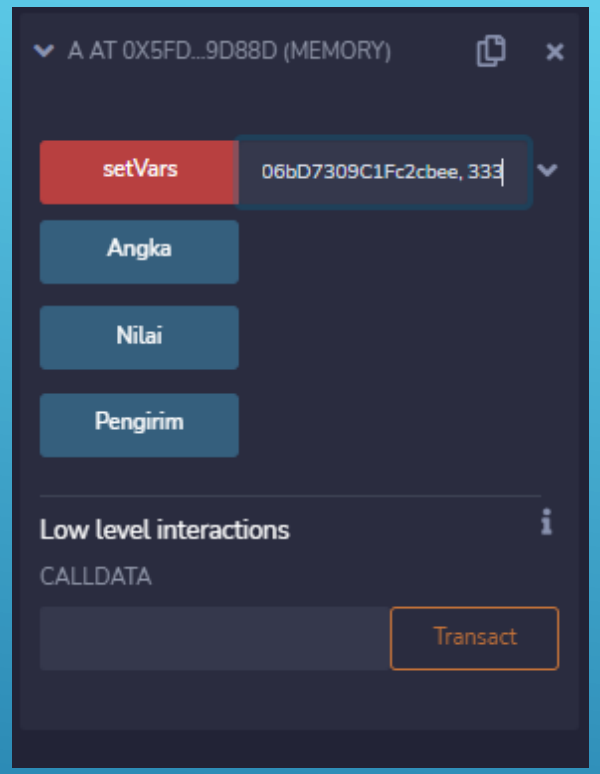
KITA SETVARS LAGI PADA CONTRACT B

COPY ALAMAT CONTRACT B DAN MASUKAN DI SETVARS PADA A BESERTA NILAI YANG DIINGINKAN, MAKA CONTRACT A AKAN BEKERJA PERSIS SAMA DENGAN CONTRACT B, WALAUPUN DENGAN NAMA YANG BERBEDA



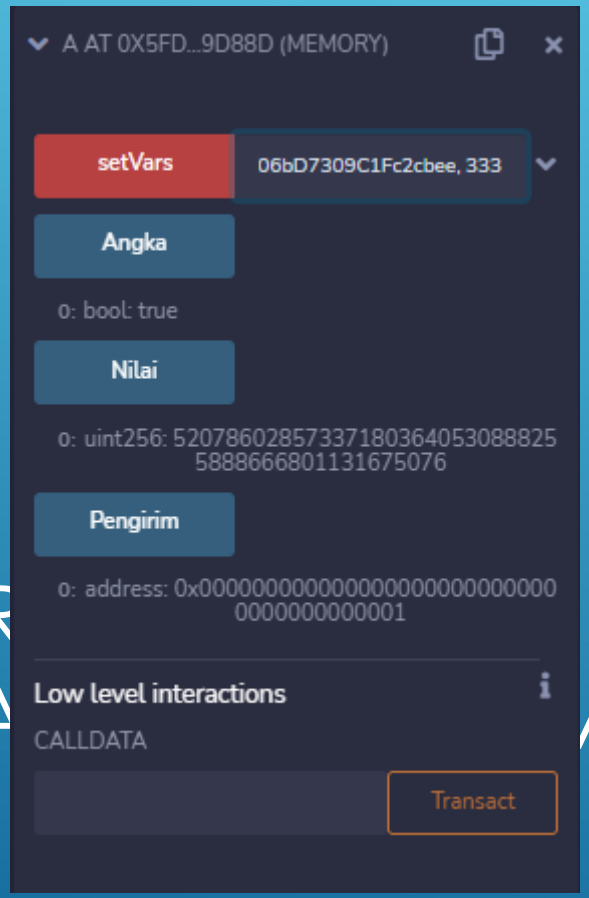


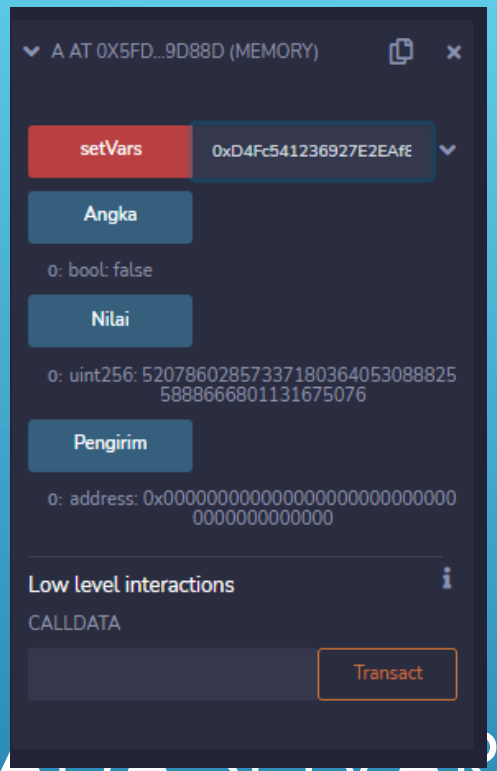
SEKARANG COBA KITA UBAH TIPE VARIABLE ANGKA PADA CONTRACT A JADI BOOLEAN



KITA DEPLOY KEMBALI CONTRACT A DAN ISI KEMBALI ADDRESS CONTRACT B SERTA SETVARS YANG KITA INGINKAN

KLIK ANGKA, PENGIR DI CONTRACT A, MAKA AN TRUE KARENA BOOLEAN





JIKA KITA ISI 0 PADA SETVARSNYA, MAKA AKAN BERNILAI FALSE KARENA PADA CONTRACT B VALUENYA ADALAH INTEGER/ANGKA JADI SELAIN 0 AKAN BERNILAI TRUE, SEDANGKAN 0 FALSE