Bhartiya Vidya Bhavan’s
Sardar Patel Institute of Technology, Mumbai-400058
Department of Electronics and Telecommunication Engineering
IT424:Blockchain Technology and Applications

Lab-6: Ethereum Blockchain Part-I
Setup an Ethereum Private Blockchain

Name:Pankaj Pandurang Gaikwad Class:TEIT UID:2019140019

Objective: Setup an Ethereum Blockchain
Outcomes: After successful completion of lab students should be able to

1. Implement an Ethereum private blockchain
2. Create the genesis block
3. Start the Ethereum blockchain
4. Create an account on the blockchain
5. Transact Ethers on blockchain
6. Query the blockchain using Geth- Geth console, Geth attach, Geth JSON RPC
7. Use Eth and Web3 interface over RPC

System Requirements:
PC (C2D, 4GB RAM, 100GB HDD space and NIC),Ubuntu Linux 14.04/20.04
Internet connectivity,Python Cryptography and Pycrypto,REST API,Go Lanaguage
Go Ethereum

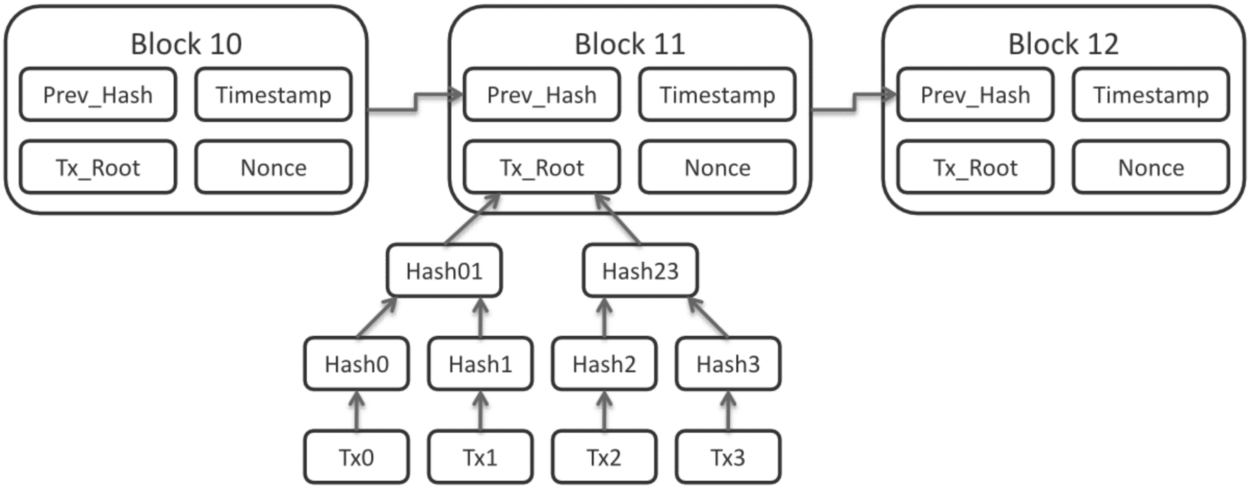


Figure-1: Blockchain Implementation

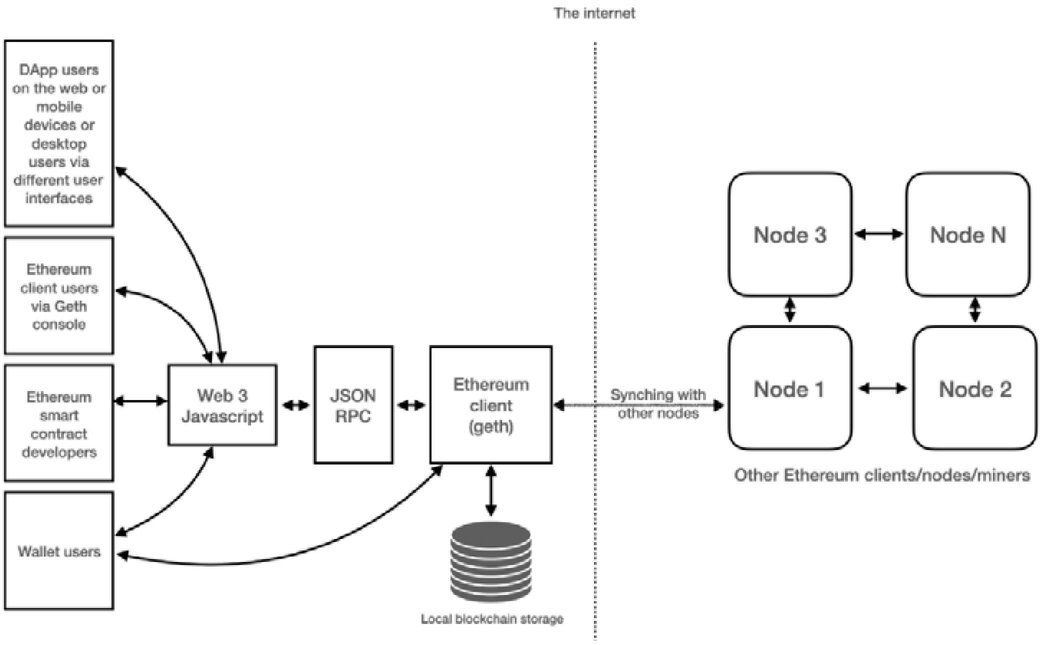

The internet
DApp users on the web or mobile devices or desktop users via different user interfaces
Ethereum client users via Geth console
Ethereum smart contract developers
Wallet users
Web 3 Javascript
JSON RPC
Ethereum client (geth)
Local blockchain storage
Synching with other nodes
Node 3
Node N
Node 1
Node 2
Other Ethereum clients/nodes/miners

Figure-2: Ethereum high-level ecosystem

About Ethereum Blockchain: Ethereum is an open-source, public,blockchain-based distributed computing platform. It features smart contract (scripting) functionality, which facilitates online contractual agreements. The Ethereum elements include:

- Blocks and blockchain
- Wallets and client software
- Nodes and miners
- APIs and tools
- Supporting protocols
- Programming languages

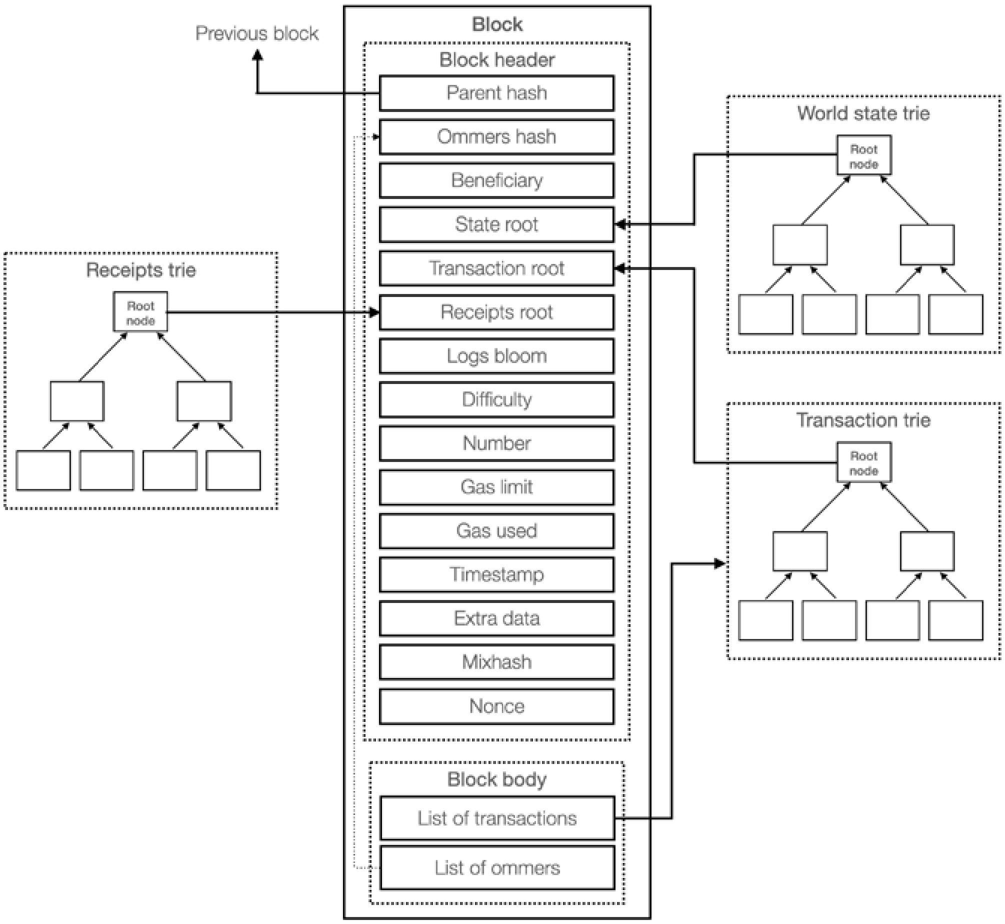


Figure-3: A detailed diagram of the block structure with a block header and relationship with tries

Blocks and blockchain

Blocks are the main building structure of a blockchain. Ethereum blocks consist of various elements, which are described as follows:

- The block header

- The transactions list

- The list of headers of ommers or uncles

The transaction list is simply a list of all transactions included in the block. Also, the list of headers of uncles is also included in the block.

Block header: Block headers are the most critical and detailed components of an Ethereum block. The header contains various elements, which are described in detail here:

- **Parent hash:** This is the Keccak 256-bit hash of the parent (previous) block's header.

- **Ommers hash:** This is the Keccak 256-bit hash of the list of ommers (or uncles) blocks included in the block.

- **The beneficiary:** The beneficiary field contains the 160-bit address of the recipient that will receive the mining reward once the block is successfully mined.

- **State root:** The state root field contains the Keccak 256-bit hash of the root node of the state trie. It is calculated once all transactions have been processed and finalized.

- **Transactions root:** The transaction root is the Keccak 256-bit hash of the root node of the transaction trie. The transaction trie represents the list of transactions included in the block.

- **Receipts root:** The receipts root is the Keccak 256-bit hash of the root node of the transaction receipt trie. This trie is composed of receipts of all transactions included in the block. Transaction receipts are generated after each transaction is processed and contain useful post-transaction information. More details on transaction receipts are provided in the next section.

- **Logs bloom:** The logs bloom is a bloom filter that is composed of the logger address and log topics from the log entry of each transaction receipt of the included transaction list in the block. Logging is explained in detail in the next section.

- **Difficulty:** The difficulty level of the current block.

- **Number:** The total number of all previous blocks; the genesis block is block zero.

- **Gas limit:** This field contains the value that represents the limit set on the gas consumption per block.

- **Gas used:** This field contains the total gas consumed by the transactions included in theblock.

- **Timestamp:** The timestamp is the epoch Unix time of the time of block initialization.

Extra data: The extra data field can be used to store arbitrary data related to the block.Only up to 32 bytes are allowed in this field.

• **Mixhash:** The mixhash field contains a 256-bit hash that, once combined with the nonce, is used to prove that adequate computational effort (Proof of Work, or PoW) has been spent in order to create this block.

• **Nonce:** Nonce is a 64-bit hash (a number) that is used to prove, in combination with the mixhash field, that adequate computational effort (PoW) has been spent in order to create this block.

Problem Statement:

[1] Create autonomous private Blockchain with rules on spending money.

[2] Make a cryptocurrency with a fixed market market supply and tokens to represent real world asset values.

[3] Mine for new Ether by validating transactions.

Ethereum Blockchain Flow Diagram:

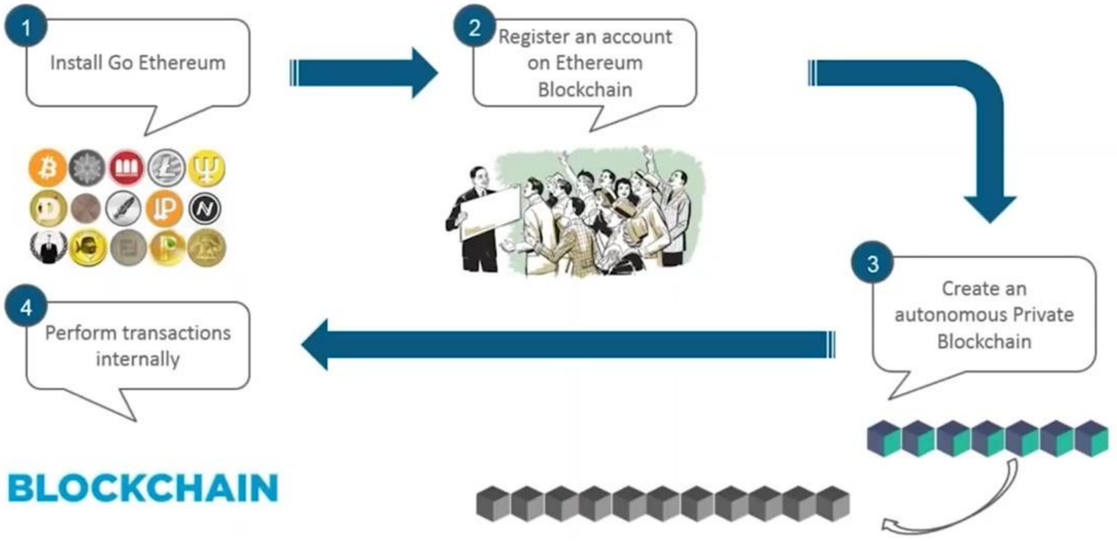

1
Install Go Ethereum
2
Register an account on Ethereum Blockchain
3
Create an autonomous Private Blockchain
4
Perform transactions internally
BLOCKCHAIN

Figure-4: Ethereum Blockchain Flow

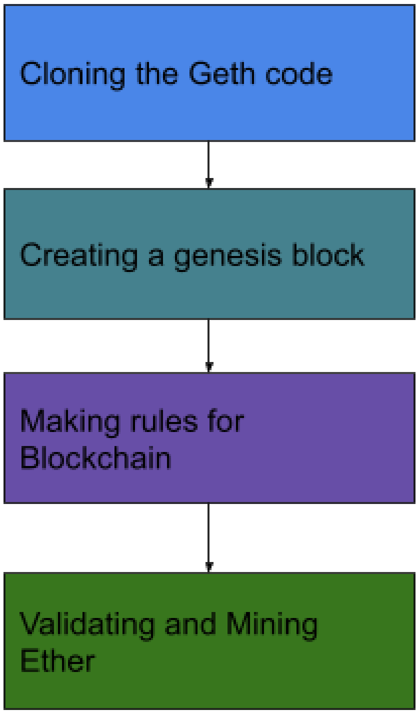


Figure-5: Ethereum Blockchain setup

Procedure:

[1] Install Ethereum Blockchain

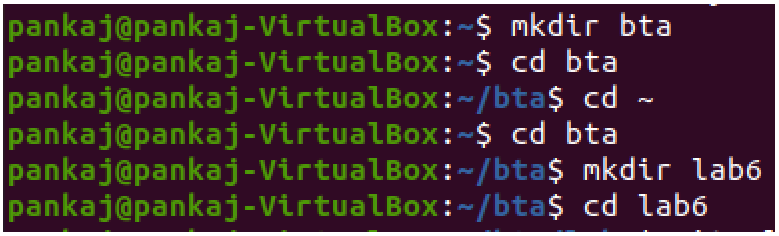
Clone it from git and compile it.

```
$cd ~

$mkdir BTA

$cd BTA
```

$mkdir lab6

$cd lab6

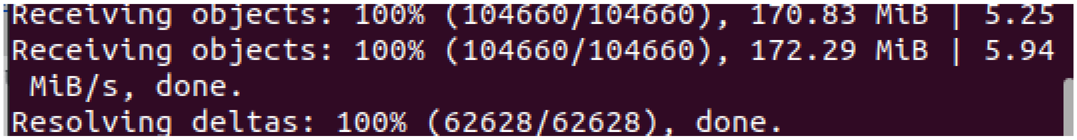
creating bta and lab6 directories



```
pankaj@pankaj-VirtualBox:~$ mkdir bta
pankaj@pankaj-VirtualBox:~$ cd bta
pankaj@pankaj-VirtualBox:~/bta$ cd ~
pankaj@pankaj-VirtualBox:~$ cd bta
pankaj@pankaj-VirtualBox:~/bta$ mkdir lab6
pankaj@pankaj-VirtualBox:~/bta$ cd lab6
```

$ git clone https://github.com/ethereum/go-ethereum

cloning go-ethereum:

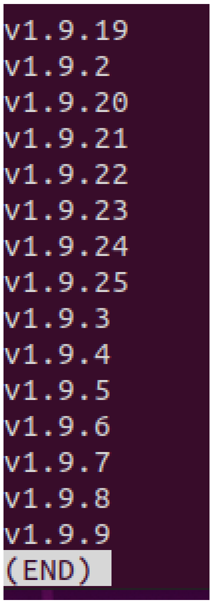


```
Receiving objects: 100% (104660/104660), 170.83 MiB | 5.25
Receiving objects: 100% (104660/104660), 172.29 MiB | 5.94
 MiB/s, done.
Resolving deltas: 100% (62628/62628), done.
```

$cd go-ethereum

$git tag

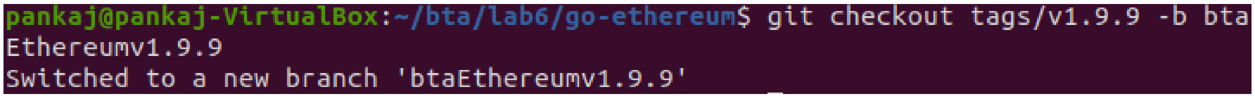
after git tagging is done:



```
v1.9.19
v1.9.2
v1.9.20
v1.9.21
v1.9.22
v1.9.23
v1.9.24
v1.9.25
v1.9.3
v1.9.4
v1.9.5
v1.9.6
v1.9.7
v1.9.8
v1.9.9
(END)
```

$git checkout tags/v1.9.9 -b btaEthereumv1.9.9

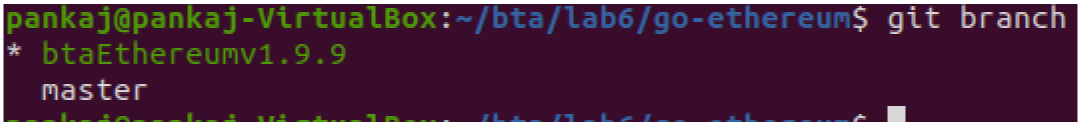
switched to new branch



```
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$ git checkout tags/v1.9.9 -b bta
Ethereumv1.9.9
Switched to a new branch 'btaEthereumv1.9.9'
```

#checking the branch

$git branch



```
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$ git branch
* btaEthereumv1.9.9
  master
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$
```

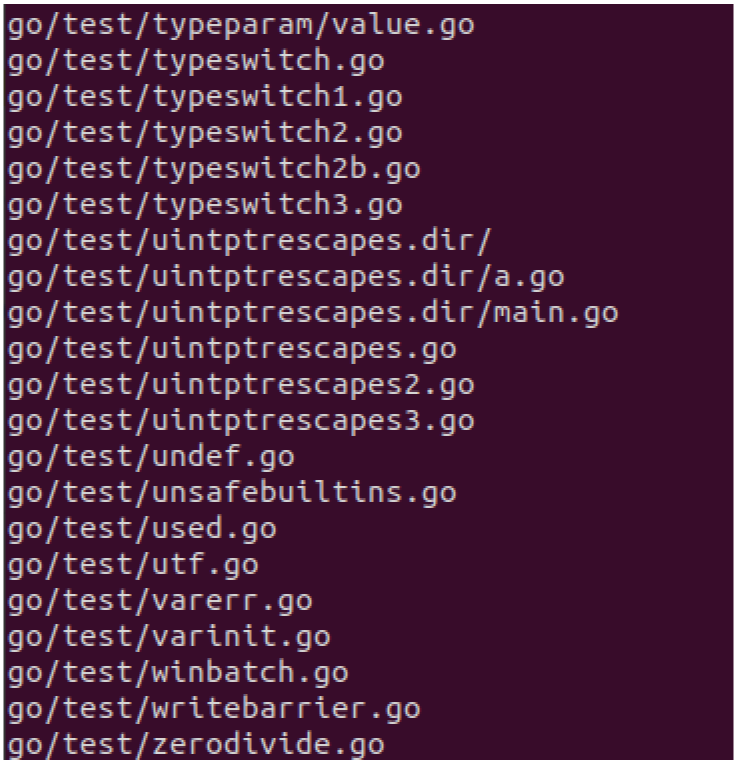
#Install golang

Download the golang from the official website of the x86-64 tarball image source.

go1.18.linux-amd64.tar.gz

$tar -xzvf go1.18.linux-amd64.tar.gz

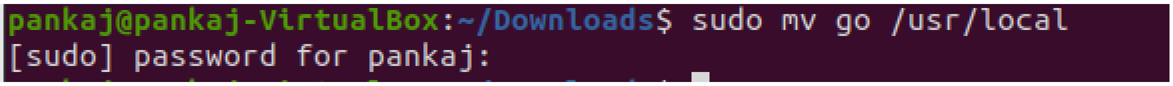
extractimg tar ball:



```
go/test/typeparam/value.go
go/test/typeswitch.go
go/test/typeswitch1.go
go/test/typeswitch2.go
go/test/typeswitch2b.go
go/test/typeswitch3.go
go/test/uintptrescapes.dir/
go/test/uintptrescapes.dir/a.go
go/test/uintptrescapes.dir/main.go
go/test/uintptrescapes.go
go/test/uintptrescapes2.go
go/test/uintptrescapes3.go
go/test/undef.go
go/test/unsafebuiltins.go
go/test/used.go
go/test/utf.go
go/test/varerr.go
go/test/varinit.go
go/test/winbatch.go
go/test/writebarrier.go
go/test/zerodivide.go
```

$sudo mv go /usr/local

moving extracted file to /usr/local



```
pankaj@pankaj-VirtualBox:~/Downloads$ sudo mv go /usr/local
[sudo] password for pankaj:
```

# Append these lines your .bashrc file and save it.

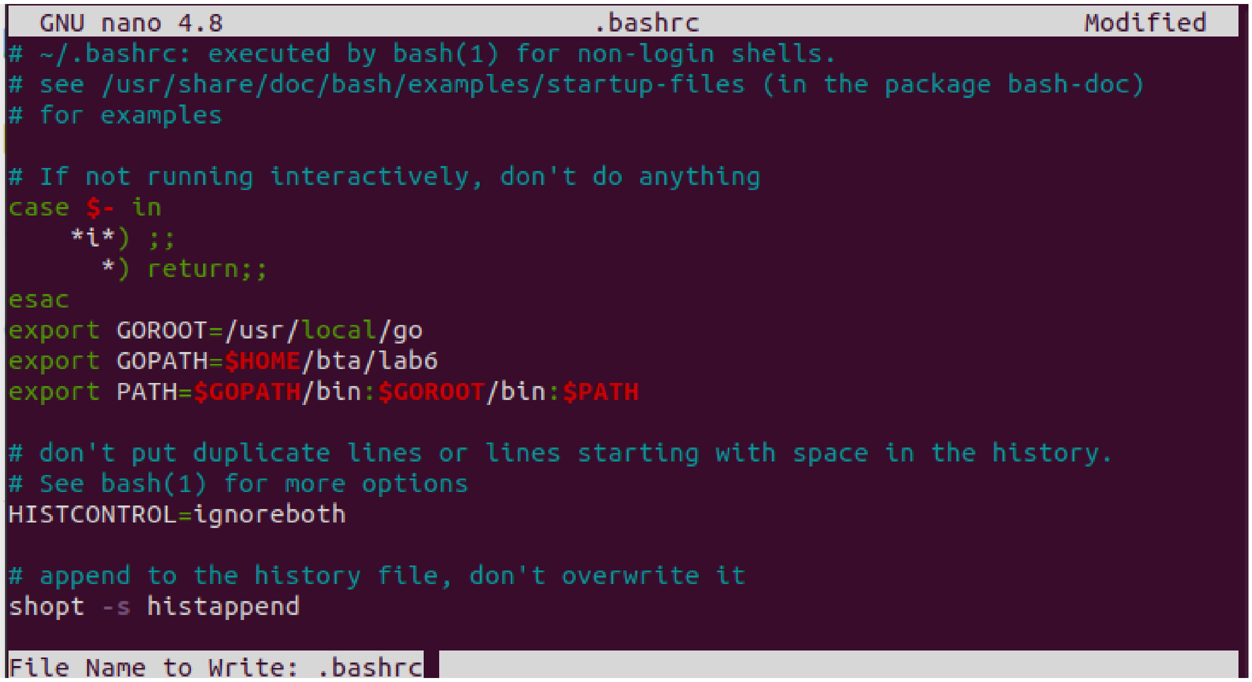
$cd ~

$nano .bashrc

export GOROOT=/usr/local/go

export GOPATH=$HOME/BTA/lab6A

export PATH=$GOPATH/bin:$GOROOT/bin:$PATH

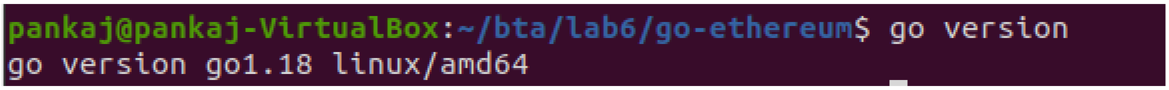


```
  GNU nano 4.8                         .bashrc                          Modified
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
      *) return;;
esac
export GOROOT=/usr/local/go
export GOPATH=$HOME/bta/lab6
export PATH=$GOPATH/bin:$GOROOT/bin:$PATH

# don't put duplicate lines or lines starting with space in the history.
# See bash(1) for more options
HISTCONTROL=ignoreboth

# append to the history file, don't overwrite it
shopt -s histappend

File Name to Write: .bashrc
```

$source .bashrc

#Check the go version

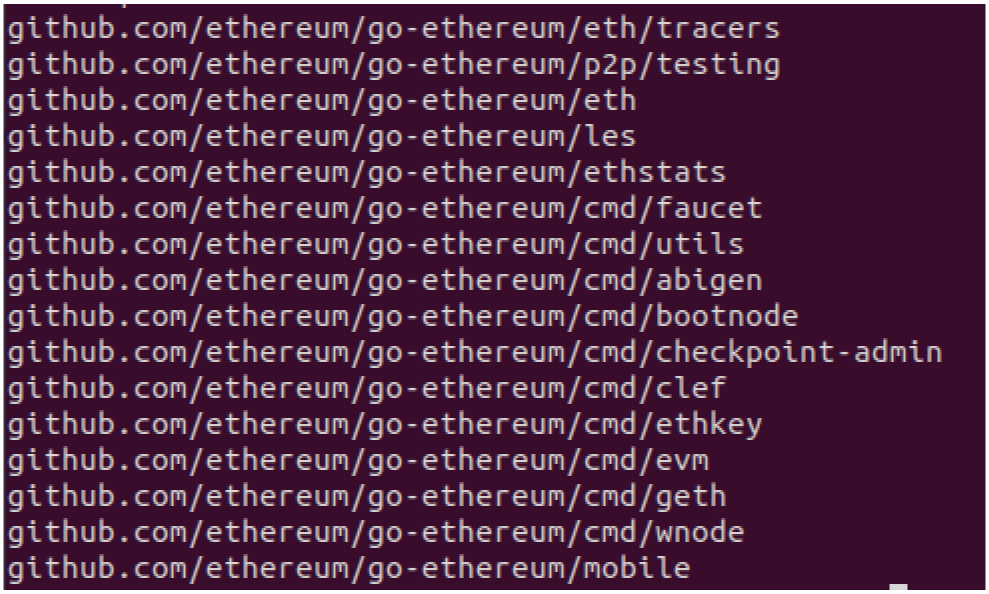
$go version



```
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$ go version
go version go1.18 linux/amd64
```

$cd BTA/lab6/go-ethereum

$make all

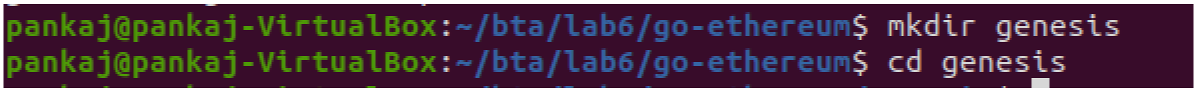


```
github.com/ethereum/go-ethereum/eth/tracers
github.com/ethereum/go-ethereum/p2p/testing
github.com/ethereum/go-ethereum/eth
github.com/ethereum/go-ethereum/les
github.com/ethereum/go-ethereum/ethstats
github.com/ethereum/go-ethereum/cmd/faucet
github.com/ethereum/go-ethereum/cmd/utils
github.com/ethereum/go-ethereum/cmd/abigen
github.com/ethereum/go-ethereum/cmd/bootnode
github.com/ethereum/go-ethereum/cmd/checkpoint-admin
github.com/ethereum/go-ethereum/cmd/clef
github.com/ethereum/go-ethereum/cmd/ethkey
github.com/ethereum/go-ethereum/cmd/evm
github.com/ethereum/go-ethereum/cmd/geth
github.com/ethereum/go-ethereum/cmd/wnode
github.com/ethereum/go-ethereum/mobile
```

Creating Blockchain- Genesis Block

$mkdir genesis

$ cd genesis

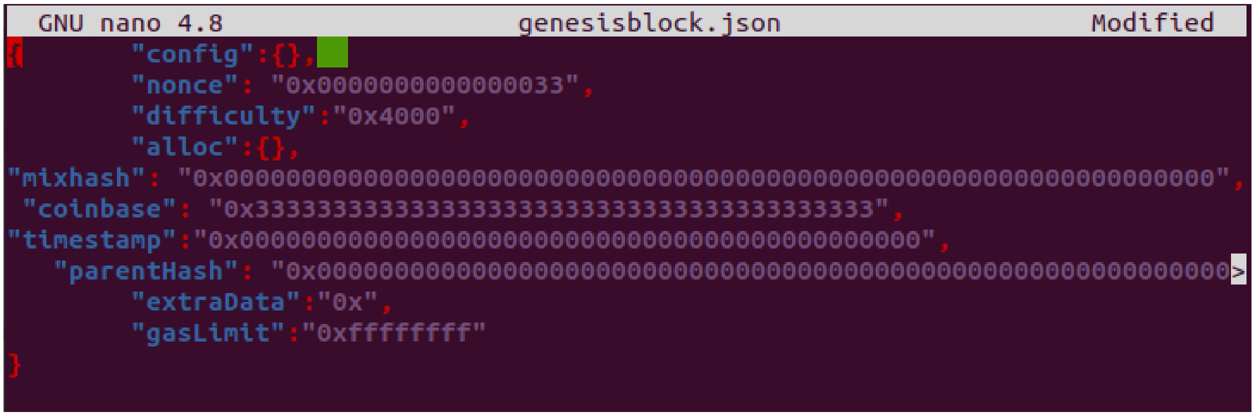


```
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$ mkdir genesis
pankaj@pankaj-VirtualBox:~/bta/lab6/go-ethereum$ cd genesis
```

#create the genesis block and add these lines

nano genesisblock.jsom

```
{      "config":{},

       "nonce": "0x0000000000000033",

       "difficulty":"0x4000",

       "alloc":{},

"mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",

 "coinbase": "0x3333333333333333333333333333333333333333",

"timestamp":"0x00000000000000000000000000000000000000000000",
   "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",

       "extraData":"0x",

       "gasLimit":"0xffffffff"

}
```



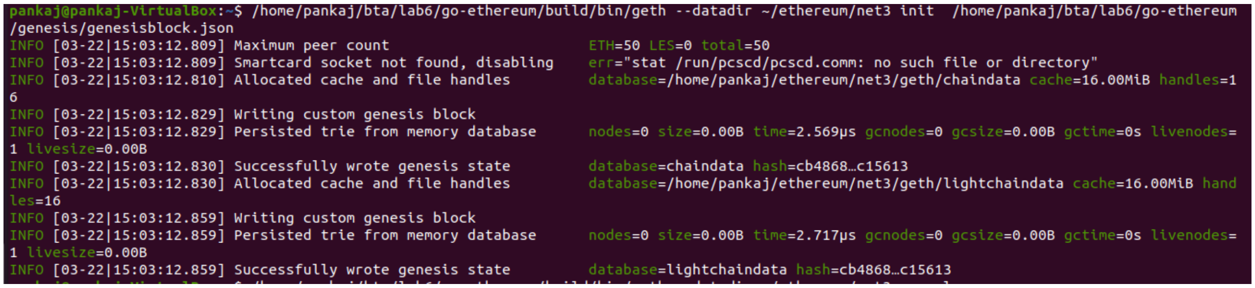
```
  GNU nano 4.8                 genesisblock.json                     Modified
{       "config":{},
        "nonce": "0x0000000000000033",
        "difficulty":"0x4000",
        "alloc":{},
"mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",
 "coinbase": "0x3333333333333333333333333333333333333333",
"timestamp":"0x00000000000000000000000000000000000000000000",
   "parentHash": "0x00000000000000000000000000000000000000000000000000000000000>
        "extraData":"0x",
        "gasLimit":"0xffffffff"
}
```

Starting the Blockchain

#Initializing the Blockchain

$/home/adaya/BTA/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 init genenesis/genesisblock.json

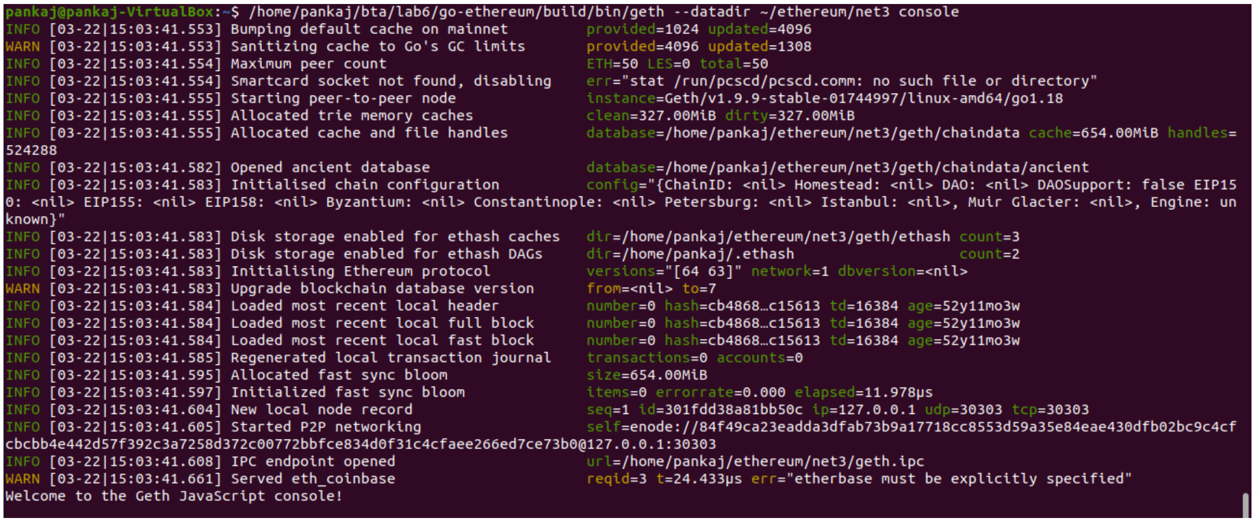
initializing block chain:



```
pankaj@pankaj-VirtualBox:~$ /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 init  /home/pankaj/bta/lab6/go-ethereum
/genesis/genesisblock.json
INFO [03-22|15:03:12.809] Maximum peer count                       ETH=50 LES=0 total=50
INFO [03-22|15:03:12.809] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
INFO [03-22|15:03:12.810] Allocated cache and file handles         database=/home/pankaj/ethereum/net3/geth/chaindata cache=16.00MiB handles=1
6
INFO [03-22|15:03:12.829] Writing custom genesis block
INFO [03-22|15:03:12.829] Persisted trie from memory database      nodes=0 size=0.00B time=2.569µs gcnodes=0 gcsize=0.00B gctime=0s livenodes=
1 livesize=0.00B
INFO [03-22|15:03:12.830] Successfully wrote genesis state         database=chaindata hash=cb4868…c15613
INFO [03-22|15:03:12.830] Allocated cache and file handles         database=/home/pankaj/ethereum/net3/geth/lightchaindata cache=16.00MiB hand
les=16
INFO [03-22|15:03:12.859] Writing custom genesis block
INFO [03-22|15:03:12.859] Persisted trie from memory database      nodes=0 size=0.00B time=2.717µs gcnodes=0 gcsize=0.00B gctime=0s livenodes=
1 livesize=0.00B
INFO [03-22|15:03:12.859] Successfully wrote genesis state         database=lightchaindata hash=cb4868…c15613
```

#Starting the geth console

$ /home/adaya/BTA/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 --networkid 3 console *geth console:*



```
pankaj@pankaj-VirtualBox:~$ /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 console
INFO [03-22|15:03:41.553] Bumping default cache on mainnet         provided=1024 updated=4096
WARN [03-22|15:03:41.553] Sanitizing cache to Go's GC limits       provided=4096 updated=1308
INFO [03-22|15:03:41.554] Maximum peer count                       ETH=50 LES=0 total=50
INFO [03-22|15:03:41.554] Smartcard socket not found, disabling    err="stat /run/pcscd/pcscd.comm: no such file or directory"
INFO [03-22|15:03:41.555] Starting peer-to-peer node               instance=Geth/v1.9.9-stable-01744997/linux-amd64/go1.18
INFO [03-22|15:03:41.555] Allocated trie memory caches             clean=327.00MiB dirty=327.00MiB
INFO [03-22|15:03:41.555] Allocated cache and file handles         database=/home/pankaj/ethereum/net3/geth/chaindata cache=654.00MiB handles=
524288
INFO [03-22|15:03:41.582] Opened ancient database                  database=/home/pankaj/ethereum/net3/geth/chaindata/ancient
INFO [03-22|15:03:41.583] Initialised chain configuration          config="{ChainID: <nil> Homestead: <nil> DAO: <nil> DAOSupport: false EIP15
0: <nil> EIP155: <nil> EIP158: <nil> Byzantium: <nil> Constantinople: <nil> Petersburg: <nil> Istanbul: <nil>, Muir Glacier: <nil>, Engine: un
known}"
INFO [03-22|15:03:41.583] Disk storage enabled for ethash caches   dir=/home/pankaj/ethereum/net3/geth/ethash count=3
INFO [03-22|15:03:41.583] Disk storage enabled for ethash DAGs     dir=/home/pankaj/.ethash                   count=2
INFO [03-22|15:03:41.583] Initialising Ethereum protocol           versions="[64 63]" network=1 dbversion=<nil>
WARN [03-22|15:03:41.583] Upgrade blockchain database version      from=<nil> to=7
INFO [03-22|15:03:41.584] Loaded most recent local header          number=0 hash=cb4868…c15613 td=16384 age=52y11mo3w
INFO [03-22|15:03:41.584] Loaded most recent local full block      number=0 hash=cb4868…c15613 td=16384 age=52y11mo3w
INFO [03-22|15:03:41.584] Loaded most recent local fast block      number=0 hash=cb4868…c15613 td=16384 age=52y11mo3w
INFO [03-22|15:03:41.585] Regenerated local transaction journal    transactions=0 accounts=0
INFO [03-22|15:03:41.595] Allocated fast sync bloom                size=654.00MiB
INFO [03-22|15:03:41.597] Initialized fast sync bloom              items=0 errorrate=0.000 elapsed=11.978µs
INFO [03-22|15:03:41.604] New local node record                    seq=1 id=301fdd38a81bb50c ip=127.0.0.1 udp=30303 tcp=30303
INFO [03-22|15:03:41.605] Started P2P networking                   self=enode://84f49ca23eadda3dfab73b9a17718cc8553d59a35e84eae430dfb02bc9c4cf
cbcbb4e442d57f392c3a7258d372c00772bbfce834d0f31c4cfaee266ed7ce73b0@127.0.0.1:30303
INFO [03-22|15:03:41.608] IPC endpoint opened                      url=/home/pankaj/ethereum/net3/geth.ipc
WARN [03-22|15:03:41.661] Served eth_coinbase                      reqid=3 t=24.433µs err="etherbase must be explicitly specified"
Welcome to the Geth JavaScript console!
```

>

# Starting Blockchain

>personal.newAccount()

>eth.accounts

>ethtBlockNumber

>miner.start()

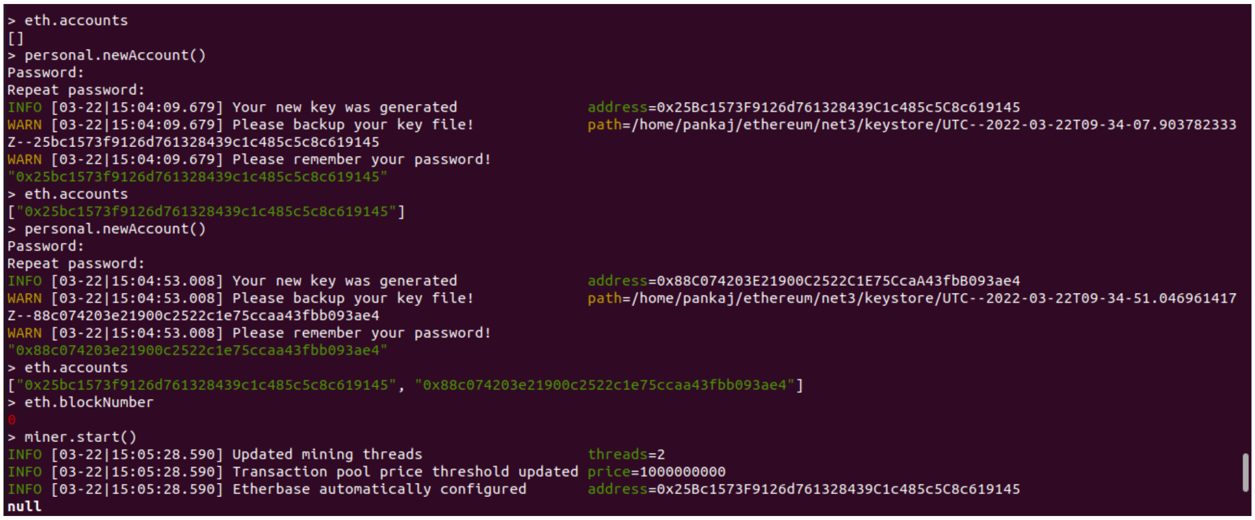
>miner.stop()

>eth.accounts

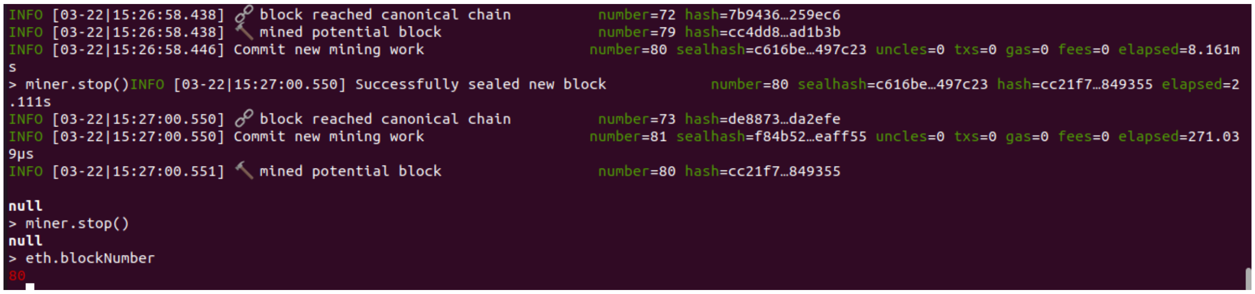
>eth.getBalance(“EAO Address”)

>exit

making accounts and starting miner:

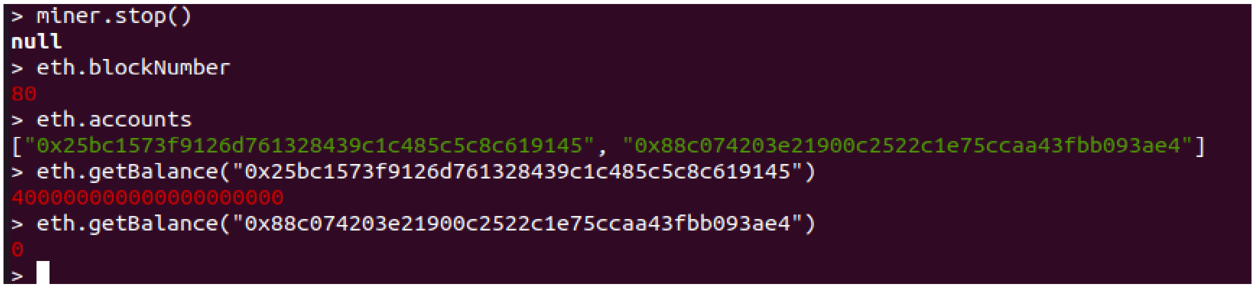


```
> eth.accounts
[]
> personal.newAccount()
Password:
Repeat password:
INFO [03-22|15:04:09.679] Your new key was generated               address=0x25Bc1573F9126d761328439C1c485c5C8c619145
WARN [03-22|15:04:09.679] Please backup your key file!             path=/home/pankaj/ethereum/net3/keystore/UTC--2022-03-22T09-34-07.903782333
Z--25bc1573f9126d761328439c1c485c5c8c619145
WARN [03-22|15:04:09.679] Please remember your password!
"0x25bc1573f9126d761328439c1c485c5c8c619145"
> eth.accounts
["0x25bc1573f9126d761328439c1c485c5c8c619145"]
> personal.newAccount()
Password:
Repeat password:
INFO [03-22|15:04:53.008] Your new key was generated               address=0x88C074203E21900C2522C1E75CcaA43fbB093ae4
WARN [03-22|15:04:53.008] Please backup your key file!             path=/home/pankaj/ethereum/net3/keystore/UTC--2022-03-22T09-34-51.046961417
Z--88c074203e21900c2522c1e75ccaa43fbb093ae4
WARN [03-22|15:04:53.008] Please remember your password!
"0x88c074203e21900c2522c1e75ccaa43fbb093ae4"
> eth.accounts
["0x25bc1573f9126d761328439c1c485c5c8c619145", "0x88c074203e21900c2522c1e75ccaa43fbb093ae4"]
> eth.blockNumber
0
> miner.start()
INFO [03-22|15:05:28.590] Updated mining threads                   threads=2
INFO [03-22|15:05:28.590] Transaction pool price threshold updated price=1000000000
INFO [03-22|15:05:28.590] Etherbase automatically configured       address=0x25Bc1573F9126d761328439C1c485c5C8c619145
null
```



```
INFO [03-22|15:26:58.438] 🔗 block reached canonical chain          number=72 hash=7b9436…259ec6
INFO [03-22|15:26:58.438] 🔨 mined potential block                  number=79 hash=cc4dd8…ad1b3b
INFO [03-22|15:26:58.446] Commit new mining work                   number=80 sealhash=c616be…497c23 uncles=0 txs=0 gas=0 fees=0 elapsed=8.161m
s
> miner.stop()INFO [03-22|15:27:00.550] Successfully sealed new block            number=80 sealhash=c616be…497c23 hash=cc21f7…849355 elapsed=2
.111s
INFO [03-22|15:27:00.550] 🔗 block reached canonical chain          number=73 hash=de8873…da2efe
INFO [03-22|15:27:00.550] Commit new mining work                   number=81 sealhash=f84b52…eaff55 uncles=0 txs=0 gas=0 fees=0 elapsed=271.03
9µs
INFO [03-22|15:27:00.551] 🔨 mined potential block                  number=80 hash=cc21f7…849355

null
> miner.stop()
null
> eth.blockNumber
80
```

stopping miner and seeing last blocknumber of chain and getting balance of both the accounts:



```
> miner.stop()
null
> eth.blockNumber
80
> eth.accounts
["0x25bc1573f9126d761328439c1c485c5c8c619145", "0x88c074203e21900c2522c1e75ccaa43fbb093ae4"]
> eth.getBalance("0x25bc1573f9126d761328439c1c485c5c8c619145")
400000000000000000000
> eth.getBalance("0x88c074203e21900c2522c1e75ccaa43fbb093ae4")
0
> 
```

Tasks:

[1] Mine at least 50 blocks and check and verify it

[2] Check the balances of each EAO accounts

[3] Send Ethers and verify

eth.sendTransaction({

    from: "EAO1",

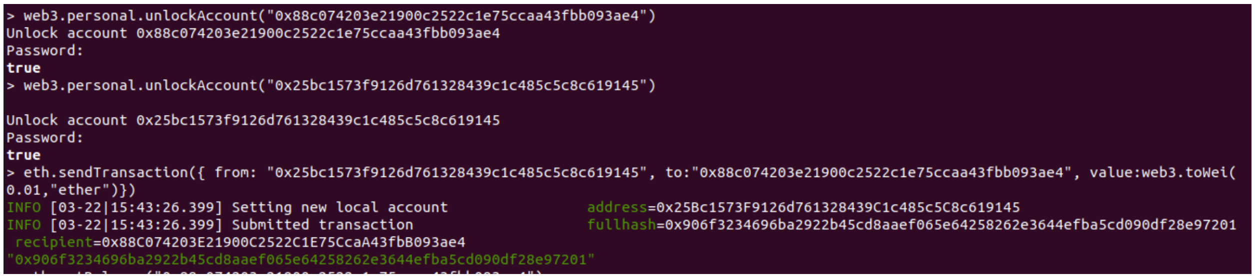
    to: "EAO2”,

    value: web3.toWei(0.01, "ether")

})

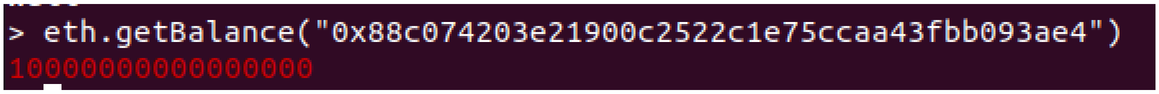
[4]

unlocking accounts and sending ethers from one account to another account



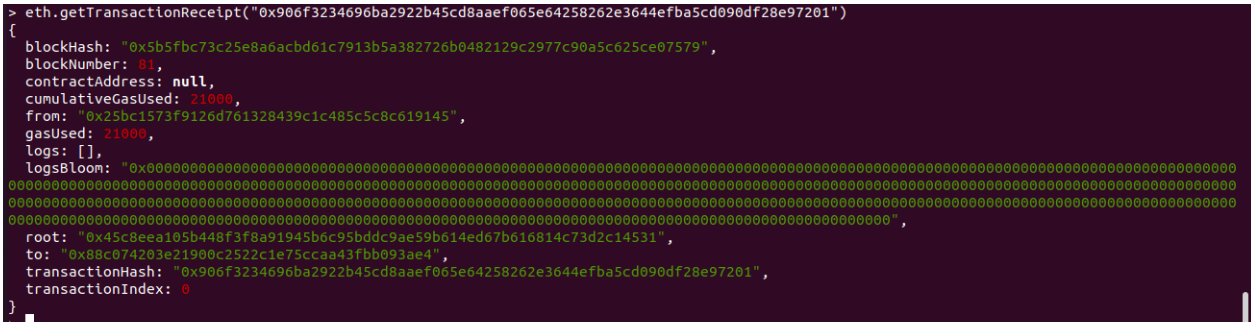
```
> web3.personal.unlockAccount("0x88c074203e21900c2522c1e75ccaa43fbb093ae4")
Unlock account 0x88c074203e21900c2522c1e75ccaa43fbb093ae4
Password:
true
> web3.personal.unlockAccount("0x25bc1573f9126d761328439c1c485c5c8c619145")

Unlock account 0x25bc1573f9126d761328439c1c485c5c8c619145
Password:
true
> eth.sendTransaction({ from: "0x25bc1573f9126d761328439c1c485c5c8c619145", to:"0x88c074203e21900c2522c1e75ccaa43fbb093ae4", value:web3.toWei(
0.01,"ether")})
INFO [03-22|15:43:26.399] Setting new local account                address=0x25Bc1573F9126d761328439C1c485c5C8c619145
INFO [03-22|15:43:26.399] Submitted transaction                    fullhash=0x906f3234696ba2922b45cd8aaef065e64258262e3644efba5cd090df28e97201
 recipient=0x88C074203E21900C2522C1E75CcaA43fbB093ae4
"0x906f3234696ba2922b45cd8aaef065e64258262e3644efba5cd090df28e97201"
```

verification:



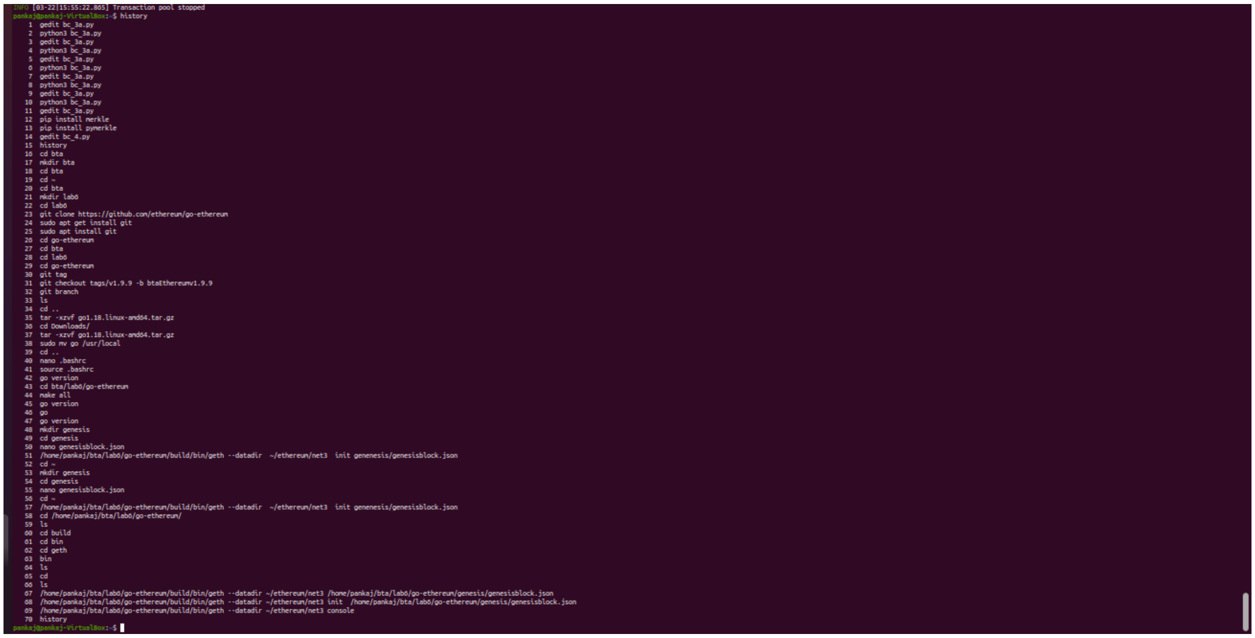
```
> eth.getBalance("0x88c074203e21900c2522c1e75ccaa43fbb093ae4")
10000000000000000
```

receipt:



```
> eth.getTransactionReceipt("0x906f3234696ba2922b45cd8aaef065e64258262e3644efba5cd090df28e97201")
{
  blockHash: "0x5b5fbc73c25e8a6acbd61c7913b5a382726b0482129c2977c90a5c625ce07579",
  blockNumber: 81,
  contractAddress: null,
  cumulativeGasUsed: 21000,
  from: "0x25bc1573f9126d761328439c1c485c5c8c619145",
  gasUsed: 21000,
  logs: [],
  logsBloom: "0x000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000",
  root: "0x45c8eea105b448f3f8a91945b6c95bddc9ae59b614ed67b616814c73d2c14531",
  to: "0x88c074203e21900c2522c1e75ccaa43fbb093ae4",
  transactionHash: "0x906f3234696ba2922b45cd8aaef065e64258262e3644efba5cd090df28e97201",
  transactionIndex: 0
}
```

history command:



```
INFO [03-22|15:55:22.865] Transaction pool stopped
pankaj@pankaj-VirtualBox:~$ history
    1  gedit bc_3a.py
    2  python3 bc_3a.py
    3  gedit bc_3a.py
    4  python3 bc_3a.py
    5  gedit bc_3a.py
    6  python3 bc_3a.py
    7  gedit bc_3a.py
    8  python3 bc_3a.py
    9  gedit bc_3a.py
   10  python3 bc_3a.py
   11  gedit bc_3a.py
   12  pip install merkle
   13  pip install pymerkle
   14  gedit bc_4.py
   15  history
   16  cd bta
   17  mkdir bta
   18  cd bta
   19  cd ~
   20  cd bta
   21  mkdir lab6
   22  cd lab6
   23  git clone https://github.com/ethereum/go-ethereum
   24  sudo apt get install git
   25  sudo apt install git
   26  cd go-ethereum
   27  cd bta
   28  cd lab6
   29  cd go-ethereum
   30  git tag
   31  git checkout tags/v1.9.9 -b btaEthereumv1.9.9
   32  git branch
   33  ls
   34  cd ..
   35  tar -xzvf go1.18.linux-amd64.tar.gz
   36  cd Downloads/
   37  tar -xzvf go1.18.linux-amd64.tar.gz
   38  sudo mv go /usr/local
   39  cd ..
   40  nano .bashrc
   41  source .bashrc
   42  go version
   43  cd bta/lab6/go-ethereum
   44  make all
   45  go version
   46  go
   47  go version
   48  mkdir genesis
   49  cd genesis
   50  nano genesisblock.json
   51  /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir  ~/ethereum/net3  init genenesis/genesisblock.json
   52  cd ~
   53  mkdir genesis
   54  cd genesis
   55  nano genesisblock.json
   56  cd ~
   57  /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir  ~/ethereum/net3  init genenesis/genesisblock.json
   58  cd /home/pankaj/bta/lab6/go-ethereum/
   59  ls
   60  cd build
   61  cd bin
   62  cd geth
   63  bin
   64  ls
   65  cd
   66  ls
   67  /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 /home/pankaj/bta/lab6/go-ethereum/genesis/genesisblock.json
   68  /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 init  /home/pankaj/bta/lab6/go-ethereum/genesis/genesisblock.json
   69  /home/pankaj/bta/lab6/go-ethereum/build/bin/geth --datadir ~/ethereum/net3 console
   70  history
pankaj@pankaj-VirtualBox:~$
```

Conclusion:

Successfully implemented Ethereum private blockchain,created the genesis block.started the Ethereum blockchain,created accounts on the blockchain,transacted Ethers on blockchain,queried the blockchain using Geth- Geth console, Geth attach, Geth JSON RPC,used Eth and Web3 interface over RPC and verified transaction with receipt

References:

[1] Go Language Installation

https://go.dev/dl/go1.18.linux-amd64.tar.gz

[2] Download or clone and compile ethereum code

https://github.com/ethereum/go-ethereum

[3] Official Ethereum website

Go Ethereum

[4] Mastering Blockchain Technology by Imran Bashir 3rd Edition Chapter 11,12 and 13,Packt Publications