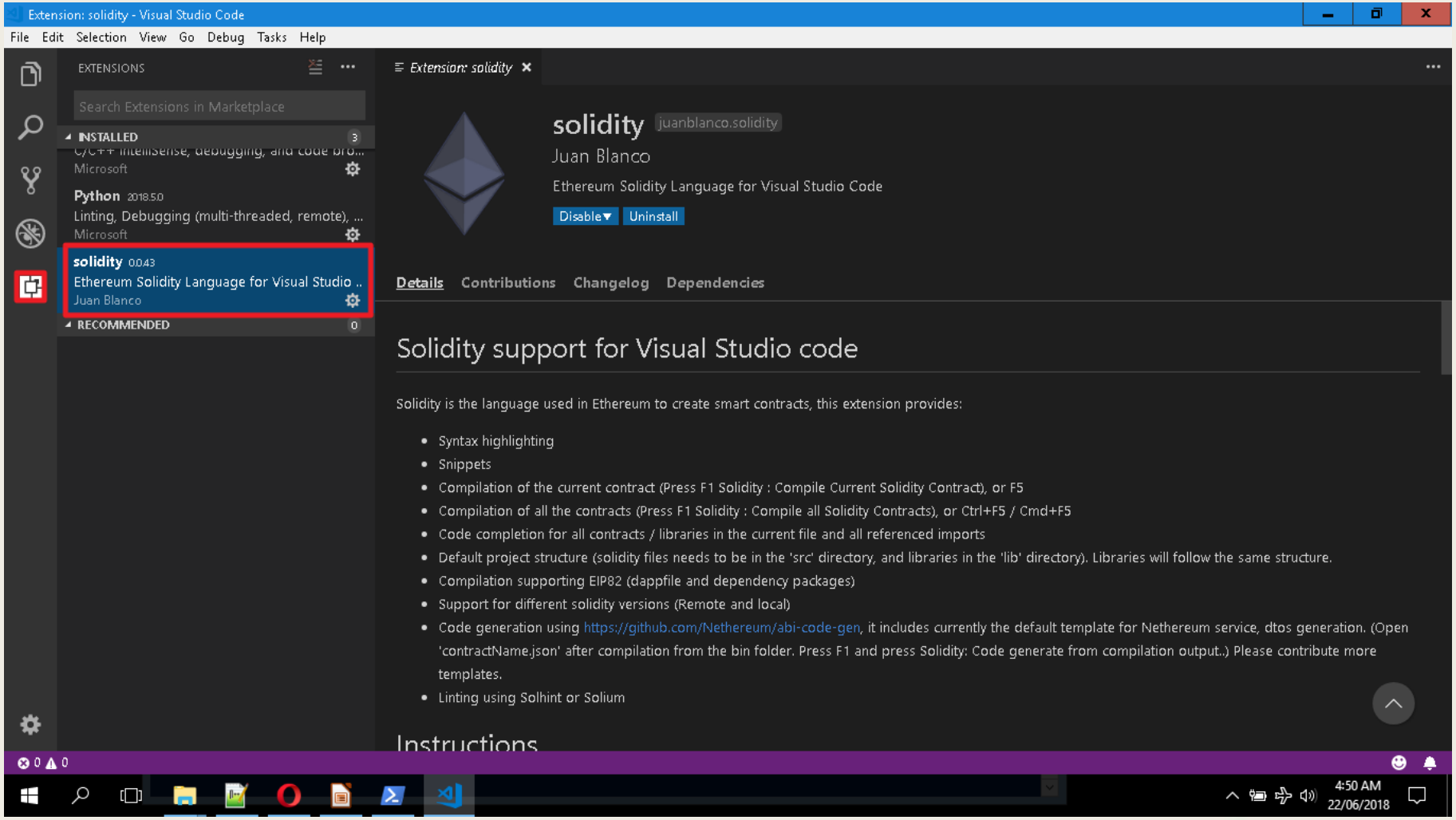
Blockchain Training

code LAGOS

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

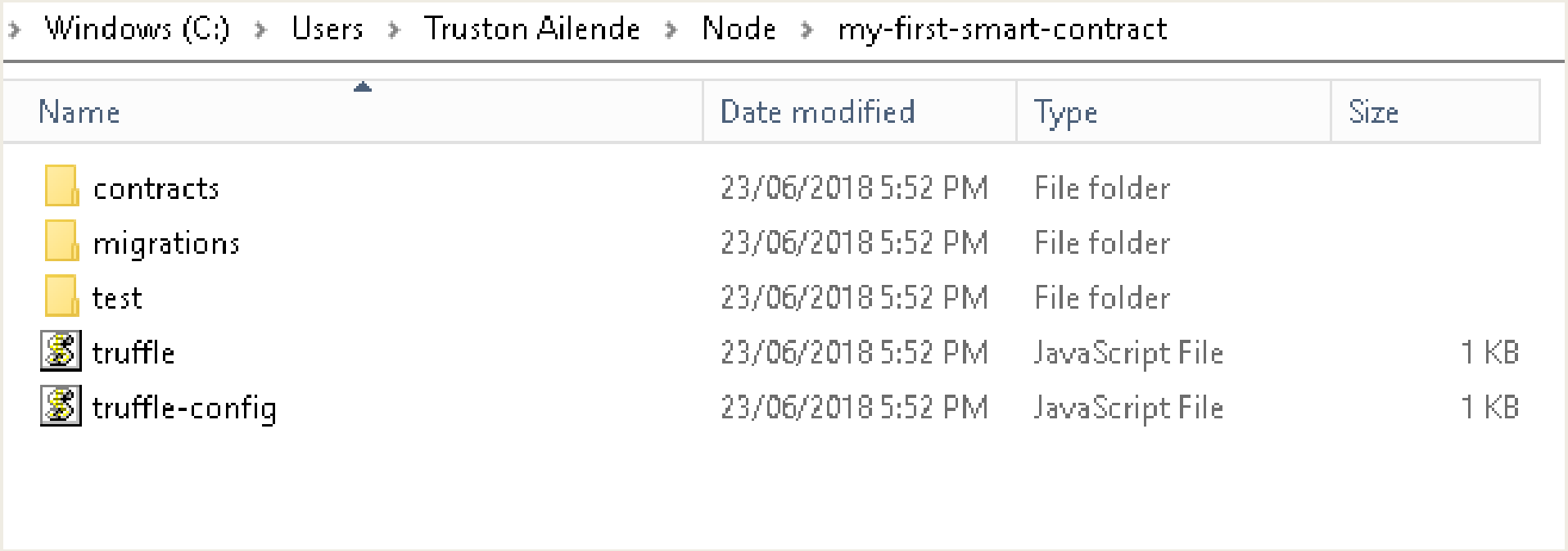
cd C:/Node

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
$
```

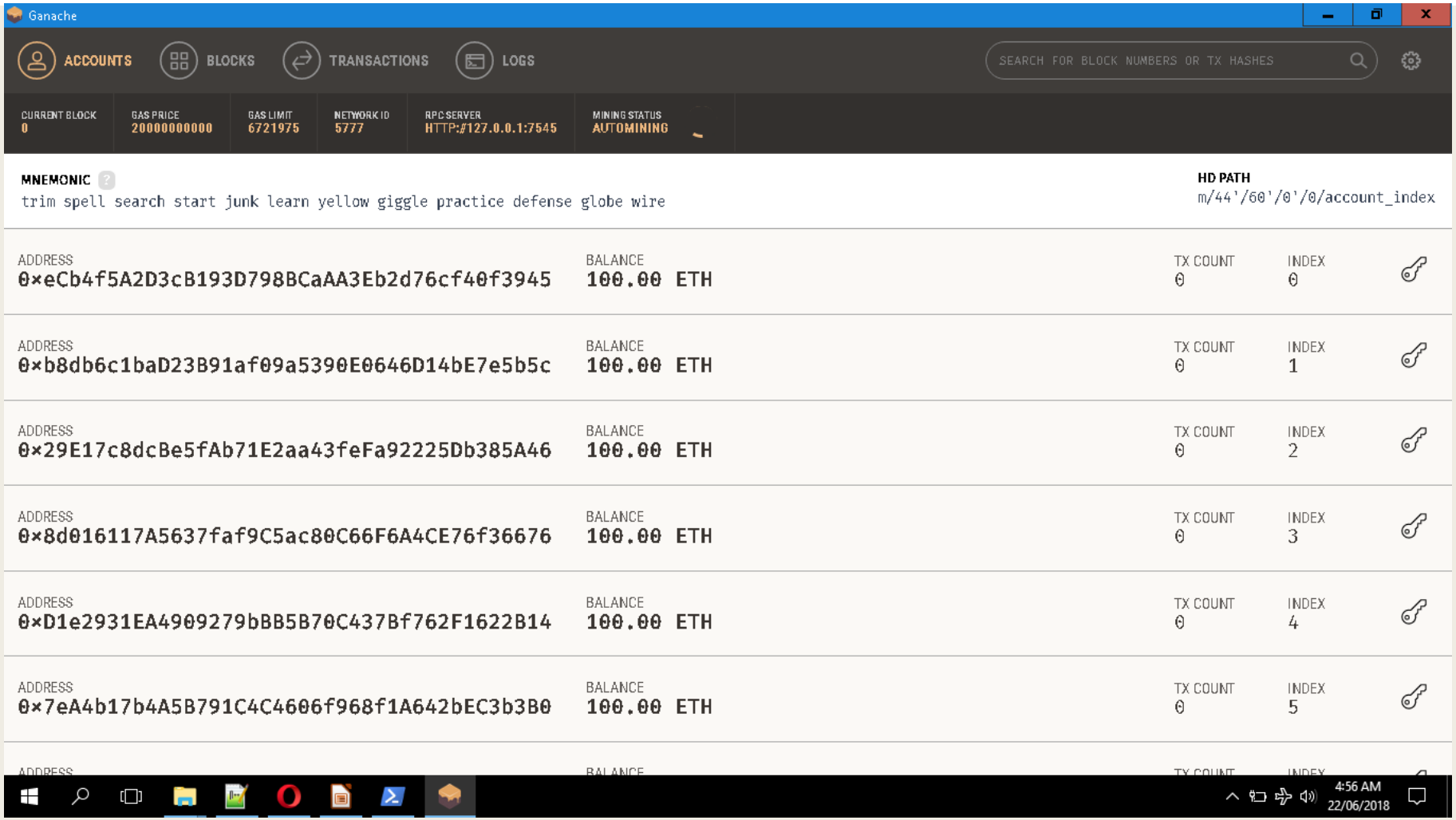
Use Powershell

Folder Structure

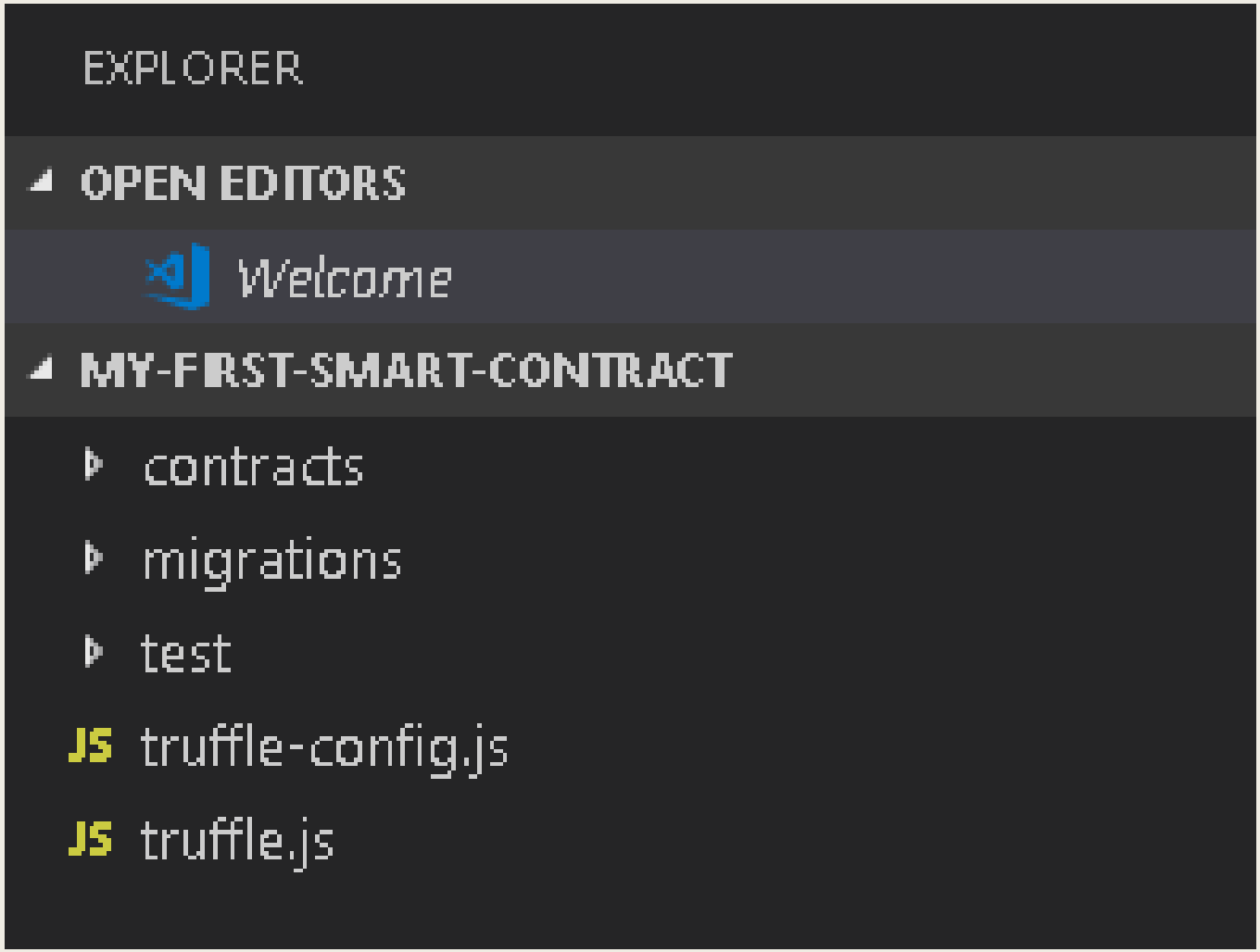


Enter the folder you created

Open Ganache



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

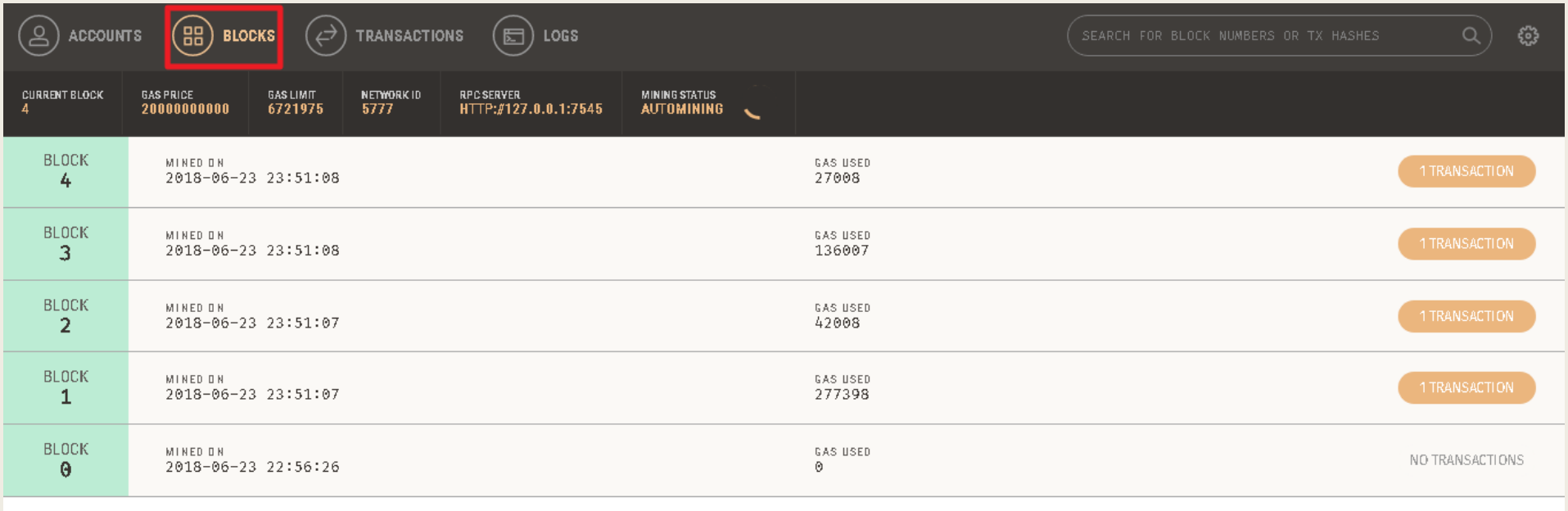
- In the shell, type in the following commands:

  truffle compile

  truffle migrate

- Run the first command and wait for it to finish before running the second one

Check Your Blocks in Ganache



code LAGOS

Interacting with the Contract

- Run the following commands in order:

  truffle console

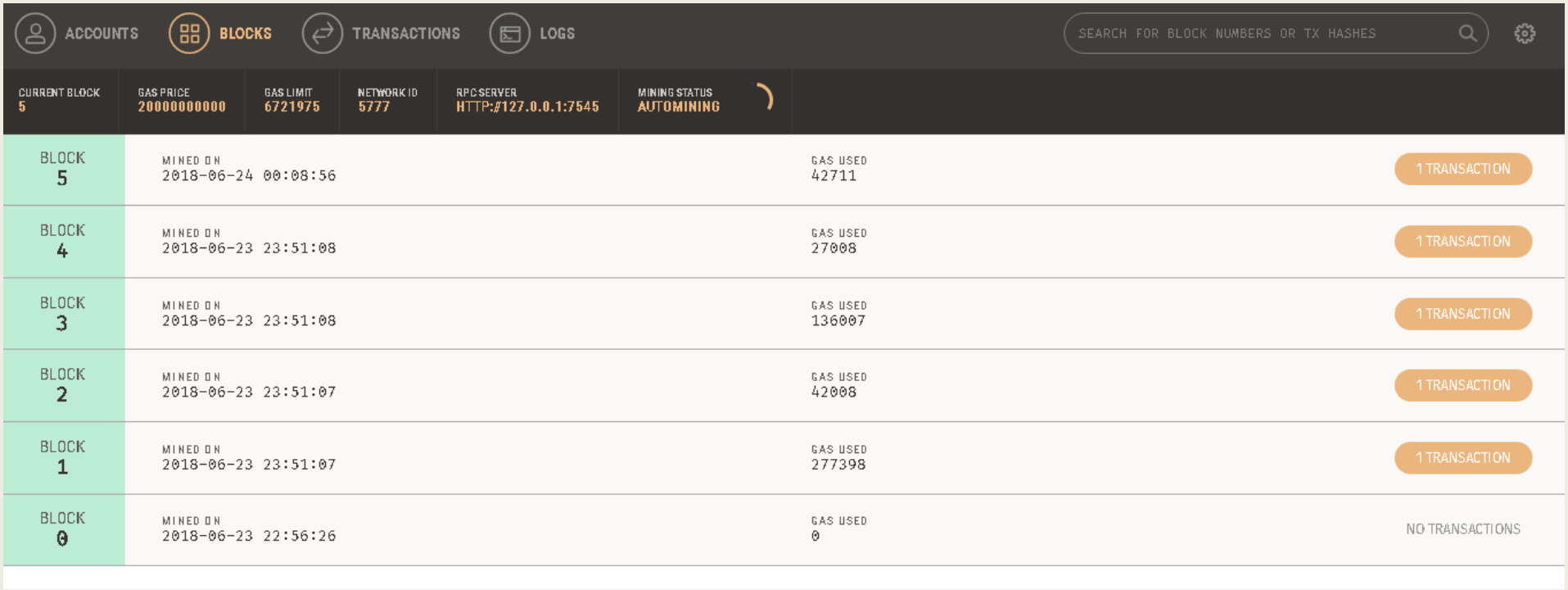
  var message;

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

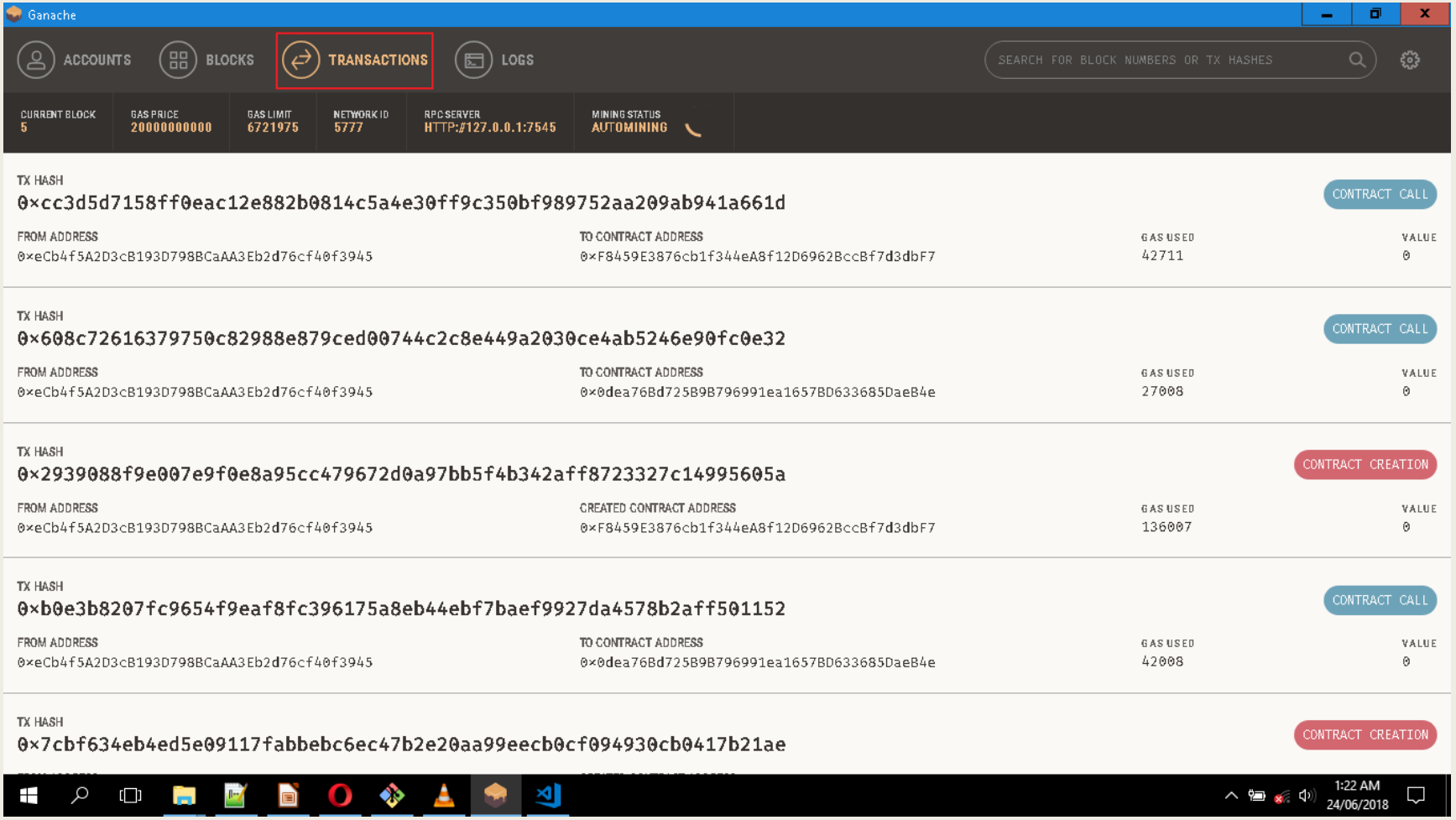
  message.setMessage(web3.fromAscii('My First Message!'));

Use the Powershell Console

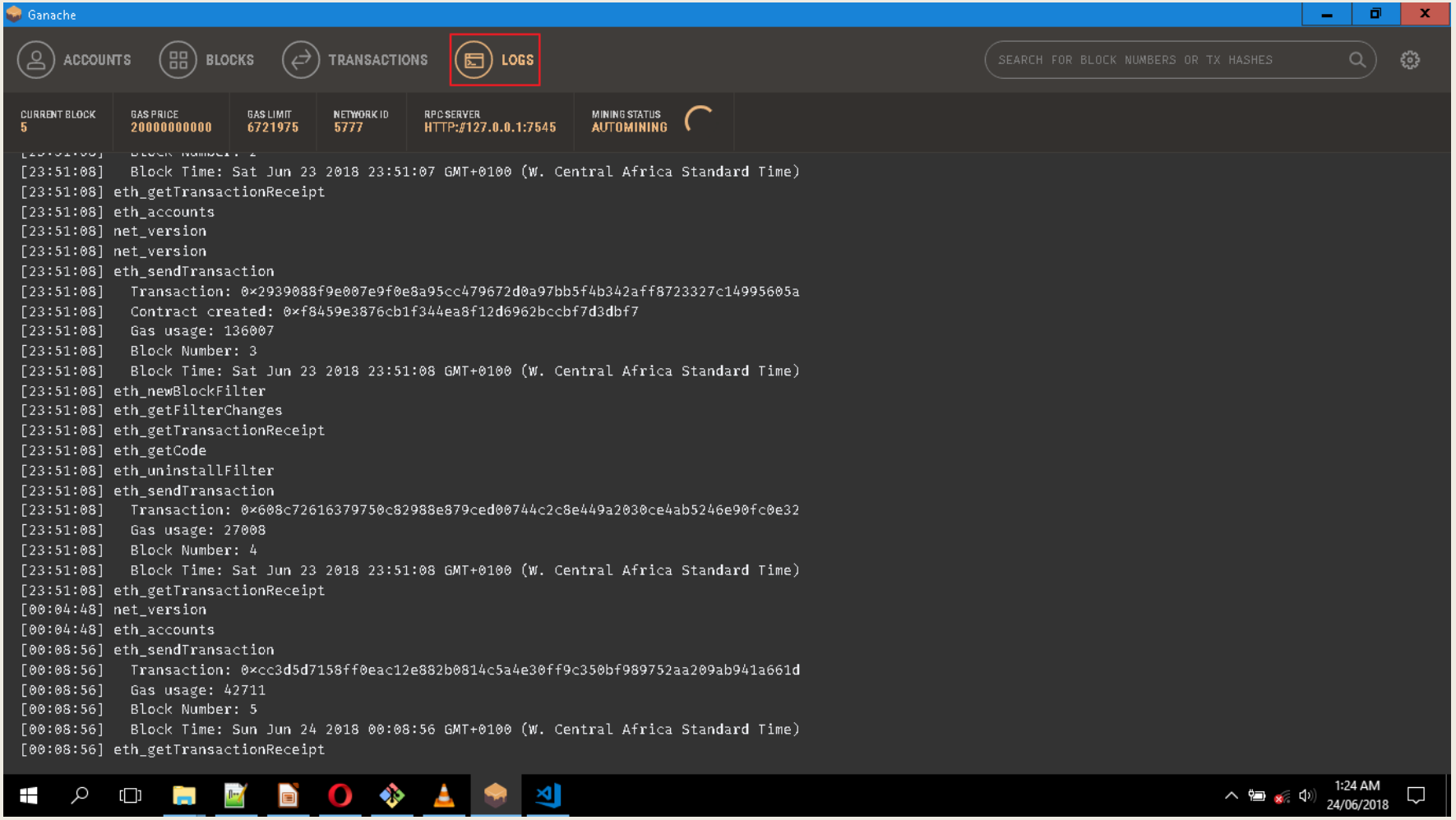
A New Block is Added



Check Your Transactions



Check Your Logs

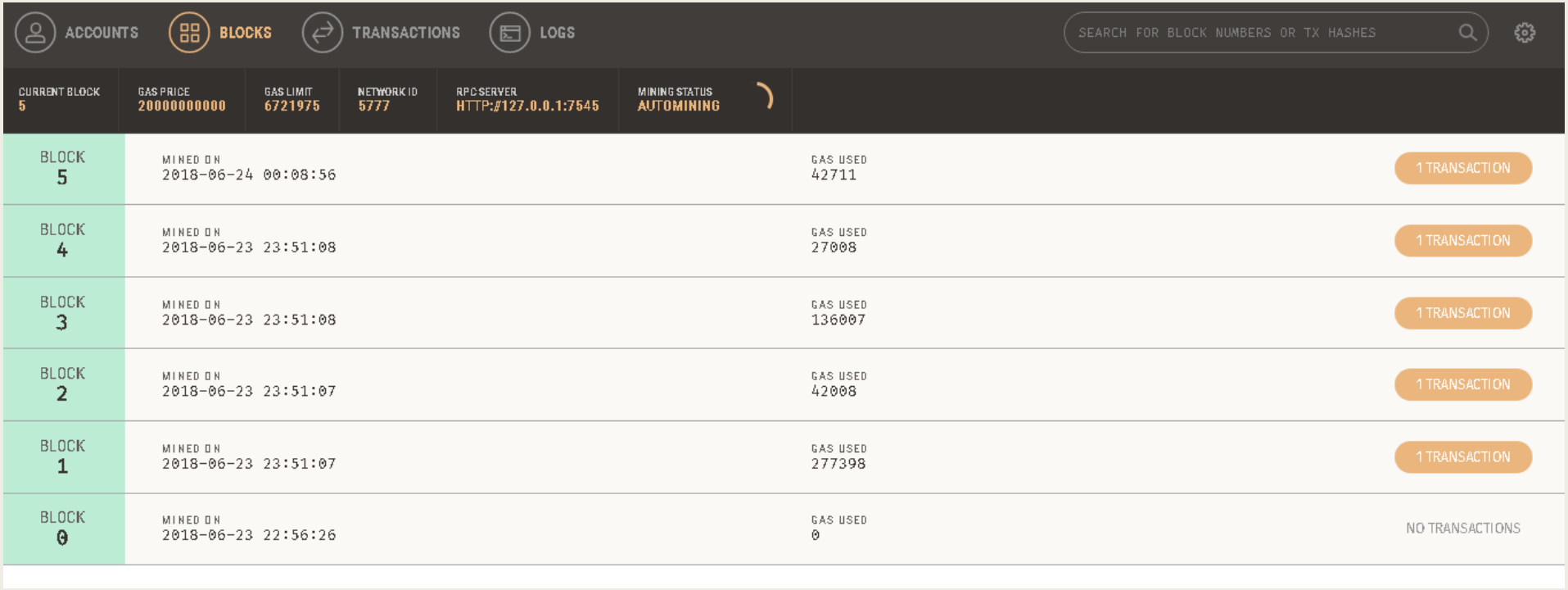


Get Your Message

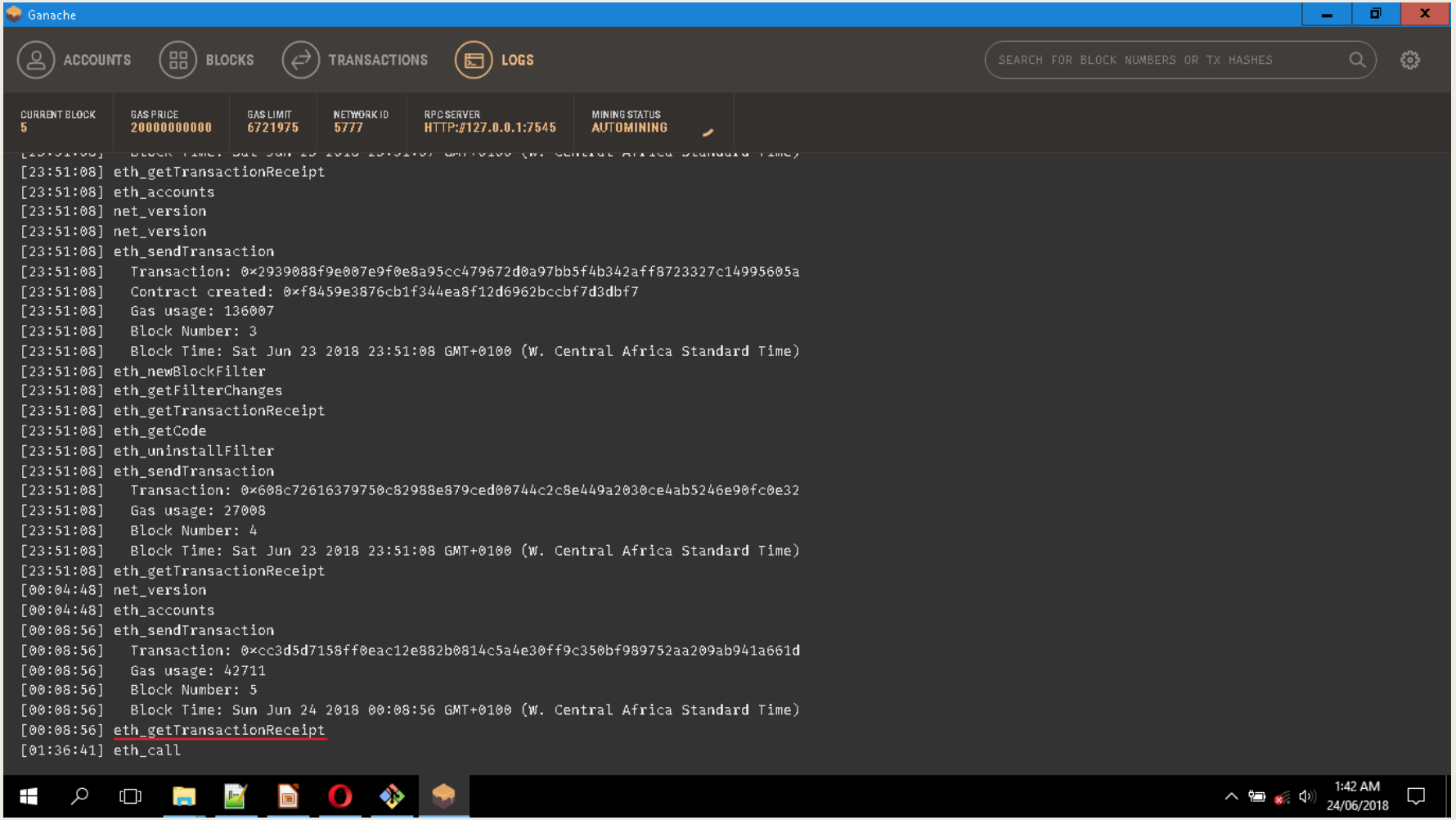
- To get your message from the console, enter the command shown below:

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

Check Your Blocks



Check Your Logs



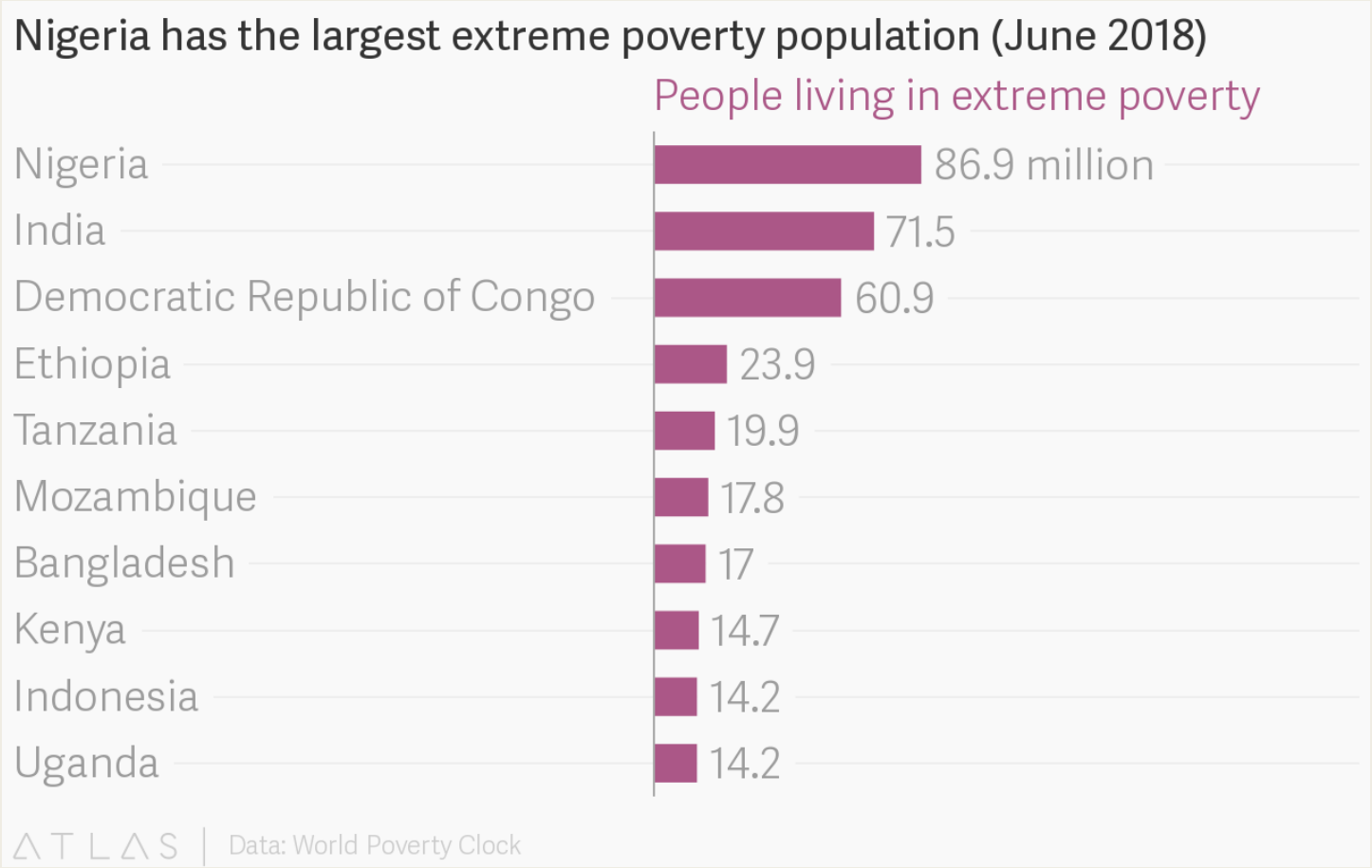
Conclusion

- 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 to 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



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