**Blockchain Developers Course**

**COURSE OVERVIEW**

This course in a course designed for developers who are seeking to learn blockchain from a development perspective. It is an 6-week full time program in blockchain from basics to advanced. Delivered in a live-classroom setting. This course will provide the participants with the ability to implement powerful blockchain applications. Participants will be spending large amounts of time creating and developing real blockchain tools.

The course will provide working sandboxes for all of the coding that is developed.

**ADMISSIONS STANDARDS**

Basics of Computers  
Basics of Programming  
1+ years of programming experience (validated by CV or certification)   
JavaScript / Python programming is a PLUS  
Entrance Exam

**AT THE END OF THIS COURSE YOU WILL BE ABLE TO**

Work with Blockchain technology on a practical level  
Build real-life tools based upon The Blockchain.  
Have validated code to share with prospective employers showing a working knowledge of blockchain programming.

**ACCREDITATION**

Accreditation is given to participants who pass with 70% score on all course examinations and successful completion of all of the coding assignments.

**SCHEDULE**

**PRICING**

**Blockchain Developers Course Outline**

**Week 1: Blockchain Foundations**

Origins of Blockchain

Disintermediation?

How Bitcoin Works

Private vs. Public Blockchains

Permissioned vs. Permissionless

Setting up working Environment

Attacks on Blockchain?

**Week 2: Bitcoin protocol**

Bits and Bytes – Manually counting and converting

Hex + ASCII – working with JS/ Python and buffers

Connecting to Peers – reading the bitcoin protocol

Protocol messaging tools

Breaking it to payload and header

Creating basic messages manually and with JS/ Python

Protocol messaging tools continued…

Version Payload – applying protocol rules to create real messages

Hash – one way functions, introduction and JS/Python

Headers and version headers – creating the template for bitcoin message headers

TCP dump + Wireshark – the basics of packet sniffing

**Week 3: Keys & Transactions**

Keys – creating a pair of keys

Signing messages and encryption with Key Pair

Private key to a bitcoin address

Mathematically background to Keys

From private keys to bitcoin address

Transactions – Segwit, filling fields according to protocol rules (JS/ Python) UTXOs and inputs

Scripts – creating scripts for variety of transactions (P2PKH, OP\_RETURN, P2SH)

Writing one mock-script

Stack architecture

Modifying mock-scripts to support stacks

OP\_CODES

Hashing Transactions

Signing Transactions

**Week 4: The blockchain**

Individual Blocks

Consensus rules

Coinbase transactions

Merkle trees, light clients

POW – Proof of Work, calculating difficulty/target, block-headers and nonce

Segwit, lightning network, merged mining and side chains

**Week 5: Ethereum & Smart Contracts**

Moving to Ethereum – Basic Concepts

Ethereum EVM

Solidity – concepts in smart contract architecture

Solidity – variables, arrays and memory structure

Solidity – basic functions, modifiers, events and constructor functions

Solidity – contract inheritance

Solidity – relations between smart contacts

Solidity – calls, delegations, libraries and upgradeable contracts

**Week 6: Security and Standards**

Solidity – advanced concepts

Truffle

testrpc

Web3.js– basic concepts and commands

Ethereum and bitcoin standardization

BIPS, ERC 20

Offline transactions

**CONTACT INFORMATION:**

Bryant Nielson, Executive Director  
[bryant.nielson@capitalwave.com](mailto:bryant.nielson@capitalwave.com)

+ (212) 335-0097

<http://www.TheBlockchainAcademy.com>