

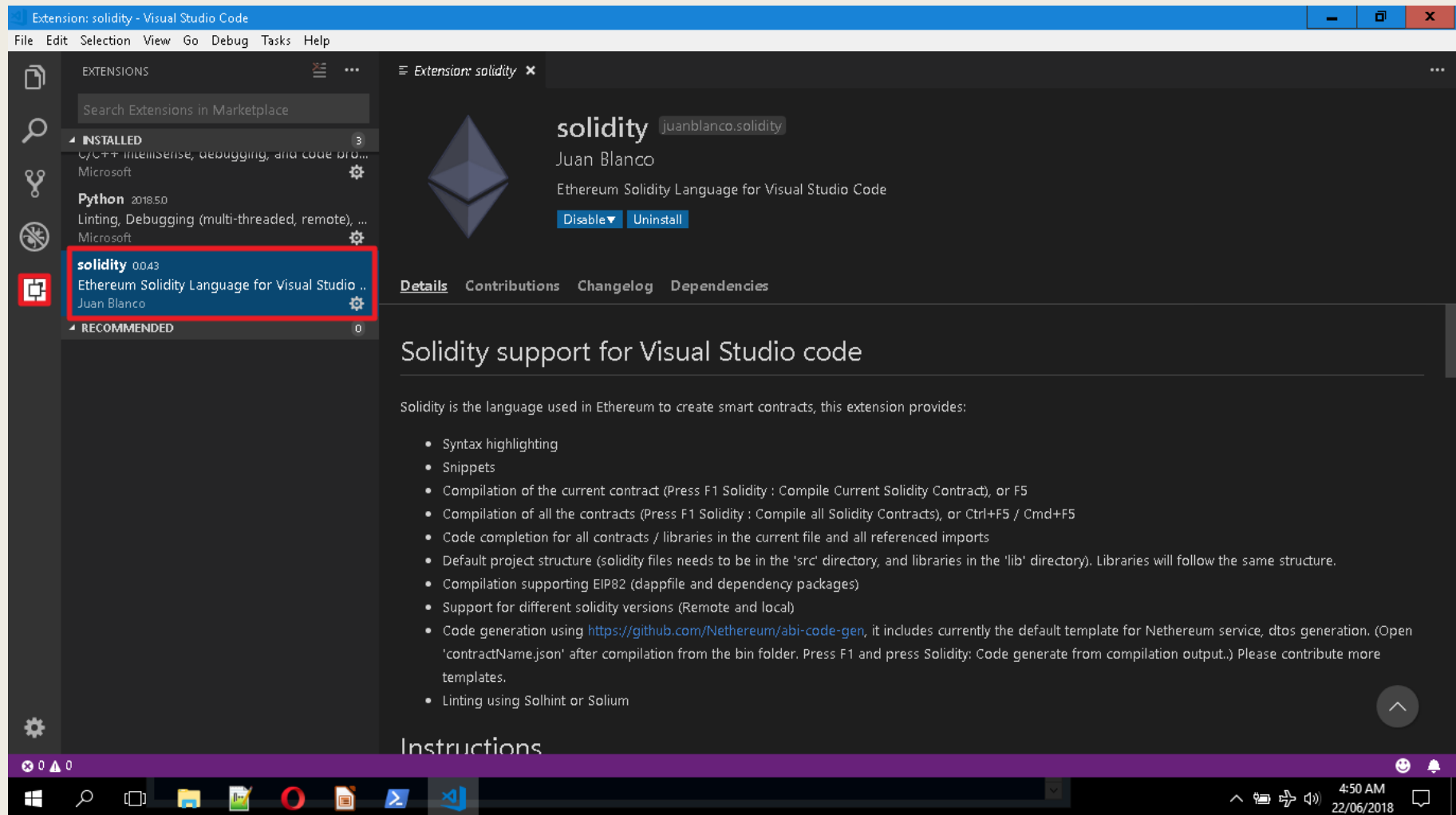
# Blockchain Training



An initiative of the Lagos State Ministry of Education

Week 3

# Check for the Visual Studio Code Solidity Extension



# Check for Truffle installation using Powershell

```
PS C:\Users\Truston Ailende> truffle
Truffle v4.1.11 - a development framework for Ethereum

Usage: truffle <command> [options]

Commands:
  init      Initialize new and empty Ethereum project
  compile   Compile contract source files
  migrate   Run migrations to deploy contracts
  deploy    (alias for migrate)
  build     Execute build pipeline (if configuration present)
  test      Run JavaScript and Solidity tests
  debug     Interactively debug any transaction on the blockchain (experimental)
  opcode    Print the compiled opcodes for a given contract
  console   Run a console with contract abstractions and commands available
  develop   Open a console with a local development blockchain
  create    Helper to create new contracts, migrations and tests
  install   Install a package from the Ethereum Package Registry
  publish   Publish a package to the Ethereum Package Registry
  networks  Show addresses for deployed contracts on each network
  watch     Watch filesystem for changes and rebuild the project automatically
  serve     Serve the build directory on localhost and watch for changes
  exec      Execute a JS module within this Truffle environment
  unbox     Download a Truffle Box, a pre-built Truffle project
  version   Show version number and exit

See more at http://truffleframework.com/docs

PS C:\Users\Truston Ailende>
```

# Ganache

- Creates a virtual Ethereum blockchain
- Generates fake accounts pre-populated with Ether
- Blockchain simulator
- By default runs on localhost:7545 on the RPC Server

# Smart Contracts

- A smart contract is a collection of code and data that reside at a specific address on the Ethereum blockchain
- Smart contracts are written in some high level programming language and then compiled into bytecode to be uploaded into the blockchain

# Steps to Create a Smart Contract

- 1) A contract account with rules is created
- 2) The contract is coded in a Ethereum high level language such as Solidity and then deployed on the Ethereum blockchain
- 3) Once deployed, the contract gets a public key address, that can be used to reach the contract and trigger its code execution
- 4) Once a smart contract is deployed to the Ethereum blockchain, it cannot be changed

# Initialize Your Project

Open FILE EXPLORER and create a folder called Node in Local Disk (C:/)

Open Powershell and type

```
cd C:/Node
```

# Initialize Your Project

```
Truston Ailende@Oiselenjakhian MINGW64 ~/Node
$ mkdir my-first-smart-contract

Truston Ailende@Oiselenjakhian MINGW64 ~/Node
$ cd my-first-smart-contract

Truston Ailende@Oiselenjakhian MINGW64 ~/Node/my-first-smart-contract
$ truffle init
Downloading...
Unpacking...
Setting up...
Unbox successful. Sweet!

Commands:

  Compile:      truffle compile
  Migrate:      truffle migrate
  Test contracts: truffle test

Truston Ailende@Oiselenjakhian MINGW64 ~/Node/my-first-smart-contract
$ !
```






Use Powershell





# Folder Structure

Windows (C:) > Users > Truston Ailende > Node > my-first-smart-contract

Name	Date modified	Type	Size
 contracts	23/06/2018 5:52 PM	File folder	
 migrations	23/06/2018 5:52 PM	File folder	
 test	23/06/2018 5:52 PM	File folder	
 truffle	23/06/2018 5:52 PM	JavaScript File	1 KB
 truffle-config	23/06/2018 5:52 PM	JavaScript File	1 KB

Enter the folder you created

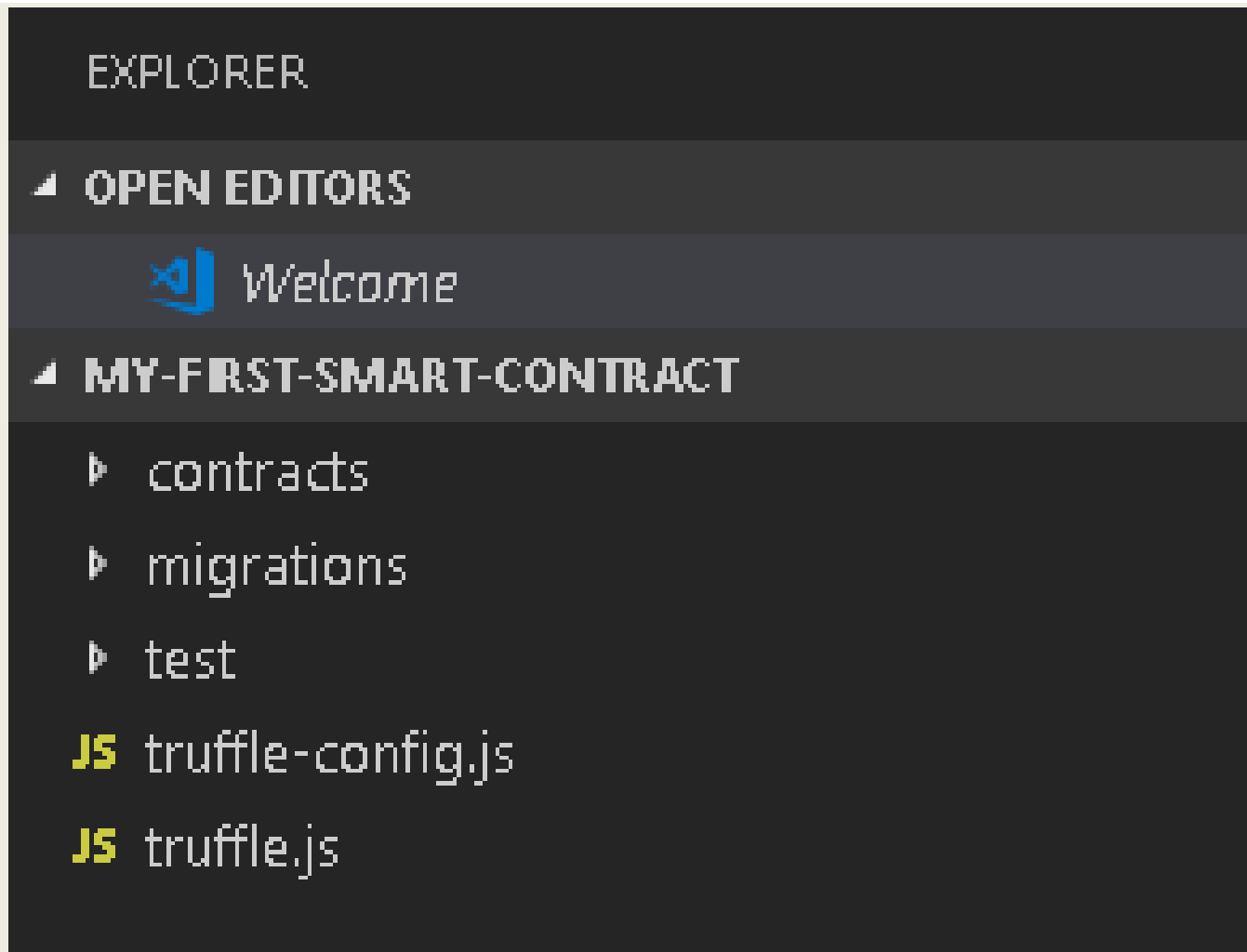
# Open Ganache

The screenshot shows the Ganache desktop application. At the top, there's a blue header bar with the Ganache logo and window controls. Below this is a navigation bar with icons for ACCOUNTS, BLOCKS, TRANSACTIONS, and LOGS. A search bar is on the right. Below the navigation bar is a status bar showing various metrics: CURRENT BLOCK (0), GAS PRICE (20000000000), GAS LIMIT (6721975), NETWORK ID (5777), RPC SERVER (HTTP://127.0.0.1:7545), and MINING STATUS (AUTOMINING). The main area displays the MNEMONIC (trim spell search start junk learn yellow giggle practice defense globe wire) and the HD PATH (m/44'/60'/0'/0'/account\_index). Below this is a table of accounts.

ADDRESS	BALANCE	TX COUNT	INDEX	
0xeCb4f5A2D3cB193D798BCaAA3Eb2d76cf40f3945	100.00 ETH	0	0	
0xb8db6c1baD23B91af09a5390E0646D14bE7e5b5c	100.00 ETH	0	1	
0x29E17c8dcBe5fAb71E2aa43feFa92225Db385A46	100.00 ETH	0	2	
0x8d016117A5637faf9C5ac80C66F6A4CE76f36676	100.00 ETH	0	3	
0xD1e2931EA4909279bBB5B70C437Bf762F1622B14	100.00 ETH	0	4	
0x7eA4b17b4A5B791C4C4606f968f1A642bEC3b3B0	100.00 ETH	0	5	

The bottom of the screenshot shows the Windows taskbar with various application icons and the system clock displaying 4:56 AM on 22/06/2018.

# Open Folder in Visual Studio Code



# Modify Your truffle.js File

```
module.exports = {  
  networks: {  
    development: {  
      host: 'localhost',  
      port: 7545,  
      network_id: '*'  
    }  
  }  
};
```

# Our First Smart Contract

```
pragma solidity ^0.4.17;

contract Message {
    bytes32 public message;

    function setMessage(bytes32 newMessage) public {
        message = newMessage;
    }

    function getMessage() public view returns (bytes32) {
        return message;
    }
}
```

NB: Use pragma solidity >0.4.23; on first line  
Create the Message.sol file  
in the contracts Folder



# Deploying to the Test Blockchain

- To deploy a contract we have to write a migration file
- In the migrations folder, create a migration file called `2_message_migration.js`

# 2\_message\_migration.js

```
var Message = artifacts.require('./Message.sol');  
module.exports = function(deployer) {  
  ...  
  deployer.deploy(Message);  
};
```

# Contract Deployment



- In the shell, type in the following commands:  
truffle compile  
truffle migrate
- Run the first command and wait for it to finish before running the second one



# Check Your Blocks in Ganache

ACCOUNTS	BLOCKS	TRANSACTIONS	LOGS	SEARCH FOR BLOCK NUMBERS OR TX HASHES		
CURRENT BLOCK 4	GAS PRICE 20000000000	GAS LIMIT 6721975	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING	
BLOCK 4	MINED ON 2018-06-23 23:51:08			GAS USED 27008	1 TRANSACTION	
BLOCK 3	MINED ON 2018-06-23 23:51:08			GAS USED 136007	1 TRANSACTION	
BLOCK 2	MINED ON 2018-06-23 23:51:07			GAS USED 42008	1 TRANSACTION	
BLOCK 1	MINED ON 2018-06-23 23:51:07			GAS USED 277398	1 TRANSACTION	
BLOCK 0	MINED ON 2018-06-23 22:56:26			GAS USED 0	NO TRANSACTIONS	

# Interacting with the Contract

- Run the following commands in order:

```
truffle console
```

```
var message;
```

```
Message.deployed().then(function(instance)  
{message = instance;})
```

```
message.setMessage(web3.fromAscii('My First  
Message!'));
```

Use the Powershell Console

# A New Block is Added

ACCOUNTS	BLOCKS	TRANSACTIONS	LOGS	SEARCH FOR BLOCK NUMBERS OR TX HASHES			
CURRENT BLOCK 5	GAS PRICE 20000000000	GAS LIMIT 6721975	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING		
BLOCK 5	MINED ON 2018-06-24 00:08:56			GAS USED 42711	1 TRANSACTION		
BLOCK 4	MINED ON 2018-06-23 23:51:08			GAS USED 27008	1 TRANSACTION		
BLOCK 3	MINED ON 2018-06-23 23:51:08			GAS USED 136007	1 TRANSACTION		
BLOCK 2	MINED ON 2018-06-23 23:51:07			GAS USED 42008	1 TRANSACTION		
BLOCK 1	MINED ON 2018-06-23 23:51:07			GAS USED 277398	1 TRANSACTION		
BLOCK 0	MINED ON 2018-06-23 22:56:26			GAS USED 0	NO TRANSACTIONS		

# Check Your Transactions

Ganache

ACCOUNTS BLOCKS **TRANSACTIONS** LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

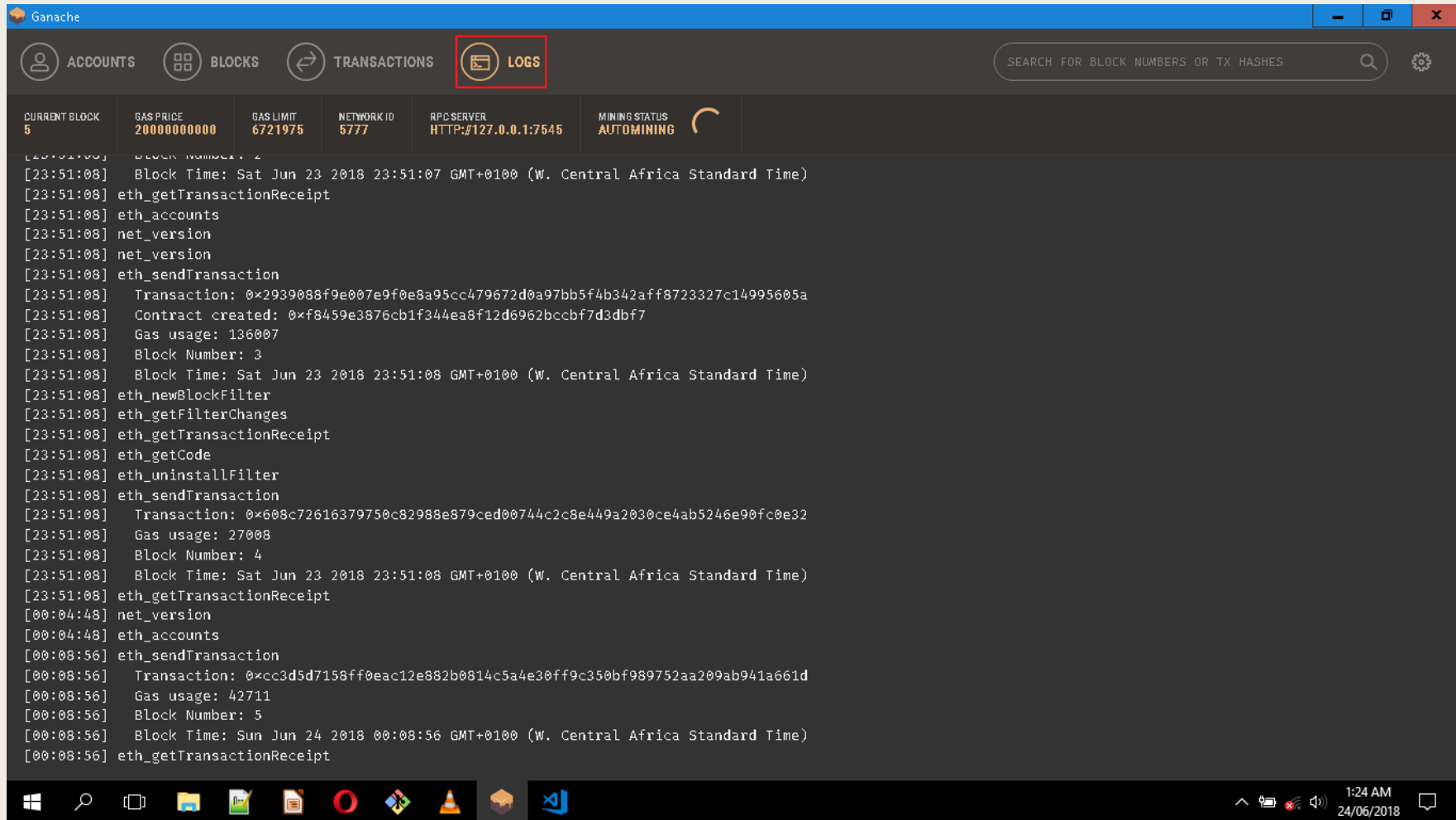
CURRENT BLOCK	GAS PRICE	GAS LIMIT	NETWORK ID	RPC SERVER	MINING STATUS
5	20000000000	6721975	5777	HTTP://127.0.0.1:7545	AUTOMINING

TX HASH	FROM ADDRESS	TO CONTRACT ADDRESS	GAS USED	VALUE	
0xccc3d5d7158ff0eac12e882b0814c5a4e30ff9c350bf989752aa209ab941a661d	0xeCb4f5A2D3cB193D798BCaAA3Eb2d76cf40f3945	0xF8459E3876cb1f344eA8f12D6962BccBf7d3dbF7	42711	0	CONTRACT CALL
0x608c72616379750c82988e879ced00744c2c8e449a2030ce4ab5246e90fc0e32	0xeCb4f5A2D3cB193D798BCaAA3Eb2d76cf40f3945	0x0dea76Bd725B9B796991ea1657BD633685DaeB4e	27008	0	CONTRACT CALL
0x2939088f9e007e9f0e8a95cc479672d0a97bb5f4b342aff8723327c14995605a	0xeCb4f5A2D3cB193D798BCaAA3Eb2d76cf40f3945	0xF8459E3876cb1f344eA8f12D6962BccBf7d3dbF7	136007	0	CONTRACT CREATION
0xb0e3b8207fc9654f9eaf8fc396175a8eb44ebf7baef9927da4578b2aff501152	0xeCb4f5A2D3cB193D798BCaAA3Eb2d76cf40f3945	0x0dea76Bd725B9B796991ea1657BD633685DaeB4e	42008	0	CONTRACT CALL
0x7cbf634eb4ed5e09117fabbebc6ec47b2e20aa99eecb0cf094930cb0417b21ae					CONTRACT CREATION

1:22 AM 24/06/2018

# Check Your Logs



# Get Your Message



- To get your message from the console, enter the command shown below:

```
message.getMessage().then(function(message)
){console.log(web3.toAscii(message));})
```

# Check Your Blocks



ACCOUNTS	BLOCKS	TRANSACTIONS	LOGS	SEARCH FOR BLOCK NUMBERS OR TX HASHES		
CURRENT BLOCK 5	GAS PRICE 20000000000	GAS LIMIT 6721975	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING	
BLOCK 5	MINED ON 2018-06-24 00:08:56			GAS USED 42711	1 TRANSACTION	
BLOCK 4	MINED ON 2018-06-23 23:51:08			GAS USED 27008	1 TRANSACTION	
BLOCK 3	MINED ON 2018-06-23 23:51:08			GAS USED 136007	1 TRANSACTION	
BLOCK 2	MINED ON 2018-06-23 23:51:07			GAS USED 42008	1 TRANSACTION	
BLOCK 1	MINED ON 2018-06-23 23:51:07			GAS USED 277398	1 TRANSACTION	
BLOCK 0	MINED ON 2018-06-23 22:56:26			GAS USED 0	NO TRANSACTIONS	

# Check Your Logs



Ganache

ACCOUNTS BLOCKS TRANSACTIONS LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK	GAS PRICE	GAS LIMIT	NETWORK ID	RPC SERVER	MINING STATUS
5	20000000000	6721975	5777	HTTP://127.0.0.1:7545	AUTOMINING

```
[23:51:08] BLOCK TIME: Sat Jun 23 2018 23:51:08 GMT+0100 (W. Central Africa Standard Time)
[23:51:08] eth_getTransactionReceipt
[23:51:08] eth_accounts
[23:51:08] net_version
[23:51:08] net_version
[23:51:08] eth_sendTransaction
[23:51:08] Transaction: 0x2939088f9e07e9f0e8a95cc479672d0a97bb5f4b342aff8723327c14995605a
[23:51:08] Contract created: 0xf8459e3876cb1f344ea8f12d6962bccbf7d3dbf7
[23:51:08] Gas usage: 136007
[23:51:08] Block Number: 3
[23:51:08] Block Time: Sat Jun 23 2018 23:51:08 GMT+0100 (W. Central Africa Standard Time)
[23:51:08] eth_newBlockFilter
[23:51:08] eth_getFilterChanges
[23:51:08] eth_getTransactionReceipt
[23:51:08] eth_getCode
[23:51:08] eth_uninstallFilter
[23:51:08] eth_sendTransaction
[23:51:08] Transaction: 0x608c72616379750c82988e879ced00744c2c8e449a2030ce4ab5246e90fc0e32
[23:51:08] Gas usage: 27008
[23:51:08] Block Number: 4
[23:51:08] Block Time: Sat Jun 23 2018 23:51:08 GMT+0100 (W. Central Africa Standard Time)
[23:51:08] eth_getTransactionReceipt
[00:04:48] net_version
[00:04:48] eth_accounts
[00:08:56] eth_sendTransaction
[00:08:56] Transaction: 0xcc3d5d7158ff0eac12e882b0814c5a4e30ff9c350bf989752aa209ab941a661d
[00:08:56] Gas usage: 42711
[00:08:56] Block Number: 5
[00:08:56] Block Time: Sun Jun 24 2018 00:08:56 GMT+0100 (W. Central Africa Standard Time)
[00:08:56] eth_getTransactionReceipt
[01:36:41] eth_call
```

Windows taskbar: 1:42 AM 24/06/2018



# Conclusion



- This week we created our first smart contract
- We deployed it on Ganache
- Whenever a transaction runs, it modifies the state of our contract and adds a new block to the Blockchain
- At this point, you should have finished the CryptoZombies lessons

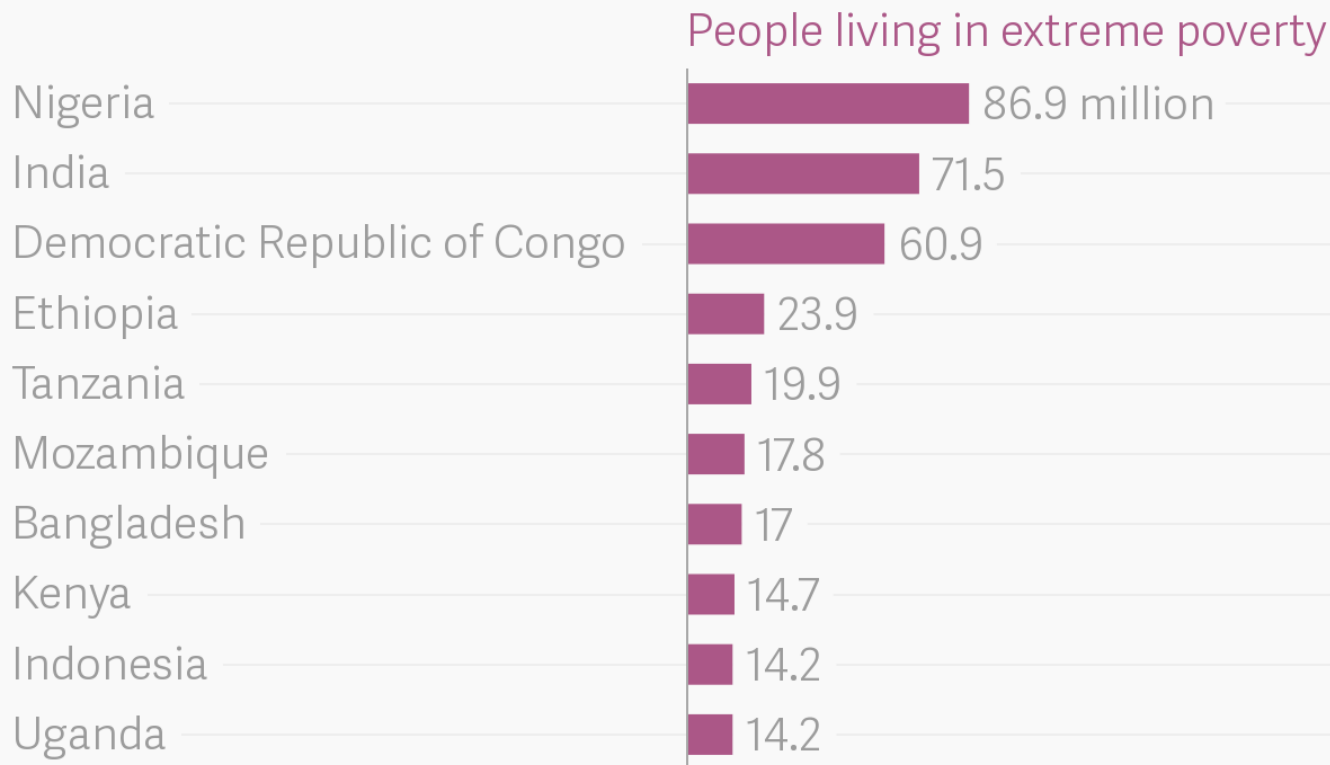
# A Time To Reflect



- Why did you decide to take this course?
- What have you done with the knowledge you have gained so far?
- How do you plan to use the knowledge you gain from this course to improve your economic welfare?
- Do you make money online?
- What is the use of a system that controls people but does not protect them?

# Nigeria is the Poverty Capital of the World

Nigeria has the largest extreme poverty population (June 2018)



△ T L △ S | Data: World Poverty Clock

<https://qz.com/1313380/nigerias-has-the-highest-rate-of-extreme-poverty-globally/>

# A Response is Needed



CodeLagos Blockchain  
Technology Group



# Objectives



- To keep abreast of emerging trends in Blockchain technology
- To facilitate economic development in Lagos State
- To explore use cases for Blockchain technology and apply them locally
- To improve the training of Blockchain developers
- Empower members to achieve financial independence using Blockchain technology

# Requirements



- Focus
- Incubation
- Resources/Roadmap
- Effort

# Xerox PARC



- Palo Alto Research Company (PARC)
- Opened on July 1, 1970
- A division of Xerox Corp
- Developed the Xerox Alto

# Impact



- Developed the first computer system to use the GUI
- A number of its GUI engineers left to join Apple Computer
- John Warnock a former researcher in Xerox PARC became one of the two founders of Adobe Systems



# Conclusion



- Blockchain technology is at the nexus of technology, economic development and governance
- The Problem You Solve Is More Important Than The Code You Write
- At this point you should be able to write and deploy a smart contract