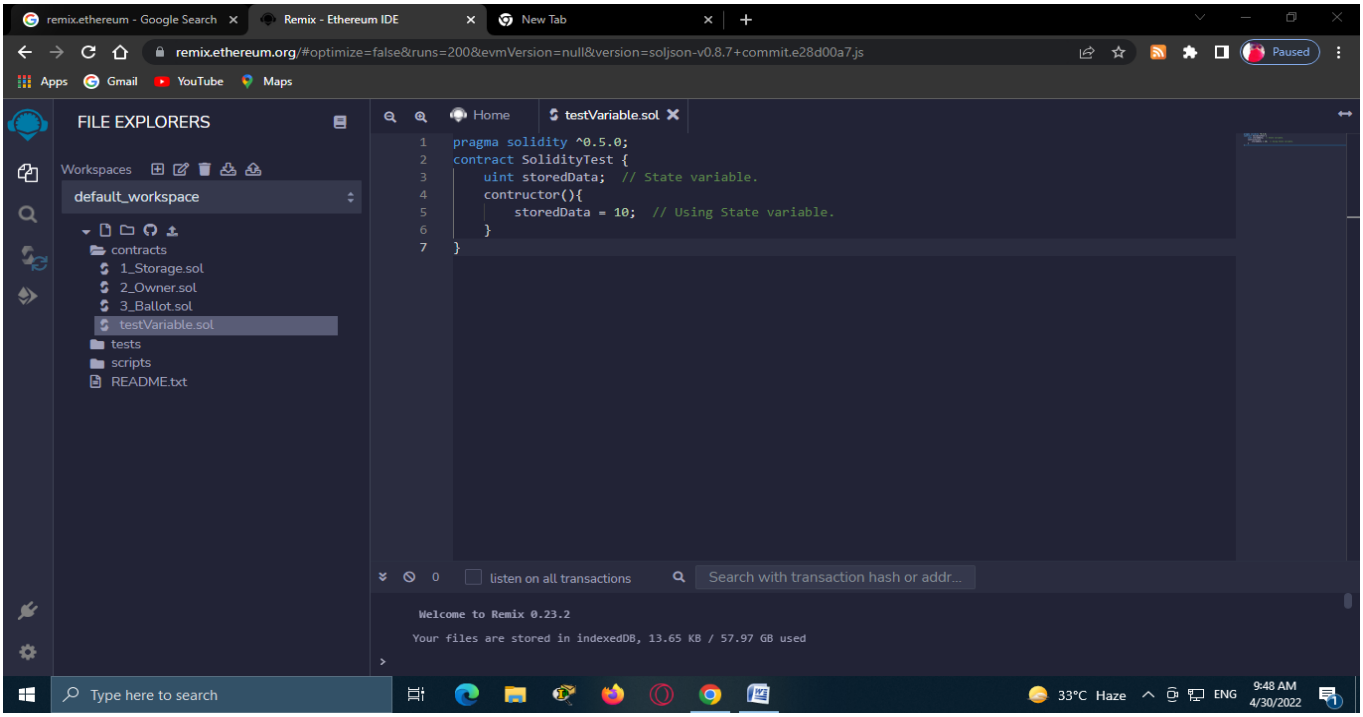
Practical 7

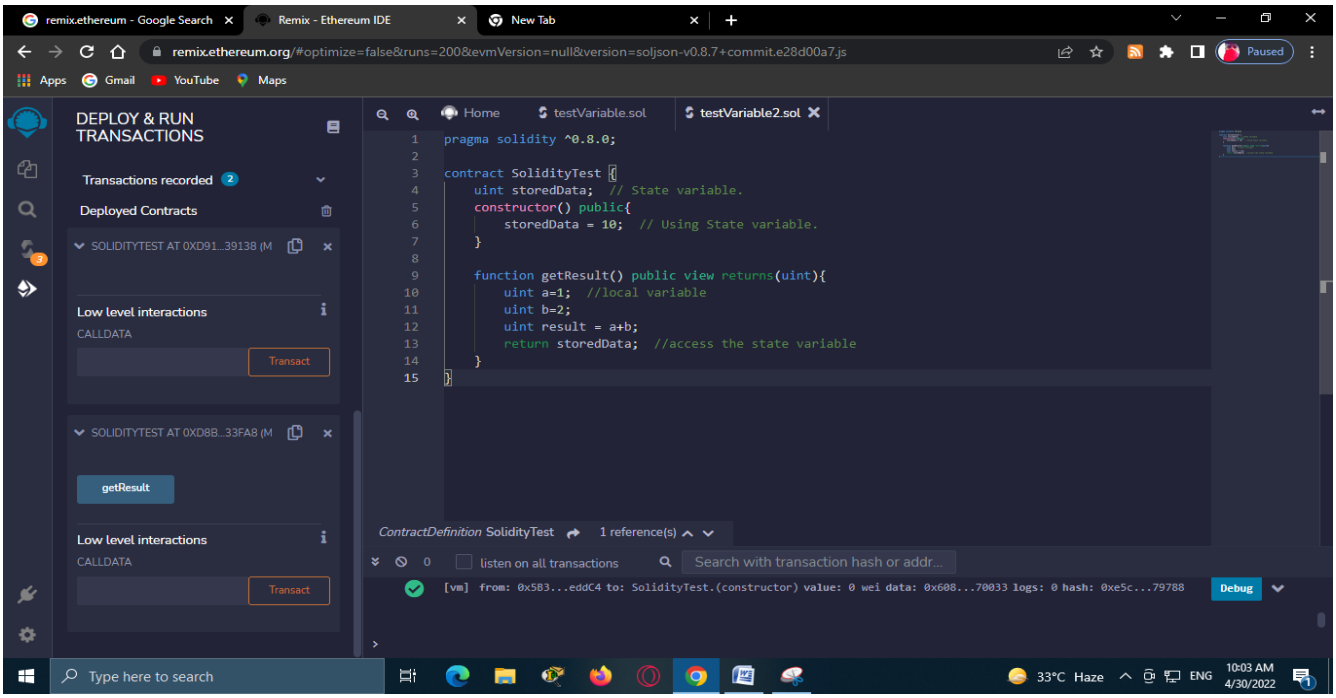
Implement and demonstrate the use of following in solidity

Variables

testVariable



testVariable2



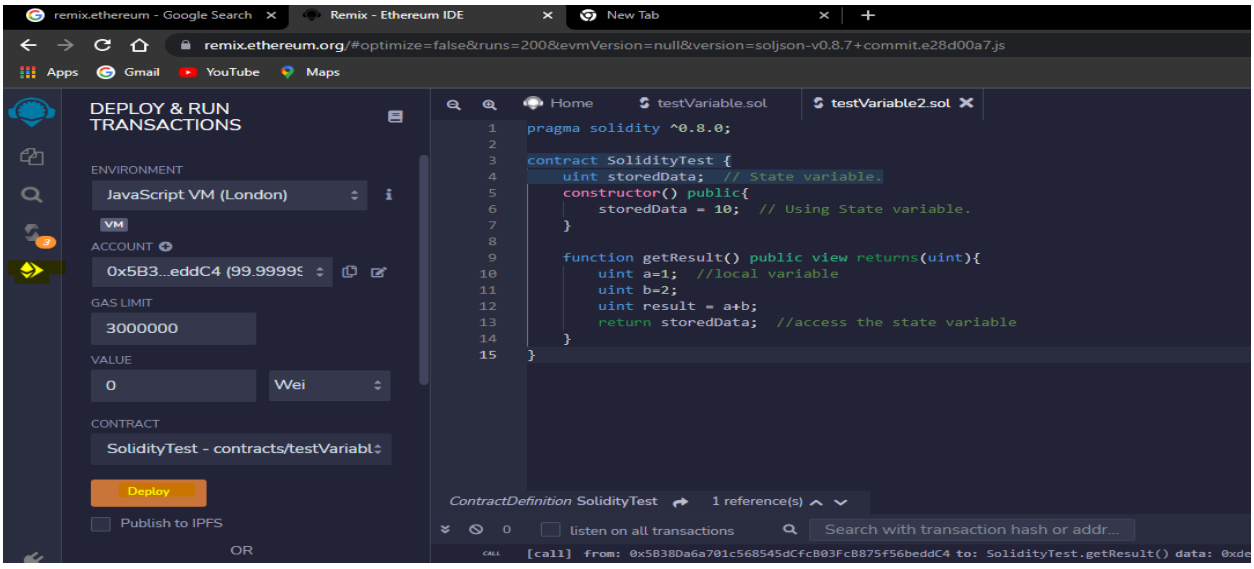
Code

```
pragma solidity ^0.8.0;

contract SolidityTest {
    uint storedData;  // State variable.
    constructor() public{
        storedData = 10;  // Using State variable.
    }

    function getResult() public view returns(uint){
        uint a=1;  //local variable
        uint b=2;
        uint result = a+b;
        return storedData;  //access the state variable
    }
}
```

Steps to deploy

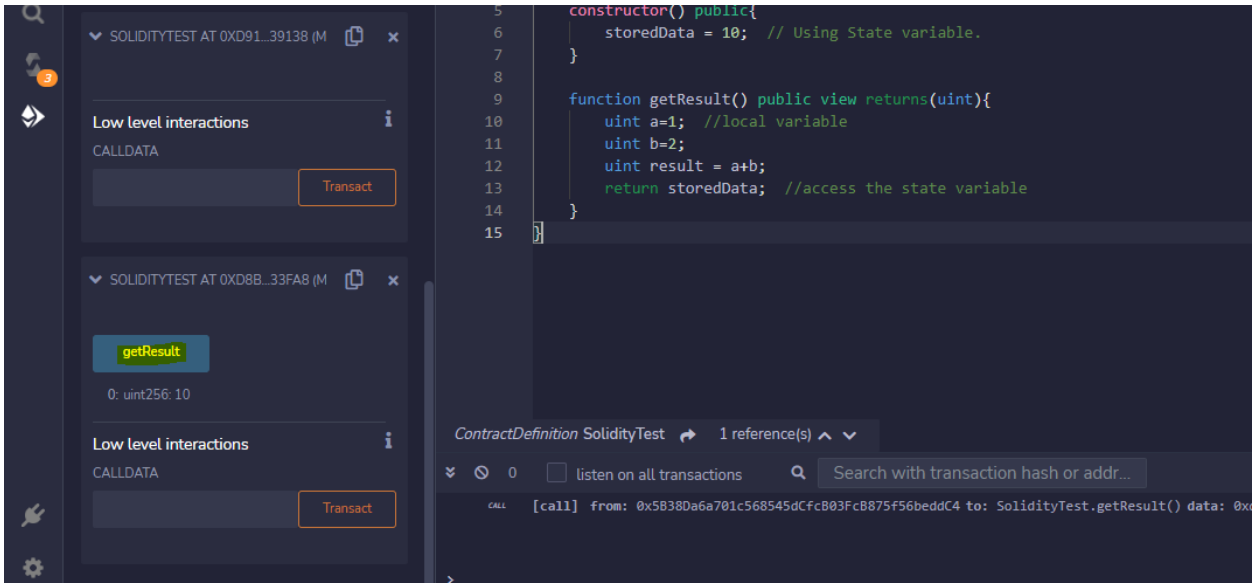
Click to the arrow symbol and check the name to the contract.

And then click on deploy.



Output:

Scroll down the left hand side. Below the deploy button, you will get the button getResult. Click on it and you will the output of your program.



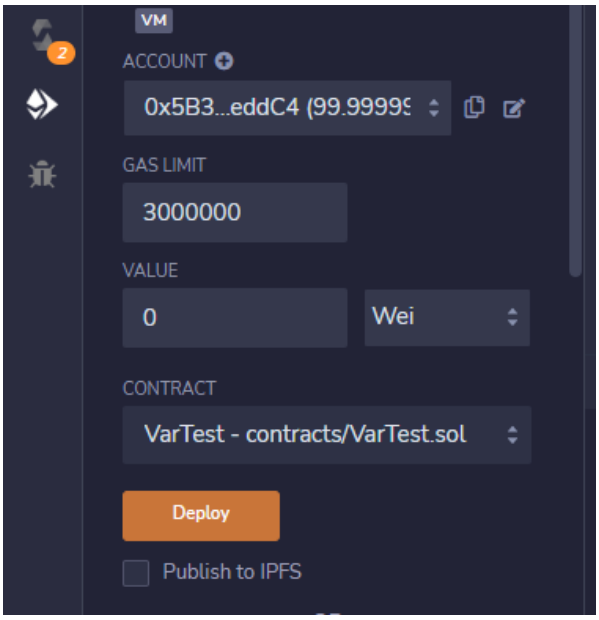
Note : Every time you make the changes in the program, you have to first deploy and then check for the output.

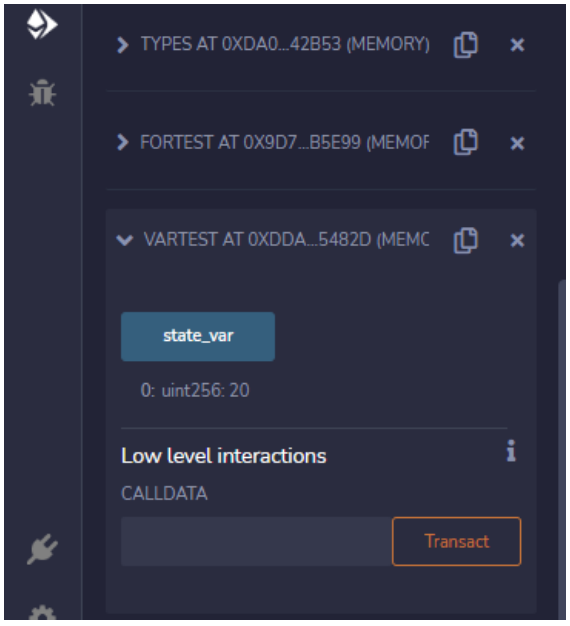
Variable Test : State variable

varTest

```
pragma solidity ^0.8.0;

contract VarTest {
    uint public state_var;  // State variable.
    constructor() public{
        state_var = 20;  // Using State variable.
    }
}
```

Deploy :



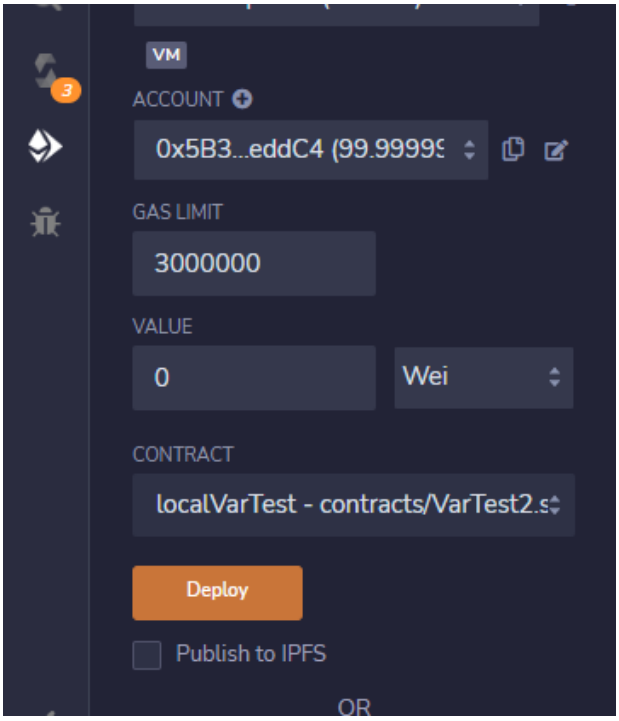


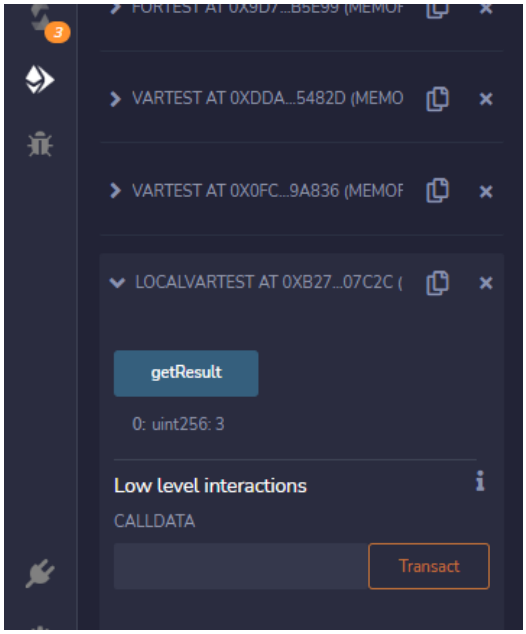
Output :

VarTest2 : Local variable

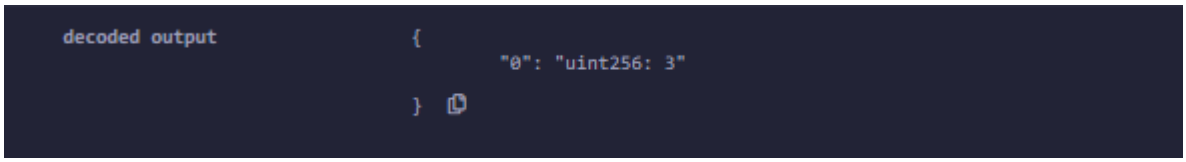
```
pragma solidity ^0.8.0;

contract localVarTest {
    uint storedData;  // State variable.
    constructor() public{
        storedData = 10;  // Using State variable.
    }

    function getResult() public view returns(uint){
        uint a=1;  //local variable
        uint b=2;
        uint result = a+b;
        return result;  //access the local variable
    }
}
```

Deploy:





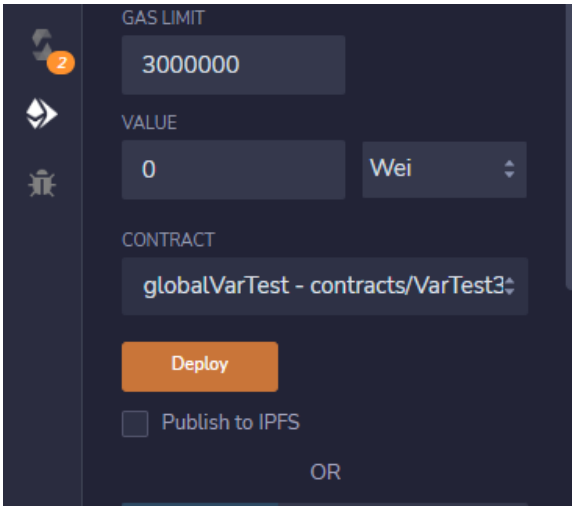
Output:

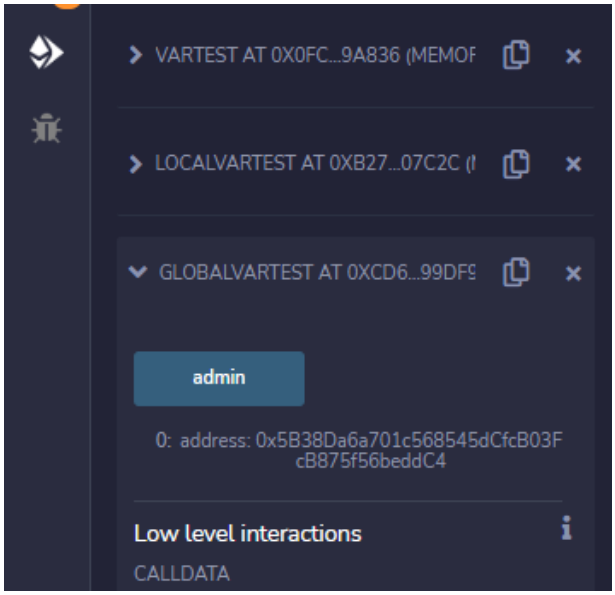


Variable test : Global variable

```
pragma solidity ^0.8.0;

contract globalVarTest {

    address public admin;

    constructor() public{
        admin = msg.sender;
    }
}
```

Deploy





Output: