

Lab: Working with Variables

Prerequisites

1. Chrome or Firefox browser.
2. An Internet connection

This Document has been updated in March 2020 to reflect Solidity 0.6 changes

Open Remix with the following Smart Contract:

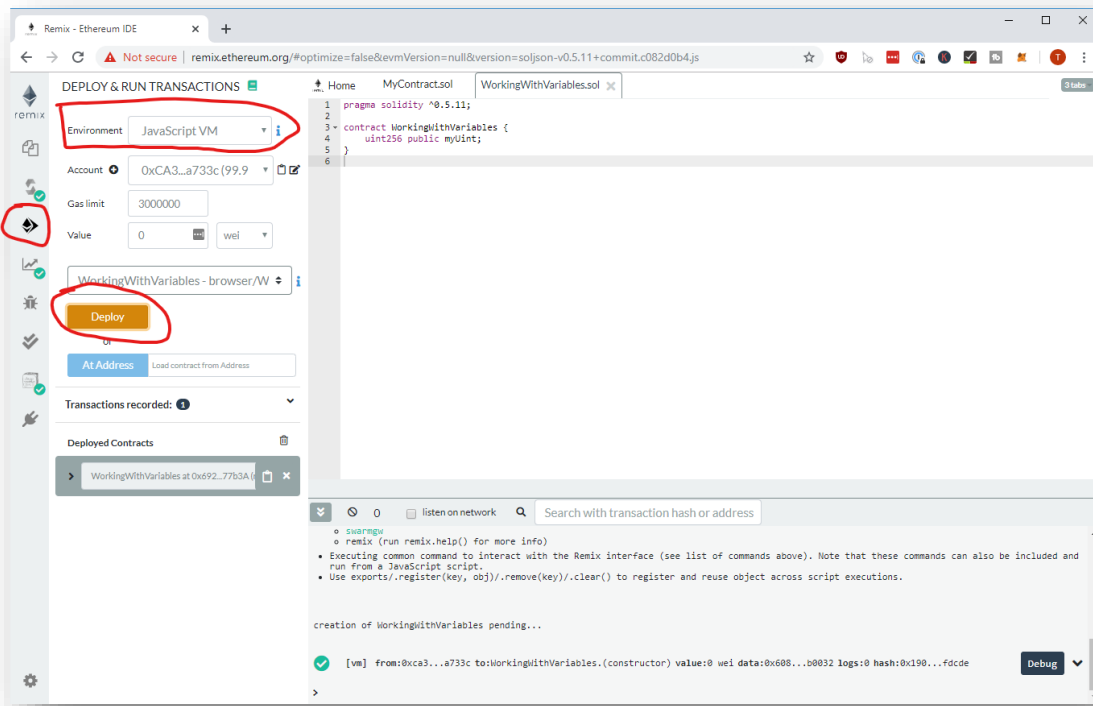
```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    uint256 public myUint;
}
```

Step by Step Instruction

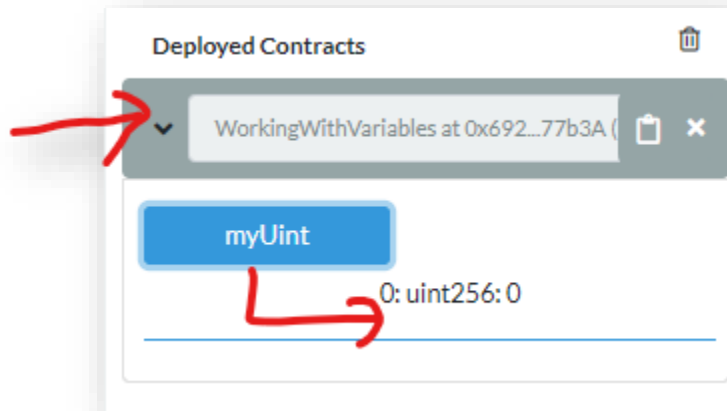
Deploy the Smart Contract in the JavaScript VM

Open the “Deploy and Run Transactions” view in Remix with the smart contract



Interact with the Smart Contract

Now we are reading the myUint, although it has not been initialized:

*We add a Setter-Function*

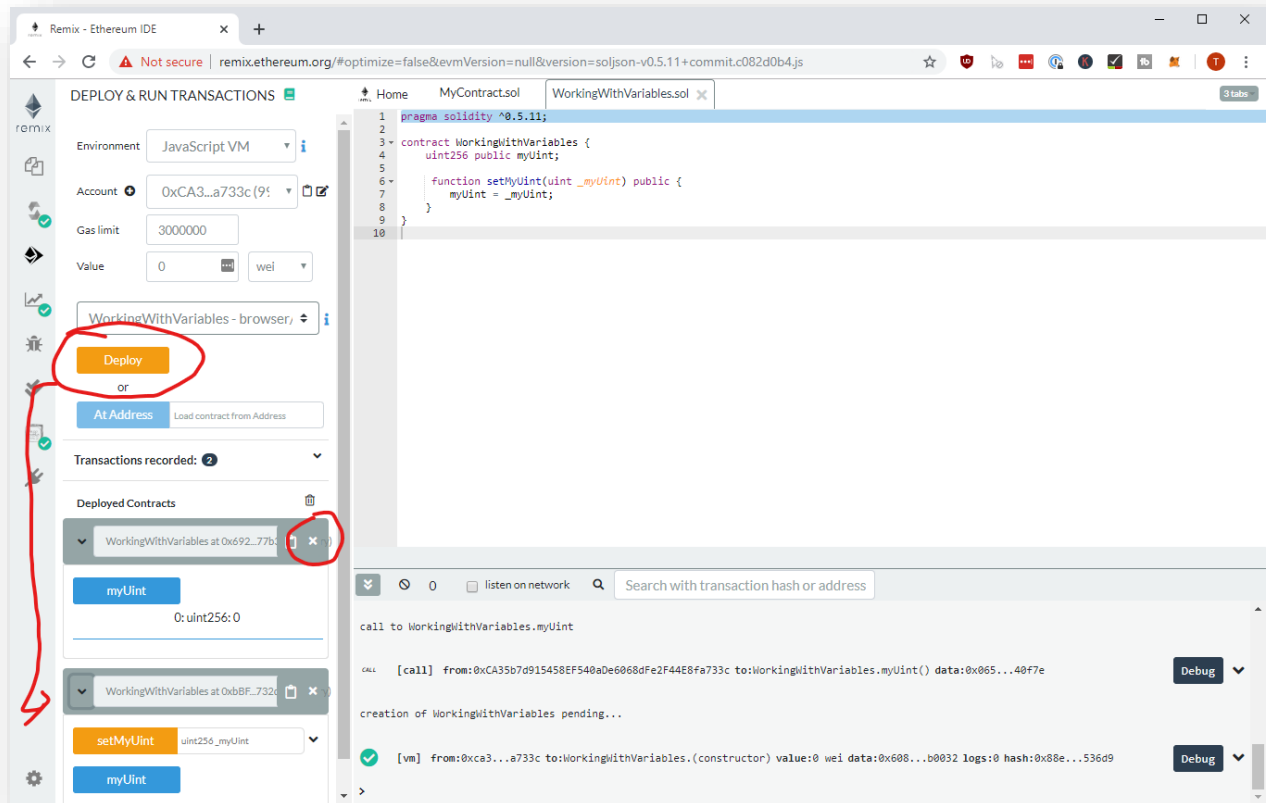
```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    uint256 public myUint;

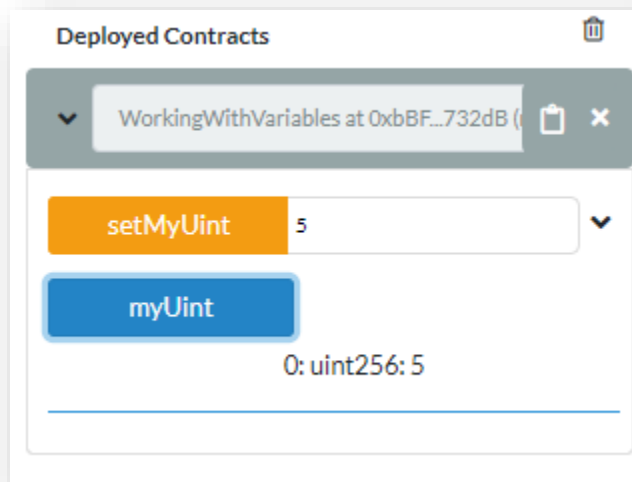
    function setMyUint(uint _myUint) public {
        myUint = _myUint;
    }
}
```

Deploy a new version of the smart contract

First you need to deploy a new version of the smart contract and you can close the previous Instance:



Set the Integer and get the result



Add a Boolean and Interact with it

Don't forget to deploy a new instance before it appears:

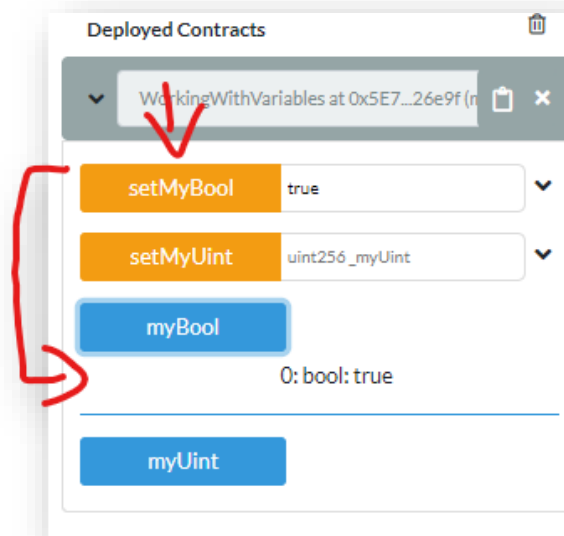
```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    uint256 public myUint;

    function setMyUint(uint _myUint) public {
        myUint = _myUint;
    }

    bool public myBool;

    function setMyBool(bool _myBool) public {
        myBool = _myBool;
    }
}
```



Add a uint8 and increment/decrement functions

Uint8 ranges from 0 to 255. We add two functions to increment and decrement the variable.

```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    uint256 public myUint;

    function setMyUint(uint _myUint) public {
        myUint = _myUint;
    }

    bool public myBool;

    function setMyBool(bool _myBool) public {
        myBool = _myBool;
    }

    uint8 public myUint8;

    function incrementUint() public {
        myUint8++;
    }

    function decrementUint() public {
        myUint8--;
    }
}
```

The screenshot shows the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel is active. It features a 'At Address' button, a 'Load contract from Address' input, and a 'Transactions recorded: 9' dropdown. Below these are 'Deployed Contracts' and a list of buttons for interacting with the 'C01HelloWorld' contract at address '0xef5...46e41'. The 'incrementUint' button is highlighted with a red circle. The main editor shows the Solidity code for 'MyContract.sol' and 'WorkingWithVariables.sol'. The bottom panel displays the transaction history, including a successful transaction for 'C01HelloWorld.incrementUint()' and a call to 'C01HelloWorld.myUint8'.

Observe Overflow and Underflow

With Solidity you have to be careful about overflows and underflows. There are no warnings!

Decrement the uint to "-1" and observe it automatically rolls over to 255:

The screenshot shows the Remix Ethereum IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel is active, displaying a list of deployed contracts. The contract 'C01HelloWorld' is selected, and its functions are listed. The 'decrementUInt' function is highlighted with a red circle and a red 'X' mark. Below the function list, the 'myUInt8' variable is shown with its current value of 255. On the right, the Solidity code for the contract is displayed, showing the 'decrementUInt' function and the 'myUInt8' variable. The console at the bottom shows the transaction 'transact to C01HelloWorld.decrementUInt pending ...' and the result '[vm] from:0xca3...a733c to:C01HelloWorld.decrementUInt() 0xef5...46e41 value:0 wei data:0 hash:0x3c1...0e2ac'. Below this, a call to 'C01HelloWorld.myUInt8' is shown, and the console output indicates the value of 'myUInt8' is 255.

Add a Variable of the type "Address"

```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    //[...]

    Address public myAddress;

    function setAddress(address _address) public {
        myAddress = _address;
    }

    function getBalanceOfAccount() public view returns(uint) {
        return myAddress.balance;
    }
}
```



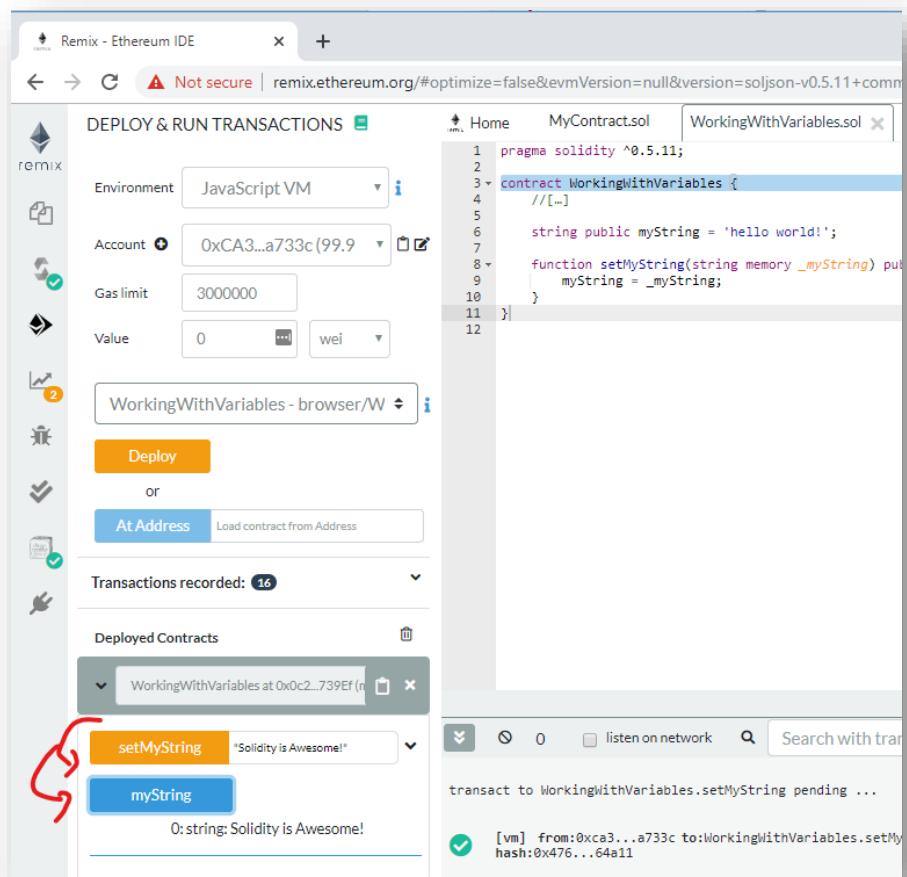

Add a String

```
pragma solidity >=0.5.11 <0.7.0;

contract WorkingWithVariables {
    //[...]

    string public myString = 'hello world!';

    function setMyString(string memory _myString) public {
        myString = _myString;
    }
}
```



Congratulations, LAB is completed



From the Course “Ethereum Blockchain Developer – Build Projects in Solidity”



FULL COURSE:

<https://www.udemy.com/course/blockchain-developer/?referralCode=E8611DF99D7E491DFD96>