Lab 10 — Blockchain

- Study the GitHub repository Lesson 10
- Run hash_value.py twice and compare results
- Run snakecoin.py
- Run snakecoin-server-full-code.py on Terminal 1 and mine a new block on Terminal 2
- Clone Python blockchain app and uncomment the last line of node_server.py
- Run node_server.py on Terminal 1 and run_app.py on Terminal 2
- Install pyota[ccurl]
- Run iri_node_info.py

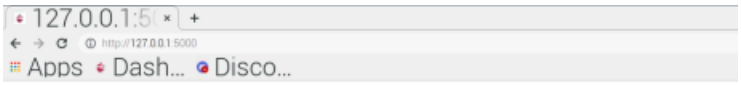
```
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ python3 hash_value.py
The hash for 1 is: 1
The hash for 1.0 is: 1
The hash for 3.14 is: 1846836513
The hash for Python is: 1893820470
The hash for a tuple of vowels is: 1241348522
The hash for an object of person is: -111852159
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ python3 hash_value.py
The hash for 1 is: 1
The hash for 1.0 is: 1
The hash for 3.14 is: 1846836513
The hash for Python is: -1783306572
The hash for a tuple of vowels is: -444620313
The hash for an object of person is: -2051330152
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $
```

Build the tiniest blockchain in less than 50 lines of Python by Gerald Nash

```
>>> import hashlib
>>> m = hashlib.sha256(b"hello, world")
>>> m.hexdigest()
'09ca7e4eaa6e8ae9c7d261167129184883644d07dfba7cbfbc4c8a2e08360d5b'
>>> m.digest_size
32
>>> m.block_size
64
>>> exit()
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $
```

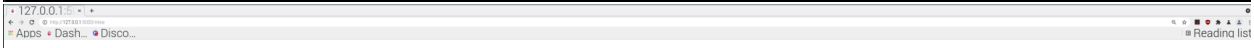
```
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ python3 snakecoin.py
Block #1 has been added to the blockchain!
Hash: 39918108313764972aedd0205c8caa13c2d0dd75ad470000db0b5a102349aa9d

Block #2 has been added to the blockchain!
Hash: e254b7149561f0262eb7c09363c90addfd93c20b2eb900fbed7fd1b657504ef7

Block #3 has been added to the blockchain!
Hash: b1d99ec35408b94aa92598cf90571334714939b940ead6ddbd5b0919c1d51866

Block #4 has been added to the blockchain!
Hash: 81b622c08640848d095fb88870a5349c31272065c466d4d4fe483156162bd2bd

Block #5 has been added to the blockchain!
Hash: 4eb7a1d5850db2bb5a83f80bf936506ae4160cd71495c770f418007a39b2f2d0

Block #6 has been added to the blockchain!
Hash: 058feeb44b7f2dd49ee402954e3f68cf894fee273e13fe33c8741cdebd4598da

Block #7 has been added to the blockchain!
Hash: 767fa00cd80c7049ebc56c5516b96b7746251835d17d4f5e8b4aee773e5d6294

Block #8 has been added to the blockchain!
Hash: 7e11ca84abcee2d9caa18b63d8c309c4e7d5535f45ce49a23f38a8439244ec59
```

```
node.run()
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ python3 snakecoin-server-full-code.py
 * Serving Flask app "snakecoin-server-full-code" (lazy loading)
 * Environment: production
   WARNING: Do not use the development server in a production environment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [01/May/2022 12:42:39] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [01/May/2022 12:42:40] "GET /favicon.ico HTTP/1.1" 200 -
```



SnakeCoin Server

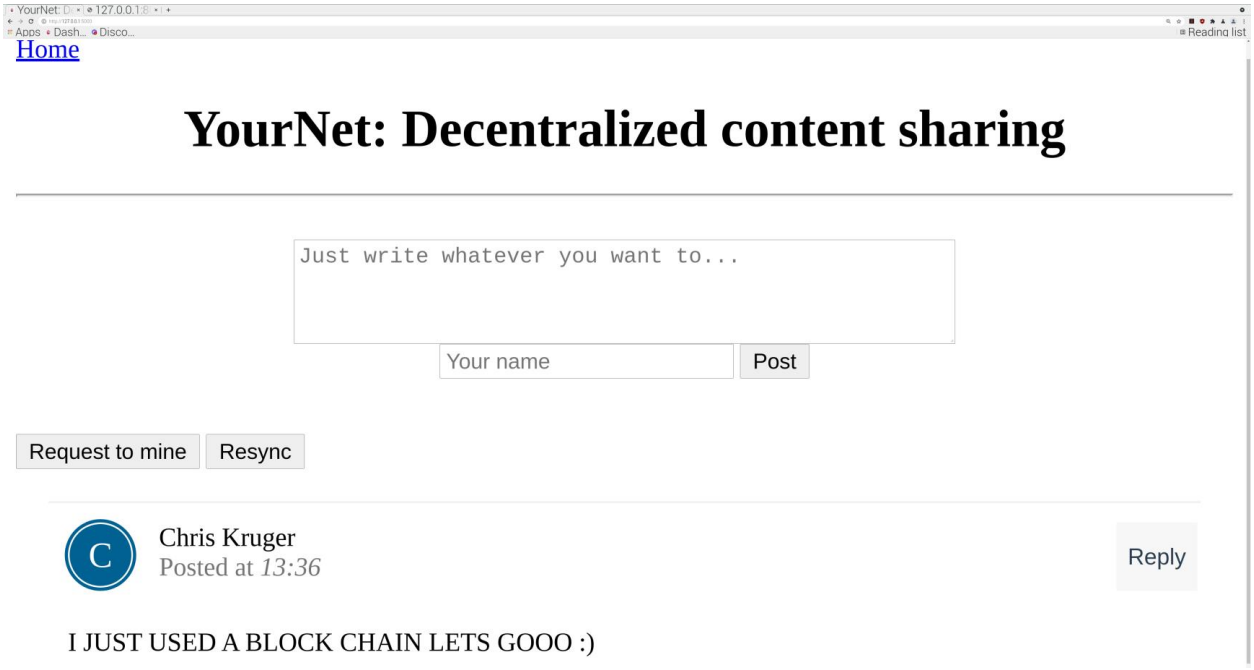
```
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ curl "localhost:5000/txion" \
>       -H "Content-Type: application/json" \
>       -d '{"from": "akjflw", "to":"fjlakdj", "amount": 3}'
Transaction submission successful
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ ccurl "localhost:5000/txion" \
>       -H "Content-Type: application/json" \
> ^C
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $ curl localhost:5000/mine
{"index": 1, "timestamp": "2022-05-01 12:44:07.176931", "data": {"proof-of-work": 18, "transactions": [{"from": "akjflw
", "to": "fjlakdj", "amount": 3}, {"from": "network", "to": "q3nf394hjg-random-miner-address-34nf3i4nflkn3oi", "amount"
: 1}]}, "hash": "b489693ff5f82162bf91973719498378e7a9f31ae9c2db4fa20b473d25e9d22a"}
pi@raspberrypi:~/Desktop/CPE322iot/iot/lesson10 $
```



{"index": 2, "timestamp": "2022-05-01 12:44:34.032072", "data": {"proof-of-work": 36, "transactions": [{"from": "network", "to": "q3nf394hjg-random-miner-address-34nf3i4nflkn3oi", "amount": 1}]}, "hash": "aeecf12123ab9bd3b798fb170b494376da5a66a5e0bfc4834da2d6bfc807872a"}

```
pi@raspberrypi:~ $ cd python_blockchain_app/
pi@raspberrypi:~/python_blockchain_app $ ls
app  CONTRIBUTING.md  node_server.py  README.md  requirements.txt  run_app.py  screenshots
pi@raspberrypi:~/python_blockchain_app $ nano node_server.py
pi@raspberrypi:~/python_blockchain_app $ nano node_server.py
pi@raspberrypi:~/python_blockchain_app $ python3 node_server.py
 * Serving Flask app "node_server" (lazy loading)
 * Environment: production
   WARNING: Do not use the development server in a production environment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:8000/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 301-184-290
```

Block #1 is mined.


YourNet: D
127.0.0.1
Apps
Dash...
Disco...
Reading list
Home
YourNet: Decentralized content sharing
Just write whatever you want to...
Your name
Post
Request to mine
Resync
C
Chris Kruger
Posted at 13:36
Reply
I JUST USED A BLOCK CHAIN LETS GOOO :)