

2018 CDSE Workshop: Blockchain Programming: A Hands-on Tutorial

by Bina Ramamurthy April 12, 2018

bina@buffalo.edu

Instructions for hands-on components: You can get the best out of this workshop if you prepare ahead and follow along as instructed in this document.

Introduction:

Blockchain is an emerging area with extraordinary interest from wide variety of people and industries. There is a lot to learn. In this hands-on tutorial we will cover enough to get you started on programming for the blockchain.

Learning outcomes:

On Completion of this workshop, you will be able to

- (i) Design, code a Solidity smart contract for Ethereum blockchain and test the smart contract on Remix Web IDE (Integrated Development Environment)
- (ii) Design and develop a Dapp (decentralized app) on Truffle IDE, deploy it on an Ethereum test chain and test it using web interface and Metamask wallet.

****Preparation before the workshop:****

1. Copy from <https://tinyurl.com/y7fzs2p4> the two Solidity code files (Greeter.sol, Coin.sol) into your laptop. Save it in a location/directory where you can access it during the workshop.
 2. Make sure you are able to access the **Remix IDE**: <http://remix.ethereum.org>
 3. Download and install Virtual box on your laptop. <https://www.virtualbox.org/wiki/Downloads>
 4. Download Virtual Machine image from this location: <https://tinyurl.com/y7fzs2p4>
 5. Start the virtual box by double clicking on it. Import the virtual image you downloaded in Step 3. File→ import appliance..
-

We will work on the following steps during the workshop. Do NOT try them now.

6. Now are all set to develop and deploy Ethereum Blockchain based application using Truffle IDE.
7. These are the commands for the **truffle IDE**: (it has been preloaded the VM image)
To compile code: `truffle compile`
To deploy a test chain: `truffle develop`
(Copy the seed word somewhere: Observe the accounts and addresses created)
To deploy a smart contract: `truffle migrate --reset`
To test a smart contract : `truffle test`
To run a lite web server: `npm run dev`
8. To **install Metamask** plugin to Chrome browser:
Make sure you have a Chrome browser installed.. Others requires a little bit more work.
In the truffle default port for the test chain deployed by “truffle develop” command is 9545.
The account addresses are exposed through this port. Lets now link Metamask to manage it for our Dapp. You should have done Truffle Develop before starting the following steps. Copy the seed words listed by the test chain: it starts with “candy”..

1. Start the Chrome browser
2. On the right top, if there is Metamask plugin already present, left click on it and remove it. This is to ensure you a clean environment for testing your Dapp.
3. Open Metamask.io web site, click on “get chrome extension”, click on Add to Chrome button on the next screen. There is one button to add before you will see the Metamask icon in the corner of your browser.
4. Right click on the icon, click on Accept button, scroll through all the way and then Accept.
5. (Be patient) Now you will see the Metamask interface for setting up the connection. Click on Import Existing Den at the bottom of the interface.
6. In the next screen, paste seed words you copied, enter a new password, repeat the password. Then click OK button.
7. In the next interface, at the top, you will see Main Network. Click drop down arrow, click custom RPC.
8. In the next interface that appears, enter `http://localhost:9545`, click save.
9. Click on left arrow to return to your metamask wallet interface.
10. You will see Account 1, with 99.690 Ethers. Every account was given 100 Ethers, but part of account 1's was used up when deploying our smart contract.
11. Now you are ready to test the complete Dapp.
12. Explore the various controls and displays on Metamask before you proceed. Explore how you can create (or connect to) a new account, for example.

Interact and understand the working of the Dapp and the Dapp stack.