

# Homework-1: Programming a Faucet contract

```
1  // SPDX-License-Identifier: CC-BY-SA-4.0
2
3  // Version of Solidity compiler this program was written for
4  pragma solidity 0.6.4;
5
6  // Our first contract is a faucet!
7  contract Faucet {
8      // Accept any incoming amount
9      receive() external payable {}
10
11     // Give out ether to anyone who asks
12     function withdraw(uint withdraw_amount) public {
13         // Limit withdrawal amount
14         require(withdraw_amount <= 10000000000000000);
15
16         // Send the amount to the address that requested it
17         msg.sender.transfer(withdraw_amount);
18     }
19 }
```

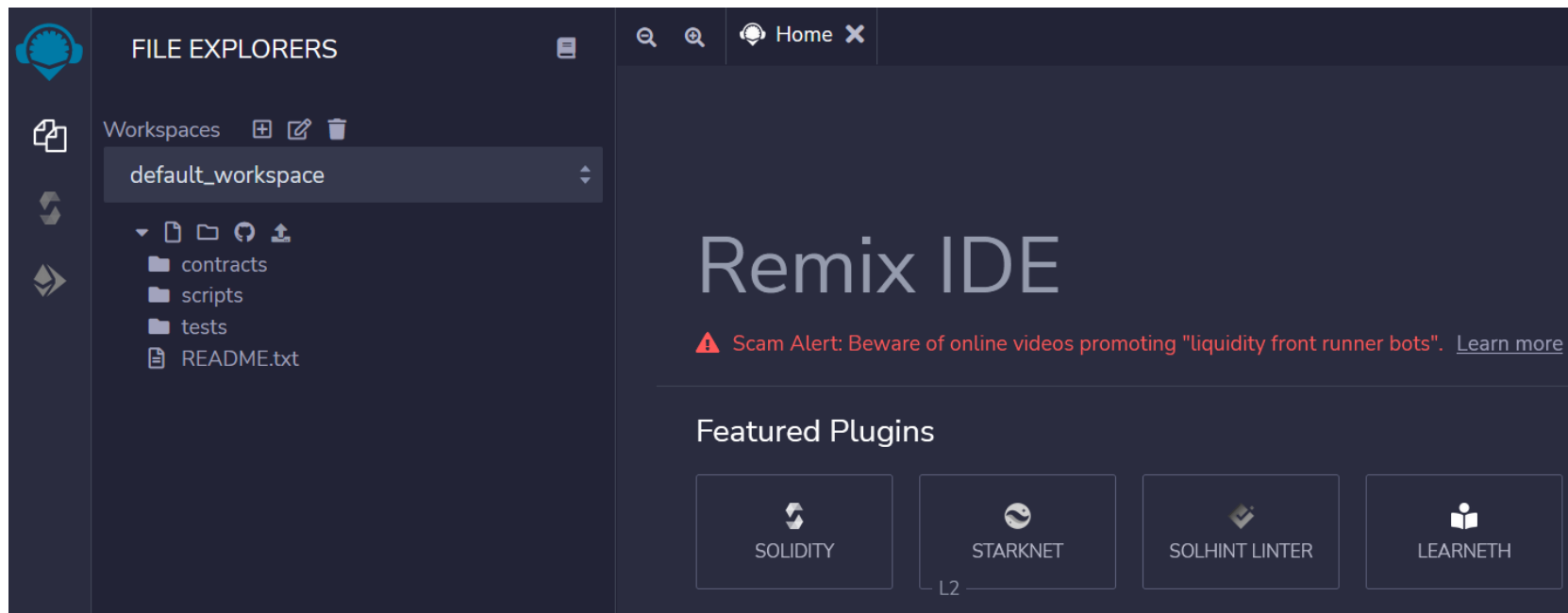
<https://github.com/ethereumbook/ethereumbook>

→ 0.1 ether

It is a *flawed* contract, demonstrating a number of bad practices and security vulnerabilities.

# Homework-1: Compiling the Contract

- Using Remix to compile the Faucet Contract
  - We need to use a Solidity compiler to convert the Solidity code into EVM bytecode so it can be executed by the EVM on the blockchain itself.
  - We use one of the more popular IDEs, called Remix.



# Homework-1: Creating the Contract

- Creating the Contract on the Blockchain
  - Registering a contract on the blockchain involves creating a special transaction whose destination is the **zero address**.
  - The zero address is a special address that tells the Ethereum blockchain that you want to register a contract.
  - Remix IDE will handle all of that for you and send the transaction to MetaMask.
  - Switch to the Run tab and select Injected Web3 in the Environment drop-down selection box. This connects the Remix IDE to the MetaMask wallet, and through MetaMask to the Ropsten test network.
  - Remix will construct the special “creation” transaction and MetaMask will ask you to approve it.

# Homework-1: Contract Deployment


## After MetaMask Confirmed

### Contract Deployment

Confirmed

Copy Transaction ID

From

 0xc91...A5e3

→

To

New Contract

#### Transaction

Nonce	1
Amount	-0 ETH
Gas Limit (Units)	106071
Gas Used (Units)	106071
Base Fee (GWEI)	0.000000011
Priority Fee (GWEI)	2.5
Total Gas Fee	0.000265 ETH
Max Fee Per Gas	0.000000003 ETH
Total	0.00026518 ETH

## Transaction Details



## Faucet Creation Transaction

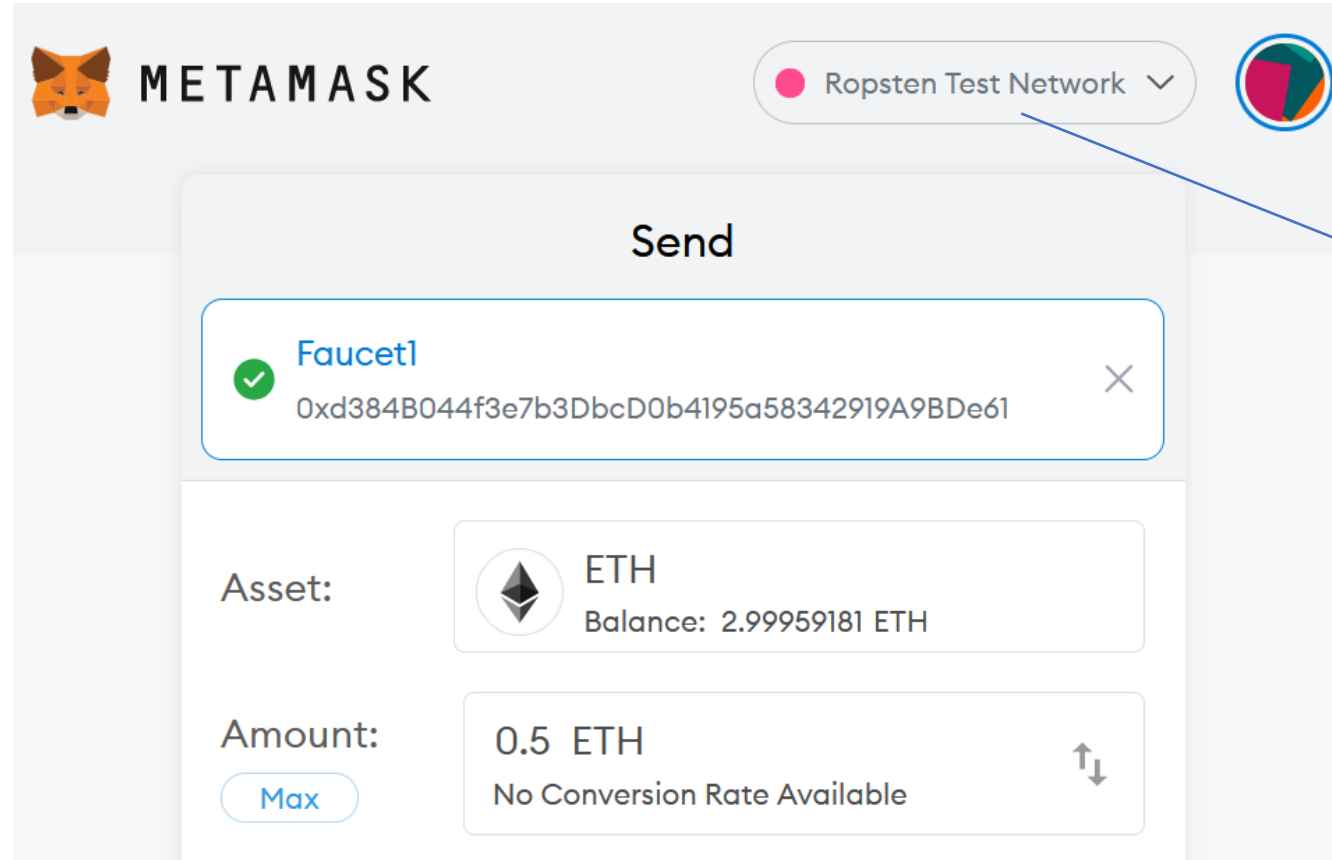
### Overview

### State

[ This is a Ropsten Testnet transaction only ]




Transaction Hash:	0xa80fe10fc24c3a05b991292333d218fb69daab1267aa927969ba4a13565ebc5b
Status:	Success
Block:	12159077 12 Block Confirmations
Timestamp:	7 mins ago (Apr-03-2022 10:51:06 AM +UTC)
From:	0xc9192277ea18ff49618e412197c9c9eacf43a5e3
To:	[Contract 0xd384b044f3e7b3dbcd0b4195a58342919a9bde61 Created]
Value:	0 Ether (\$0.00)
Transaction Fee:	0.000265177501166781 Ether (\$0.00)
Gas Price:	0.000000002500000011 Ether (2.500000011 Gwei)

# Homework-1: Funding the Faucet (1/2)



After the Merge use  
**Sepolia/Goerli** Testnet

# Homework-1: Funding the Faucet (2/2)

 Contract 0xd384B044f3e7b3DbcD0b4195a58342919A9BDe61  

### Contract Overview

Balance: 0.5 Ether

### More Info


My Name Tag: Not Available










Contract Creator: [0xc9192277ea18ff49618...](#) at txn [0xa80fe10fc24c3a05b99...](#)



Transactions

Contract

Events

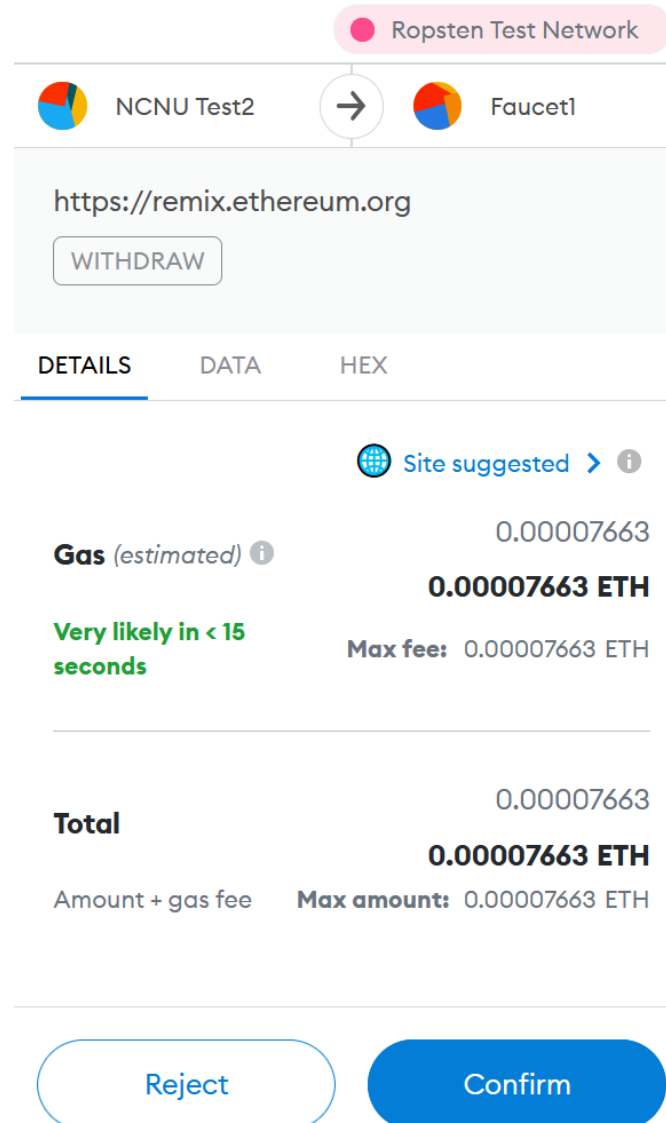
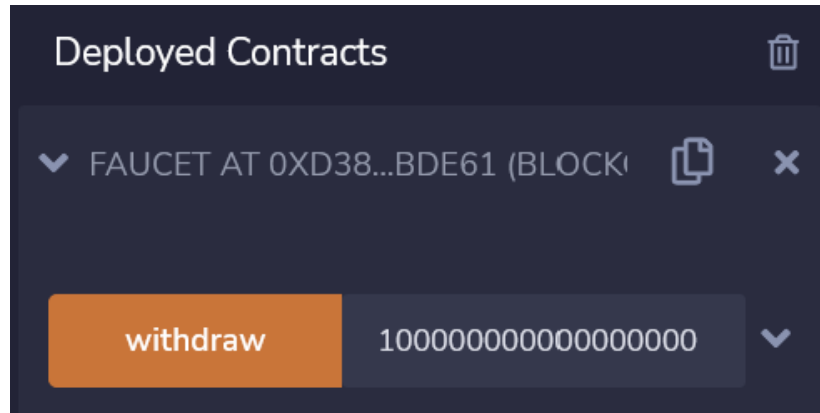
⌵ Latest 2 from a total of 2 transactions 

	Txn Hash	Method 	Block	Age	From 	To 	Value	Txn Fee
	<a href="#">0xec9b3b28f00001f7175...</a>	Transfer	<a href="#">12159154</a>	2 mins ago	<a href="#">0xc9192277ea18ff49618...</a>	 <a href="#">0xd384b044f3e7b3dbcd...</a>	0.5 Ether	0.0000315825 
	<a href="#">0xa80fe10fc24c3a05b99...</a>	0x60806040	<a href="#">12159077</a>	53 mins ago	<a href="#">0xc9192277ea18ff49618...</a>	 Contract Creation	0 Ether	0.000265177501 

# Homework-1: Withdraw from the Faucet (1/2)

## Remix



Asking MetaMask to confirm the withdraw transaction

# Homework-1: Withdraw from the Faucet (2/2)

Etherscan

Ropsten Testnet Network

Home Blockchain Tokens Misc Ropsten

Address `0xc9192277ea18ff49618E412197C9c9eaCF43A5e3`

**Overview**

Balance: 2.599489229997673274 Ether

**More Info** [More](#)

My Name Tag: Not Available

**Transactions** [Internal Txns](#)

Latest 1 internal transaction

Parent Txn Hash	Block	Age	From	To	Value
<a href="#">0xac879e25860bb7a4fc...</a>	<a href="#">12159167</a>	11 mins ago	<a href="#">0xd384b044f3e7b3dbcd...</a>	<a href="#">0xc9192277ea18ff49618...</a>	0.1 Ether

[ Download [CSV Export](#) ]

Because the 0.1 ether transfer originated from the contract code, it is an internal transaction (also called a message).



## Q. Who pays and How much for Smart Contracts

1. Who pays for the creation (deployment) of the faucet contract?  
How much is the gas fee in this homework?
2. Who pays for withdrawing from the faucet?  
How much is the gas fee in this homework?
3. Double-check your answers on MetaMask and Etherscan.