

PRACTICAL 9

AIM:

Build the private blockchain network using Geth.

STEPS OF IMPLEMENTATION:

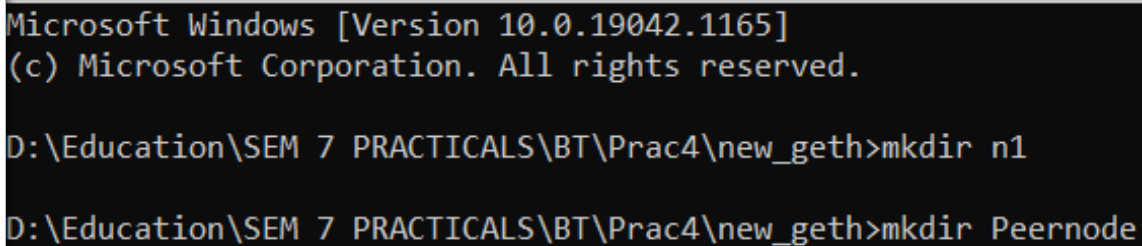
First we will create a main directory. Here we're naming it "new_geth" but any name will work. Then in cmd, we will navigate to it.

```
cd new_geth
```

Then we will make 2 directories inside that folder. We're naming then n1 and peernode.

```
mkdir n1
```

```
mkdir Peernode
```

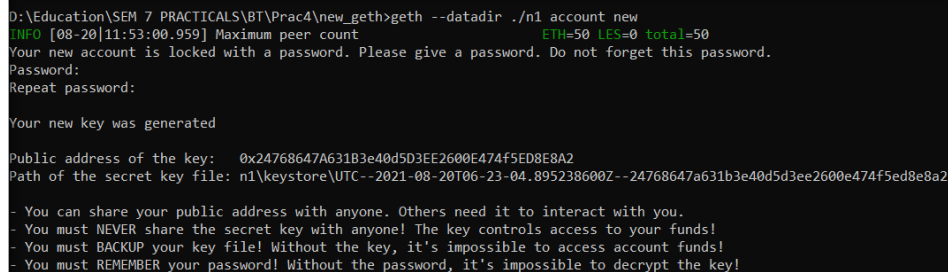


```
Microsoft Windows [Version 10.0.19042.1165]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth>mkdir n1  
  
D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth>mkdir Peernode
```

Then we will create new accounts in both the folders with following commands.

```
geth --datadir ./n1 account new
```

```
geth --datadir ./Peernode account new
```



```
D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth>geth --datadir ./n1 account new  
INFO [08-20|11:53:00.959] Maximum peer count ETH=50 LES=0 total=50  
Your new account is locked with a password. Please give a password. Do not forget this password.  
Password:  
Repeat password:  
  
Your new key was generated  
  
Public address of the key: 0x24768647A631B3e40d5D3EE2600E474f5ED8E8A2  
Path of the secret key file: n1\keystore\UTC--2021-08-20T06-23-04.895238600Z--24768647a631b3e40d5d3ee2600e474f5ed8e8a2  
  
- You can share your public address with anyone. Others need it to interact with you.  
- You must NEVER share the secret key with anyone! The key controls access to your funds!  
- You must BACKUP your key file! Without the key, it's impossible to access account funds!  
- You must REMEMBER your password! Without the password, it's impossible to decrypt the key!
```

That would generate public address of both accounts.

We will create genesis.json file in both directories with following code.

```
{
"config": {
"chainId": 1234,
"homesteadBlock": 0,
"eip150Block": 0,
"eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000",
"eip155Block": 0,
"eip158Block": 0,
"byzantiumBlock": 0
},
"difficulty": "400",
"gasLimit": "2000000",
"alloc": {
"<public-address-of-the-key>": {
"balance": "5000000000000000000000000000000000000000000000000000000000000000"
}
}
}
```

Then we will initiate both the genesis blocks.

```
geth --datadir ./n1 init n1/genesis.json
geth --datadir ./Peernode init Peernode/genesis.json
```

```
D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth>geth --datadir ./n1 init n1/genesis.json
INFO [08-20|11:55:25.742] Maximum peer count          ETH=50 LES=0 total=50
INFO [08-20|11:55:25.782] Set global gas cap          cap=50,000,000
INFO [08-20|11:55:25.791] Allocated cache and file handles databases=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\n1\geth\chaindata" cache=16.00MiB handles=16
INFO [08-20|11:55:25.978] Writing custom genesis block
INFO [08-20|11:55:25.989] Persisted trie from memory database nodes=1 size=149.00B time=0s gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [08-20|11:55:26.042] Successfully wrote genesis state database=chaindata hash=07573b..75535a
INFO [08-20|11:55:26.065] Allocated cache and file handles database=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\n1\geth\lightchaindata" cache=16.00MiB handles=16
INFO [08-20|11:55:26.251] Writing custom genesis block
INFO [08-20|11:55:26.266] Persisted trie from memory database nodes=1 size=149.00B time=1.0246ms gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [08-20|11:55:26.289] Successfully wrote genesis state database=lightchaindata hash=07573b..75535a

D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth>geth --datadir ./Peernode init Peernode/genesis.json
INFO [08-20|11:55:32.154] Maximum peer count          ETH=50 LES=0 total=50
INFO [08-20|11:55:32.160] Set global gas cap          cap=50,000,000
INFO [08-20|11:55:32.163] Allocated cache and file handles database=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\Peernode\geth\chaindata" cache=16.00MiB handles=16
INFO [08-20|11:55:32.292] Writing custom genesis block
INFO [08-20|11:55:32.305] Persisted trie from memory database nodes=1 size=149.00B time="87.6us" gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [08-20|11:55:32.337] Successfully wrote genesis state database=chaindata hash=175976..cd48c4
INFO [08-20|11:55:32.363] Allocated cache and file handles database=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\Peernode\geth\lightchaindata" cache=16.00MiB handles=16
INFO [08-20|11:55:32.514] Writing custom genesis block
INFO [08-20|11:55:32.528] Persisted trie from memory database nodes=1 size=149.00B time=0s gcnodes=0 gcsizes=0.00B gctime=0s livenodes=1 liveness=0.00B
INFO [08-20|11:55:32.563] Successfully wrote genesis state database=lightchaindata hash=175976..cd48c4
```

Then we will start interactive javascript environment to interact with our newly created blockchain.

```
geth --datadir ./n1 --networkid 1234 --ipcdisable --rpc --rpccorsdomain "*" --rpcapi
"db,eth,net,web3,personal" --rpcaddr "0.0.0.0" console
```

```

D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth --datadir .\nl --networkid 1234 --ipcdisable --rpc --rpcorsdomain "" --rpcapi "db,eth,net,web3,personal" --rpcaddr "0.0.0.0" console
INFO [08-20|11:55:40.695] Maximum peer count          ETH=50 |FS=0 |total=50
WARN [08-20|11:55:40.699] The flag --rpc is deprecated and will be removed June 2021, please use --http
WARN [08-20|11:55:40.703] The flag --rpcaddr is deprecated and will be removed June 2021, please use --http.addr
WARN [08-20|11:55:40.709] The flag --rpcorsdomain is deprecated and will be removed June 2021, please use --http.corsdomain
WARN [08-20|11:55:40.723] The flag --rpcapi is deprecated and will be removed June 2021, please use --http.api
INFO [08-20|11:55:40.728] Set global gas cap          cap=50,000,000
INFO [08-20|11:55:40.734] Allocated trie memory caches        clean=154.00MiB dirty=256.00MiB
INFO [08-20|11:55:40.739] Allocated cache and file handles   database=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\nl\geth\chaindata cache=512.00MiB handles=8192
INFO [08-20|11:55:42.094] Opened ancient database            database=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\nl\geth\chaindata\ancient readOnly=false
INFO [08-20|11:55:42.129] Initialised chain configuration   config={ChainID: 1234 Homestead: 0 DAO: <nil> DAOSupport: false EIP150: 0 EIP155: 0 EIP158: 0 Byzantium: 0 Constantinople: <nil> Petersburg: <nil> Istanbul: <nil> Muir Glacier: <nil> Berlin: <nil> London: <nil> Engine: unknown}
INFO [08-20|11:55:42.197] Disk storage enabled for ethash caches dir=D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\nl\geth\ethash count=3
INFO [08-20|11:55:42.228] Disk storage enabled for ethash DAGs dir=C:\Users\jilsa\AppData\Local\Ethash count=2
INFO [08-20|11:55:42.252] Initialising Ethereum protocol   network=1234 dbversion=<nil>
INFO [08-20|11:55:42.280] Loaded most recent local header   number=0 hash=07572b..75535a td=400 age=52y4mo2w
INFO [08-20|11:55:42.298] Loaded most recent local full block number=0 hash=07572b..75535a td=400 age=52y4mo2w
INFO [08-20|11:55:42.305] Loaded most recent local fast block number=0 hash=07572b..75535a td=400 age=52y4mo2w
WARN [08-20|11:55:42.310] Failed to load snapshot, regenerating err="missing or corrupted snapshot"
INFO [08-20|11:55:42.320] Rebuilding state snapshot         root=025935..377ad8 accounts=0 slots=0 storage=0.000 elapsed="557.5µs"
INFO [08-20|11:55:42.323] Resuming state snapshot generation transactions=0 accounts=0
INFO [08-20|11:55:42.334] Regenerated local transaction journal accounts=1 slots=0 storage=48.000 elapsed=24.482ms
INFO [08-20|11:55:42.347] Generated state snapshot          threshold=2
WARN [08-20|11:55:42.375] Gasprice oracle is ignoring threshold set error="leveldb: not found"
INFO [08-20|11:55:42.388] Error reading unclean shutdown markers instance=Geth/v1.10.7-stable-12f0ff40/windows-amd64/go1.16.4
INFO [08-20|11:55:42.402] Starting peer-to-peer node       peer=1 id=7f5c4c3cf26bd40 ip=127.0.0.1 udp=30303 tcp=30303
INFO [08-20|11:55:42.553] Started P2P networking           self=enode://88c49883ced698fcc3302d074b290c76ddc41b8a47ba9d69442249b7d117a629b27277397a40e6db6190507eee38c3ff40593e79041aba9768c7797b20dbc3f4@10.140.12.193:30303
INFO [08-20|11:55:42.554] Unavailable modules in HTTP API list unavailable=[db] available=["admin debug web3 eth txpool personal ethash miner net"]
INFO [08-20|11:55:42.645] HTTP server started              endpoint=[::]:8545 prefix=cors* vhosts=localhost
INFO [08-20|11:55:42.985] Etherbase automatically configured address=0x24768647a631b3e40d5d3e2600e474f5e08e8a2
Welcome to the Geth JavaScript console!

Instance: Geth/v1.10.7-stable-12f0ff40/windows-amd64/go1.16.4
coinbase: 0x24768647a631b3e40d5d3e2600e474f5e08e8a2
at block: 0 (Thu Jan 01 1970 05:30:00 GMT+0530 (IST))
datadir: D:\Education\SEM 7 PRACTICALS\BT\Prac4\new_geth\nl
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

```

To check balance in our account, we will simply type the following command while the background process is still running.

```
eth.getBalance(web3.eth.accounts[0])
```

```

> eth.getBalance(web3.eth.accounts[0])_
5e+23
>

```

Admin object exposes methods to interact with the RPC APIs. To get information about enode, we will write following command.

```
admin.nodeInfo.enode
```

```

> admin.nodeInfo.enode_
"enode://88c49883ced698fcc3302d074b290c76ddc41b8a47ba9d69442249b7d117a629b27277397a40e6db6190507eee38c3ff40593e79041aba9768c7797b20dbc3f4@10.140.12.193:30303"
>

```

We will copy this enode address and Request adding a new remote node to the list of tracked static nodes.

```
admin.addPeer("paste the enode address")
```

```

> admin.addPeer("enode://88c49883ced698fcc3302d074b290c76ddc41b8a47ba9d69442249b7d117a629b27277397a40e6db6190507eee38c3ff40593e79041aba9768c7797b20dbc3f4@10.140.12.193:30303")
true
>

```

To check your the etherbase account balance, we will use...

```
eth.getBalance(eth.coinbase)
```

```

> eth.getBalance(eth.coinbase)
5e+23
>

```

Now we will set the etherbase, where mining rewards will go.

```
miner.setEtherbase(web3.eth.accounts[0])
```

```
> miner.setEtherbase(web3.eth.accounts[0])
true
> _
```

Then finally, we are ready to mine some Ethereum now.

```
miner.start()
```

```
> miner.start()
INFO [08-20|11:59:34.664] Updated mining threads          threads=8
INFO [08-20|11:59:34.879] Transaction pool price threshold updated price=1,000,000,000
null
INFO [08-20|11:59:35.107] Commit new mining work          number=1 sealhash=f9eced..a6556a uncles=0 txs=0 gas=0 fees=0 elapsed=218.443ms
INFO [08-20|11:59:36.933] Generating DAG in progress      epoch=0 percentage=0 elapsed=903.890ms
INFO [08-20|11:59:37.800] Generating DAG in progress      epoch=0 percentage=1 elapsed=1.771s
INFO [08-20|11:59:38.681] Generating DAG in progress      epoch=0 percentage=2 elapsed=2.652s
INFO [08-20|11:59:39.507] Generating DAG in progress      epoch=0 percentage=3 elapsed=3.478s
INFO [08-20|11:59:40.000] Looking for peers               peercount=0 tried=119 static=1
INFO [08-20|11:59:40.375] Generating DAG in progress      epoch=0 percentage=4 elapsed=4.345s
INFO [08-20|11:59:41.219] Generating DAG in progress      epoch=0 percentage=5 elapsed=5.189s
INFO [08-20|11:59:42.036] Generating DAG in progress      epoch=0 percentage=6 elapsed=6.007s
INFO [08-20|11:59:42.837] Generating DAG in progress      epoch=0 percentage=7 elapsed=6.807s
INFO [08-20|11:59:43.633] Generating DAG in progress      epoch=0 percentage=8 elapsed=7.603s
INFO [08-20|11:59:44.497] Generating DAG in progress      epoch=0 percentage=9 elapsed=8.468s
INFO [08-20|11:59:45.303] Generating DAG in progress      epoch=0 percentage=10 elapsed=9.274s
INFO [08-20|11:59:46.112] Generating DAG in progress      epoch=0 percentage=11 elapsed=10.082s
INFO [08-20|11:59:46.920] Generating DAG in progress      epoch=0 percentage=12 elapsed=10.891s
INFO [08-20|11:59:47.739] Generating DAG in progress      epoch=0 percentage=13 elapsed=11.709s
INFO [08-20|11:59:48.537] Generating DAG in progress      epoch=0 percentage=14 elapsed=12.507s
INFO [08-20|11:59:49.371] Generating DAG in progress      epoch=0 percentage=15 elapsed=13.341s
INFO [08-20|11:59:50.212] Generating DAG in progress      epoch=0 percentage=16 elapsed=14.182s
INFO [08-20|11:59:50.518] Looking for peers               peercount=0 tried=27 static=1
INFO [08-20|11:59:51.025] Generating DAG in progress      epoch=0 percentage=17 elapsed=14.995s
INFO [08-20|11:59:51.889] Generating DAG in progress      epoch=0 percentage=18 elapsed=15.859s
INFO [08-20|11:59:52.737] Generating DAG in progress      epoch=0 percentage=19 elapsed=16.708s
INFO [08-20|11:59:53.663] Generating DAG in progress      epoch=0 percentage=20 elapsed=17.634s
```

To stop that process, we can use...

```
miner.stop()
```

```
INFO [08-20|12:02:45.270] Looking for peers               peercount=0 tried=92 static=1
INFO [08-20|12:02:45.363] Generating DAG in progress      epoch=1 percentage=98 elapsed=1m30.767s
INFO [08-20|12:02:46.443] Generating DAG in progress      epoch=1 percentage=99 elapsed=1m31.847s
INFO [08-20|12:02:46.450] Generated ethash verification cache epoch=1 elapsed=1m31.854s
> miner.stop()
null
INFO [08-20|12:02:56.380] Looking for peers               peercount=0 tried=15 static=1
```

Now we will again check balance with following command. Ideally balance will be less than previous balance i.e. the transaction fees deducted.

```
eth.getBalance(eth.coinbase)
```

```
> eth.getBalance(eth.coinbase)
5e+23
>
```

CONCLUSION:

In this practical, we used command prompt and geth to create our private Ethereum blockchain and we mined some Ethereum using it.