Lab: Starting, Pausing and Stopping Smart Contracts

Prerequisites

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

The Solidity code has been updated to be compatible to Solidtiy 0.6

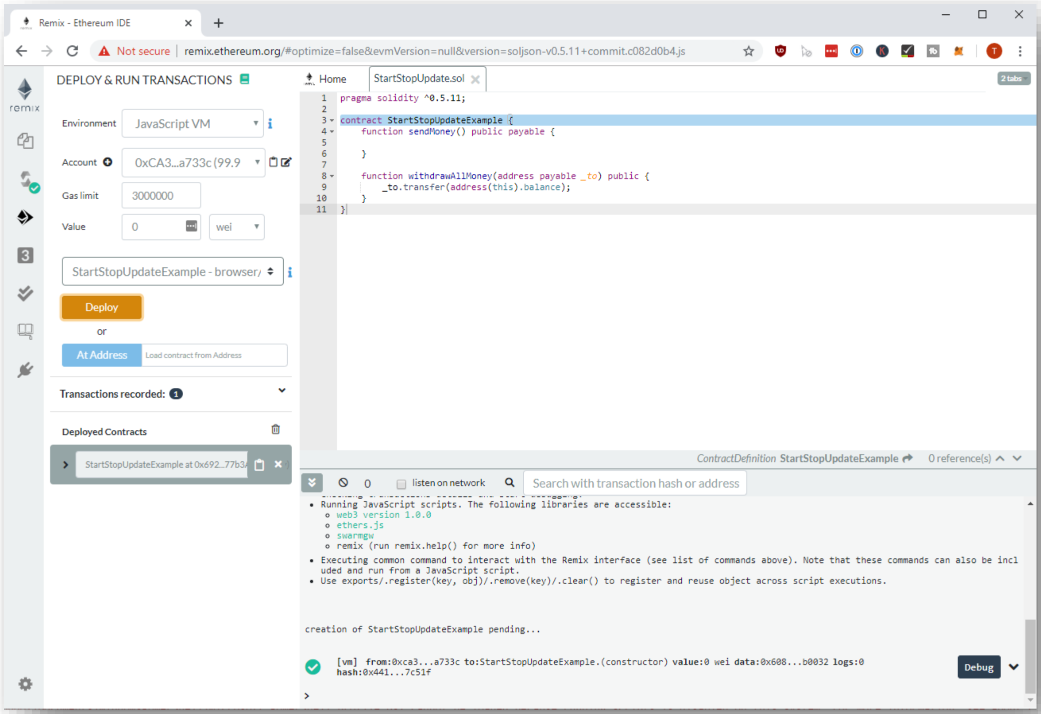
3. Open Remix with the following Smart Contract:

```
pragma solidity >=0.5.11 <0.7.0;

contract StartStopUpdateExample {
    function sendMoney() public payable {

    }

    function withdrawAllMoney(address payable _to) public {
        _to.transfer(address(this).balance);
    }
}
```

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



remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js

DEPLOY & RUN TRANSACTIONS

Environment: JavaScript VM
Account: 0xCA3...a733c (99.9
Gas limit: 3000000
Value: 0 wei

StartStopUpdateExample - browser/

Transactions recorded: 1

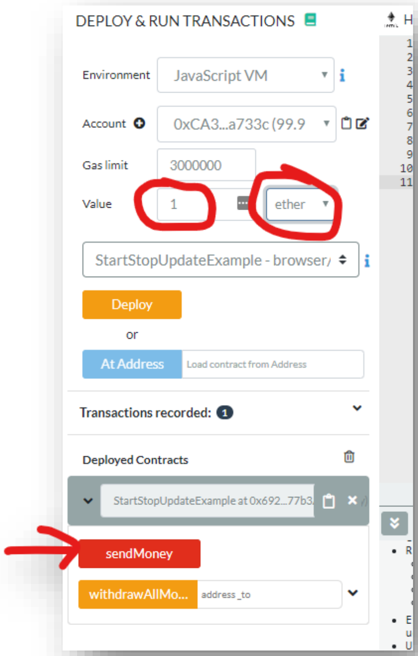
Deployed Contracts: StartStopUpdateExample at 0x692...77b3

StartStopUpdate.sol

```
pragma solidity ^0.5.11;

contract StartStopUpdateExample {
    function sendMoney() public payable {

    }

    function withdrawAllMoney(address payable _to) public {
        _to.transfer(address(this).balance);
    }
}
```

```
• Running JavaScript scripts. The following libraries are accessible:
  ○ web3 version 1.0.0
  ○ ethers.js
  ○ swarmgw
  ○ remix (run remix.help() for more info)
• Executing common command to interact with the Remix interface (see list of commands above). Note that these commands can also be incl
  uded and run from a JavaScript script.
• Use exports/.register(key, obj)/.remove(key)/.clear() to register and reuse object across script executions.


creation of StartStopUpdateExample pending...

[vm]  from:0xca3...a733c to:StartStopUpdateExample.(constructor) value:0 wei data:0x608...b0032 logs:0
hash:0x441...7c51f
```

Send some money to the Smart Contract



DEPLOY & RUN TRANSACTIONS

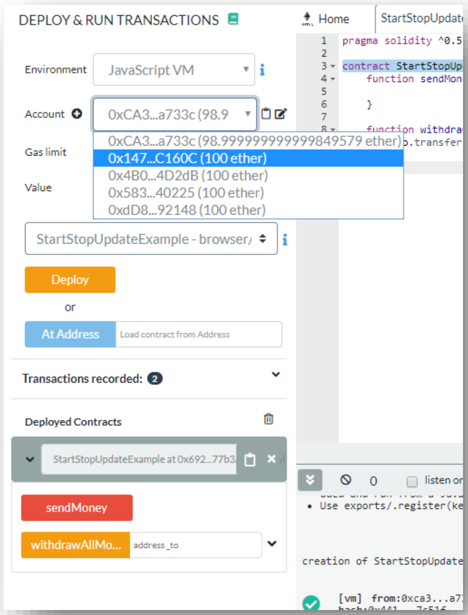
Environment: JavaScript VM
Account: 0xCA3...a733c (99.9
Gas limit: 3000000
Value: 1 ether

StartStopUpdateExample - browser/

Transactions recorded: 1

Deployed Contracts: StartStopUpdateExample at 0x692...77b3

Copy the address of the second account



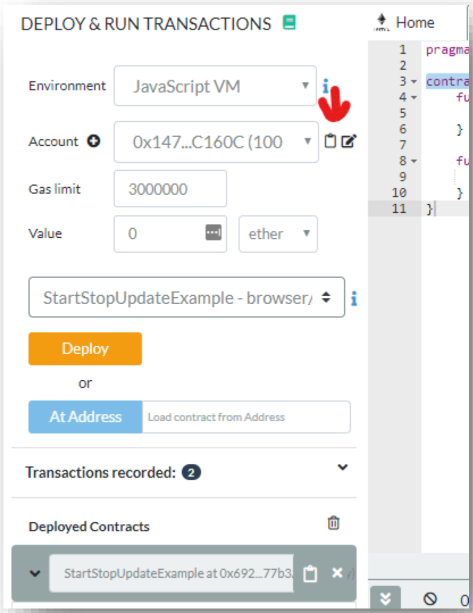
Environment: JavaScript VM

Account: 0xCA3...a733c (98.9

```
0xCA3...a733c (98.999999999999849579 ether)
0x147...C160C (100 ether)
0x4B0...4D2dB (100 ether)
0x583...40225 (100 ether)
0xdD8...92148 (100 ether)
```

```
pragma solidity ^0.5
contract StartStopUp
    function sendMon
    }
    function withdra
        .transfer
```

```
• Use exports/.register(ke
creation of StartStopUpdate
[vm] from:0xca3...a7
```


DEPLOY & RUN TRANSACTIONS
Home
Environment
Account
Gas limit
Value
StartStopUpdateExample - browser/
Deploy
or
At Address
Load contract from Address
Transactions recorded: 2
Deployed Contracts
StartStopUpdateExample at 0x692...77b3

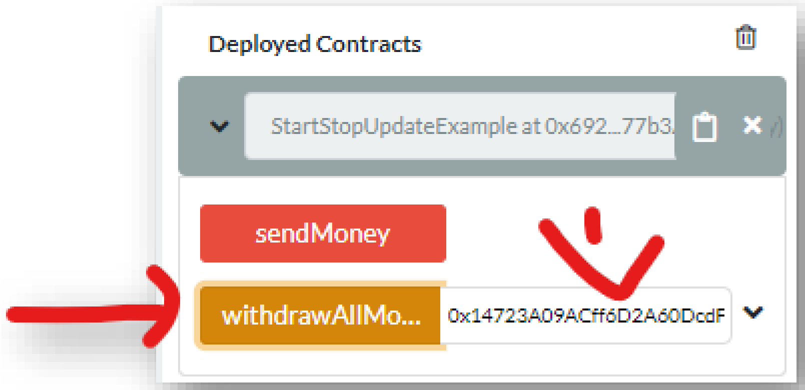
Environment: JavaScript VM

Account: 0x147...C160C (100

Gas limit: 3000000

Value: 0 ether

Withdraw the Money



withdrawAllMo... 0x14723A09ACff6D2A60DcdF

This isn't very *secure*, is it? Let's add some checks...

Update the Smart Contract

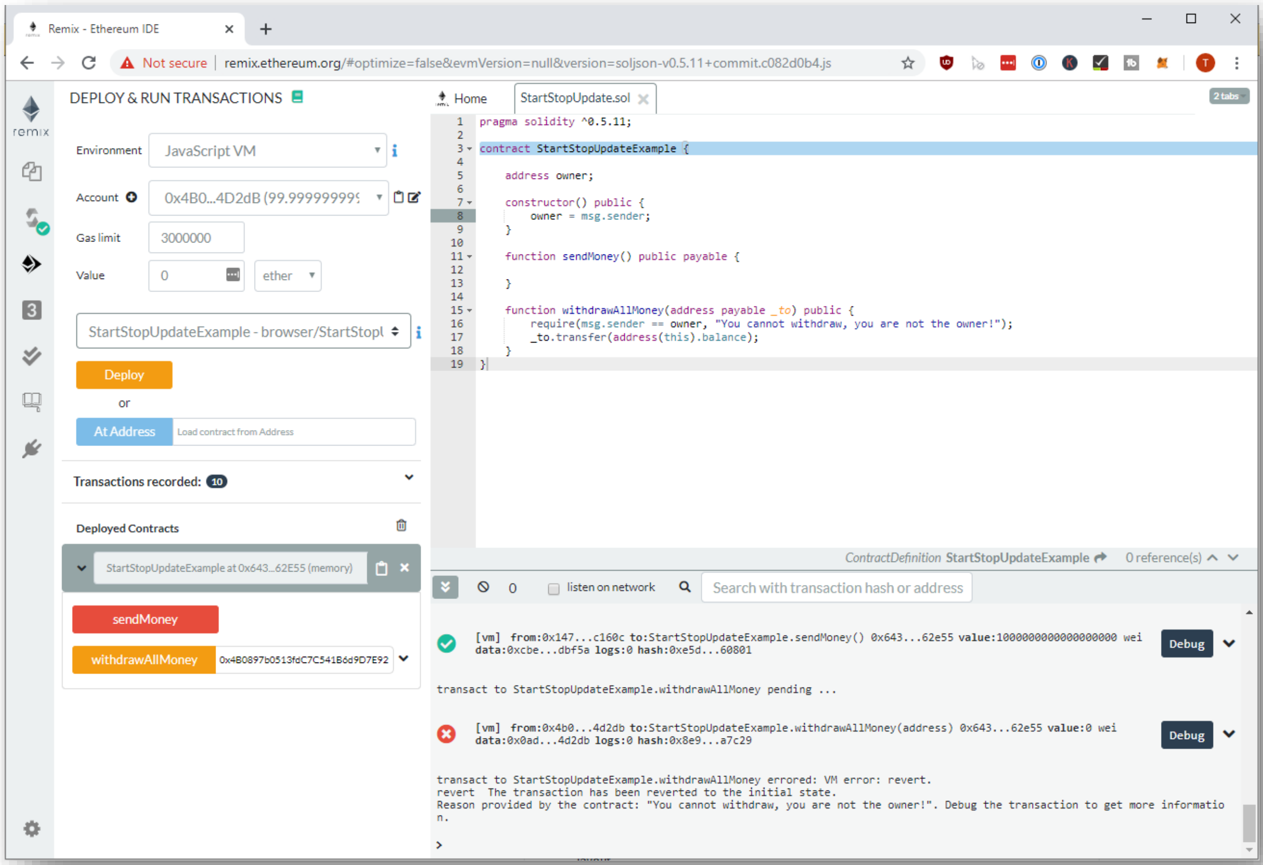
```
pragma solidity >=0.5.11 <0.7.0;

contract StartStopUpdateExample {

    address owner;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw!");
        _to.transfer(address(this).balance);
    }
}
```

Try to send and withdraw money again

Note: Don’t forget to re-deploy the smart contract.

1. Deploy the Smart Contract using **the first** account in your account list
2. Send 1 Ether to your smart contract
3. Select and Copy the **second account** from your account list
4. Try to use the withdraw method using **the second** account from your account list
5. Switch back to your first account
6. See if you can withdraw now.

This time you can see that you can send money from any account. But you can use the withdraw method only from the account which deployed the smart contract. Observe the Logs-Output:


Remix - Ethereum IDE
Not secure
remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js
DEPLOY & RUN TRANSACTIONS
Deploy
or
At Address
Load contract from Address
Deployed Contracts
sendMoney
withdrawAllMoney
Home
StartStopUpdate.sol
Debug

Environment: JavaScript VM
Account: 0x4B0...4D2dB (99.99999999
Gas limit: 3000000
Value: 0 ether
StartStopUpdateExample - browser/StartStopl
Transactions recorded: 10
StartStopUpdateExample at 0x643...62E55 (memory)
withdrawAllMoney 0x4B0897b0513fdC7C541B6d9D7E92

```
pragma solidity ^0.5.11;

contract StartStopUpdateExample {

    address owner;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw, you are not the owner!");
        _to.transfer(address(this).balance);
    }
}
```

ContractDefinition StartStopUpdateExample 0 reference(s)

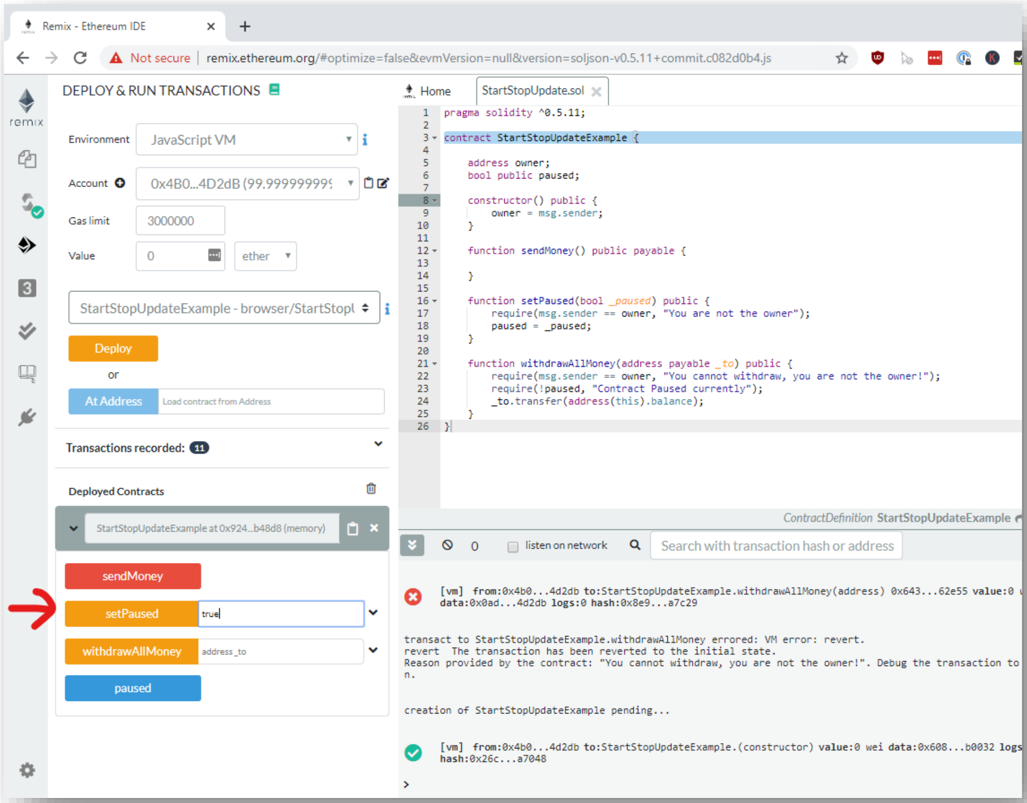
listen on network — Search with transaction hash or address

```
[vm]  from:0x147...c160c to:StartStopUpdateExample.sendMoney() 0x643...62e55 value:1000000000000000000 wei
data:0xcbe...dbf5a logs:0 hash:0xe5d...60801

transact to StartStopUpdateExample.withdrawAllMoney pending ...

[vm]  from:0x4b0...4d2db to:StartStopUpdateExample.withdrawAllMoney(address) 0x643...62e55 value:0 wei
data:0x0ad...4d2db logs:0 hash:0x8e9...a7c29

transact to StartStopUpdateExample.withdrawAllMoney errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "You cannot withdraw, you are not the owner!". Debug the transaction to get more informatio
n.
```

Pause a Smart Contract

With these new powers we got, it is easy to add a “pause” functionality. Let’s take the following code:

```
pragma solidity >=0.5.11 <0.7.0;

contract StartStopUpdateExample {

    address owner;
    bool public paused;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function setPaused(bool _paused) public {
        require(msg.sender == owner, "You are not the owner");
        paused = _paused;
    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw”);
        require(!paused, "Contract Paused currently");
        _to.transfer(address(this).balance);
    }
}
```

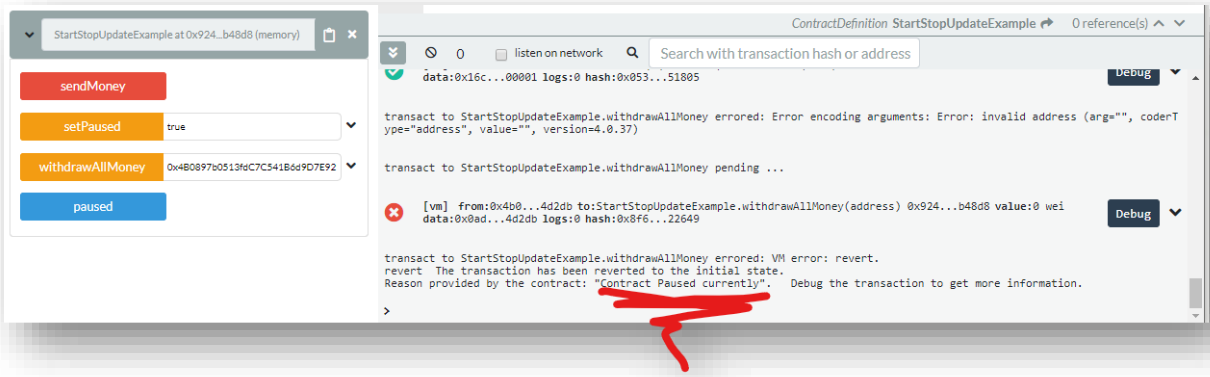

Remix - Ethereum IDE
Not secure | remix.ethereum.org/#optimize=false&evmVersion=null&version=soljson-v0.5.11+commit.c082d0b4.js
DEPLOY & RUN TRANSACTIONS
Environment
JavaScript VM
Account
0x4B0...4D2dB (99.99999999
Gas limit
3000000
Value
0
ether
StartStopUpdateExample - browser/StartStopU
Deploy
or
At Address
Load contract from Address
Transactions recorded: 11
Deployed Contracts
StartStopUpdateExample at 0x924...b48d8 (memory)
sendMoney
setPaused
true
withdrawAllMoney
address _to
paused
Home
StartStopUpdate.sol
ContractDefinition StartStopUpdateExample
listen on network
Search with transaction hash or address

```
pragma solidity ^0.5.11;

contract StartStopUpdateExample {

    address owner;
    bool public paused;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function setPaused(bool _paused) public {
        require(msg.sender == owner, "You are not the owner");
        paused = _paused;
    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw, you are not the owner!");
        require(!paused, "Contract Paused currently");
        _to.transfer(address(this).balance);
    }
}
```

```
[vm]  from:0x4b0...4d2db to:StartStopUpdateExample.withdrawAllMoney(address) 0x643...62e55 value:0
data:0x0ad...4d2db logs:0 hash:0x8e9...a7c29

transact to StartStopUpdateExample.withdrawAllMoney errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "You cannot withdraw, you are not the owner!". Debug the transaction to
n.

creation of StartStopUpdateExample pending...

[vm]  from:0x4b0...4d2db to:StartStopUpdateExample.(constructor) value:0 wei data:0x608...b0032 logs
hash:0x26c...a7048
```

Try to withdraw money

It won't work and show you an error message. The contract is paused.


StartStopUpdateExample at 0x924...b48d8 (memory)
sendMoney
setPaused
true
withdrawAllMoney
0x4B0897b0513fdC7C541B6d9D7E92
paused
ContractDefinition StartStopUpdateExample
0 reference(s)
listen on network
Search with transaction hash or address
Debug
Debug

```
data:0x16c...00001 logs:0 hash:0x053...51805

transact to StartStopUpdateExample.withdrawAllMoney errored: Error encoding arguments: Error: invalid address (arg="", coderT
ype="address", value="", version=4.0.37)

transact to StartStopUpdateExample.withdrawAllMoney pending ...

[vm]  from:0x4b0...4d2db to:StartStopUpdateExample.withdrawAllMoney(address) 0x924...b48d8 value:0 wei
data:0x0ad...4d2db logs:0 hash:0x8f6...22649

transact to StartStopUpdateExample.withdrawAllMoney errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "Contract Paused currently".   Debug the transaction to get more information.
```

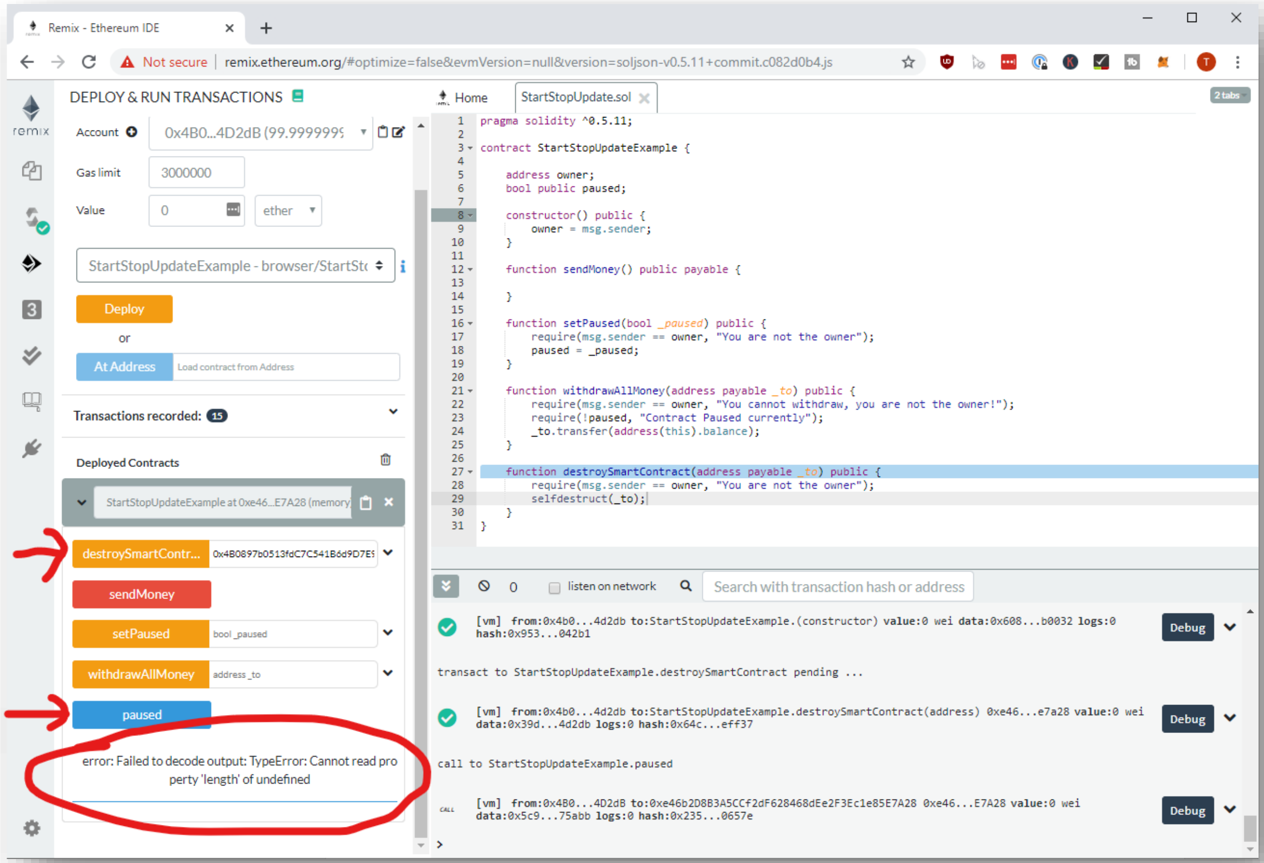
Add a destroy functionality to the Smart Contract

Consider the following source code:

```
pragma solidity >=0.5.11 <0.7.0;

contract StartStopUpdateExample {

    address owner;
    bool public paused;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function setPaused(bool _paused) public {
        require(msg.sender == owner, "You are not the owner");
        paused = _paused;
    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw!");
        require(!paused, "Contract Paused currently");
        _to.transfer(address(this).balance);
    }

    function destroySmartContract(address payable _to) public {
        require(msg.sender == owner, "You are not the owner");
        selfdestruct(_to);
    }
}
```

Stop the Smart Contract

Now deploy the new smart contract. Then copy your account address. Paste it into the “destroySmartContract” Input field. Hit the button, and then try to interact with the smart contract. It won’t work.


DEPLOY & RUN TRANSACTIONS
Account
0x4B0...4D2dB (99.9999999
Gas limit
3000000
Value
0
ether
StartStopUpdateExample - browser/StartSto
Deploy
or
At Address
Load contract from Address
Transactions recorded: 15
Deployed Contracts
StartStopUpdateExample at 0xe46...E7A28 (memory)
destroySmartContr...
0x4B0897b0513fdC7C541B6d9D7E9
sendMoney
setPaused
bool _paused
withdrawAllMoney
address _to
paused
Home
StartStopUpdate.sol
listen on network
Search with transaction hash or address
Debug

error: Failed to decode output: TypeError: Cannot read property 'length' of undefined

```
pragma solidity ^0.5.11;

contract StartStopUpdateExample {

    address owner;
    bool public paused;

    constructor() public {
        owner = msg.sender;
    }

    function sendMoney() public payable {

    }

    function setPaused(bool _paused) public {
        require(msg.sender == owner, "You are not the owner");
        paused = _paused;
    }

    function withdrawAllMoney(address payable _to) public {
        require(msg.sender == owner, "You cannot withdraw, you are not the owner!");
        require(!paused, "Contract Paused currently");
        _to.transfer(address(this).balance);
    }

    function destroySmartContract(address payable _to) public {
        require(msg.sender == owner, "You are not the owner");
        selfdestruct(_to);
    }
}
```

```
[vm] from:0x4b0...4d2db to:StartStopUpdateExample.(constructor) value:0 wei data:0x608...b0032 logs:0
hash:0x953...042b1

transact to StartStopUpdateExample.destroySmartContract pending ...

[vm] from:0x4b0...4d2db to:StartStopUpdateExample.destroySmartContract(address) 0xe46...e7a28 value:0 wei
data:0x39d...4d2db logs:0 hash:0x64c...eff37

call to StartStopUpdateExample.paused

[vm] from:0x4B0...4D2dB to:0xe46b2D8B3A5CCf2dF628468dEe2F3Ec1e85E7A28 0xe46...E7A28 value:0 wei
data:0x5c9...75abb logs:0 hash:0x235...0657e
```

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