Read Me

Requirements

To run the app ideally install Geth (https://geth.ethereum.org/docs/getting-started).

Also have a version of eclipse installed (https://www.eclipse.org/).

Use Git to clone the repository from “https://github.com/Rayah94/pv-extension”.

Starting the blockchain and adding funds

First open a command window in windows and run “geth --datadir test-chain-dir --rpc --dev --rpcapi "eth,net,web3,personal,miner"”. This will start a local blockchain.

Then make sure that every account that should run has enough funds. Adding funds can be achieved by opening another command prompt and run “geth attach “http://localhost:8545” this will open a javascript command prompt from which commands can be sent to the blockchain.

Now run “eth.sendTransaction({from:eth.coinbase, to:"**XXX**", value: web3.toWei(1, "ether")})”. **XXX** has to be replaced by the account address to which 1 ether should be sent.

The account addresses for the 11 mnemonic in the code are:

0. 0x2fe9f56859cf9f71db24ca975ebb43dc8a551b12
1. 0x7199d0a4b96d83250e86f0e09e8b2007caa14bd9
2. 0xf879ca17be1c3f696a7feb5845f78055163e02f8
3. 0x15fd708972fa9af91f86f0c36f7930f1f234730d
4. 0x42158522f0059205c09a9bb603199b37e38a2b2f
5. 0xa70ed10efa1465c4f8ae2e5421866bf9a0546b0d
6. 0x5afb97afd1be809a2ec719e90b9cec6b4a99cb7c
7. 0x5a48f615d3bc3b27323f9e0354f85669e18a1a7c
8. 0x21295ecbfac2e401b8cc13251be1966b081ad51b
9. 0x21eb4dc8c65ccf10f235961bbc4565c442e3aee1
10. 0xdf080fee0de40c24181b8e523923b1d7f83652de

Import the project

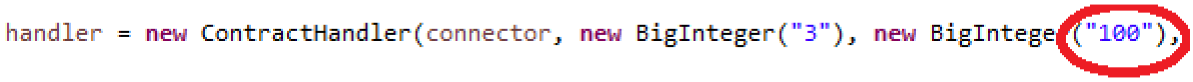
To import the project in eclipse go to **File** -> **Import** -> **Maven** -> **Existing Maven Projects** then select the root directory “pv-extension” and click “Finnish”.

Now right click the root folder in eclipse and go to **Run As** -> **Maven generate-sources**. This will recreate the smart contract from the PV-Contract.sol file.

Run the app

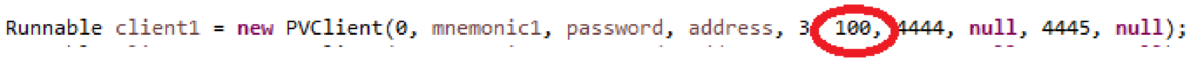
Right click on the “Main.java” file in the “testApplication” package and go to **Run As** -> **Java Application**.

To change the the amount of sessions the clients should run change the “100” in the BigInteger of line 36 to a value greater or equal to 2



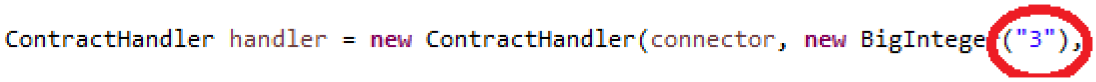
```
handler = new ContractHandler(connector, new BigInteger("3"), new BigIntege("100"),
```

Also change the 100 in every client that runs to this number:



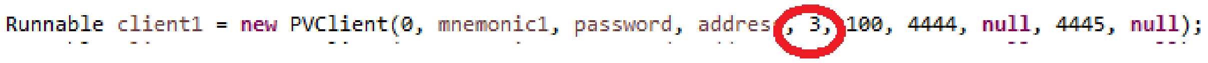
```
Runnable client1 = new PVClient(0, mnemonic1, password, address, 3, 100, 4444, null, 4445, null);
```

To change the number of clients alter the “3” in line 36 to a value greater or equal to 2



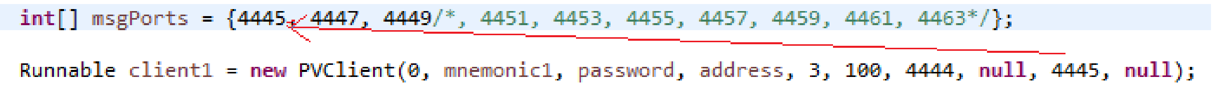
```
ContractHandler handler = new ContractHandler(connector, new BigInteger("3"),
```

And also change the number for every client



```
Runnable client1 = new PVClient(0, mnemonic1, password, address, 3, 100, 4444, null, 4445, null);
```

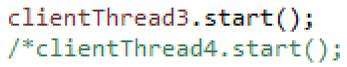
You also need to add every additional client to the msgPorts (the uneven numbers):



```
int[] msgPorts = {4445, 4447, 4449/*, 4451, 4453, 4455, 4457, 4459, 4461, 4463*/};

Runnable client1 = new PVClient(0, mnemonic1, password, address, 3, 100, 4444, null, 4445, null);
```

And also include starting the thread of the client:



```
clientThread3.start();
/*clientThread4.start();
```

Add more clients

To add more clients simply add a new mnemonic and fund the resulting account. Then initialize a new Runnable with this mnemonic and add ports for the PVServer and the MsgServer. Add the MsgServer port also into the msgPorts array.

After that build a new Thread with the Runnable and start it with the other client Threads.