Solidity Pt 2

News

1. Utah has a bill to exempt blockchain from money transmitter act. This passed in Wyoming last year.
2. Ethereum has set a goal of 2 second transactions for this year.
3. Marshall Islands to Issue Physical Banknotes for World's First Decentralized National Digital Currency.
4. South Korea is using blockchain for government documents and validation.

Channels

A really good explanation : https://golangbot.com/channels/

Lecture Notes

Software to Install (node, truffle etc.)

1. npm Download and install `node.js` from https://nodejs.org/en/download/

2. truffle You should be able to:

```
    npm install -g truffle
```

3. ganache

```
    npm install -g ganache-cli
```

4. solc (solidity)

Just for entertainment there is a 'solc' compiler and a 'solc-js' compiler. The 'solc-js' compile has different options. We will want to use the 'solc' compiler.

On Windows the general consensus is to setup the Linux Subsystem for Windows. https://medium.com/@m_mcclarty/setting-up-solidity-on-windows-10-993a1d2c615c

Or for Windows

https://github.com/ethereum/solidity/releases

Download the Windows binary from https://github.com/ethereum/solidity/releases

Extract the `solidity-windows.zip` into a new folder.

Launch a command prompt and cd into the directory where `solc.exe` was extracted to.

Move `solc.exe` to a suitable directory to run it form. Usually I have a `bin` directory in my loign home with a path set in the environment that includes `bin` .

On Mac

```
        brew update
        brew upgrade
        brew tap ethereum/ethereum
        brew install solidity
```

Notes: 1. https://truffleframework.com/docs/truffle/getting-started/installation

```
$ truffle develop
Truffle Develop started at http://127.0.0.1:9545/

Accounts:
(0) 0x627306090abab3a6e1400e9345bc60c78a8bef57
(1) 0xf17f52151ebef6c7334fad080c5704d77216b732
(2) 0xc5fdf4076b8f3a5357c5e395ab970b5b54098fef
(3) 0x821aea9a577a9b44299b9c15c88cf3087f3b5544
(4) 0x0d1d4e623d10f9fba5db95830f7d3839406c6af2
(5) 0x2932b7a2355d6fecc4b5c0b6bd44cc31df247a2e
(6) 0x2191ef87e392377ec08e7c08eb105ef5448eced5
(7) 0x0f4f2ac550a1b4e2280d04c21cea7ebd822934b5
(8) 0x6330a553fc93768f612722bb8c2ec78ac90b3bbc
(9) 0x5aeda56215b167893e80b4fe645ba6d5bab767de

Private Keys:
(0) c87509a1c067bbde78beb793e6fa76530b6382a4c0241e5e4a9ec0a0f44dc0d3
(1) ae6ae8e5ccbfb04590405997ee2d52d2b330726137b875053c36d94e974d162f
(2) 0dbbe8e4ae425a6d2687f1a7e3ba17bc98c673636790f1b8ad91193c05875ef1
(3) c88b703fb08cbea894b6aeff5a544fb92e78a18e19814cd85da83b71f772aa6c
(4) 388c684f0ba1ef5017716adb5d21a053ea8e90277d0868337519f97bede61418
(5) 659cbb0e2411a44db63778987b1e22153c086a95eb6b18bdf89de078917abc63
(6) 82d052c865f5763aad42add438569276c00d3d88a2d062d36b2bae914d58b8c8
```

```
(7) aa3680d5d48a8283413f7a108367c7299ca73f553735860a87b08f39395618b7
(8) 0f62d96d6675f32685bbdb8ac13cda7c23436f63efbb9d07700d8669ff12b7c4
(9) 8d5366123cb560bb606379f90a0bfd4769eecc0557f1b362dcae9012b548b1e5

Mnemonic: candy maple cake sugar pudding cream honey rich smooth crumble sweet treat

⚠  Important ⚠  : This mnemonic was created for you by Truffle. It is not secure.
Ensure you do not use it on production blockchains, or else you risk losing funds.

truffle(develop)>
```

This will be the basis of homework 07 (100 pts - install and setup truffle). Due in about 4 weeks.

Setup example contract

Download and Setup an example set of contracts.

```
        $ mkdir MetaCoin
        $ cd MetaCoin
        $ truffle unbox metacoin
```

Output will be (should be):

```
Downloading...
Unpacking...
Setting up...
Unbox successful. Sweet!

Commands:

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

You should have a directory tree that looks like:

```
.
├── LICENSE
├── contracts
│   ├── ConvertLib.sol
│   ├── MetaCoin.sol
│   └── Migrations.sol
├── migrations
│   ├── 1_initial_migration.js
│   └── 2_deploy_contracts.js
│
```

```
├── test
│   ├── TestMetacoin.sol
│   └── metacoin.js
└── truffle-config.js
```

Now you should be able to test the contracts you have downloadd with:

```
        $ truffle test
```

And the output should be:

```
Compiling ./contracts/ConvertLib.sol...
Compiling ./contracts/MetaCoin.sol...
Compiling ./contracts/Migrations.sol...
Compiling ./test/TestMetacoin.sol...
Compiling truffle/Assert.sol...
Compiling truffle/DeployedAddresses.sol...


  TestMetacoin
    ✓ testInitialBalanceUsingDeployedContract (92ms)
    ✓ testInitialBalanceWithNewMetaCoin (177ms)

  Contract: MetaCoin
    ✓ should put 10000 MetaCoin in the first account
    ✓ should call a function that depends on a linked library (39ms)
    ✓ should send coin correctly (163ms)


  5 passing (1s)
```

Now install Open Zeppelin

```
        $ npm install openzeppelin-solidity
```