



# 树状区块链启动和创建节点

```
# geth
personal.newAccount("123456")      # 创建账户
eth.accounts[0]
personal.unlockAccount(eth.accounts[0], "123456", 15000) # 解锁账户
personal.unlockAccount(eth.accounts[1], "123456", 15000) # 解锁账户
personal.unlockAccount(eth.accounts[2], "123456", 15000) # 解锁账户
eth.blockNumber                     # 查看区块数
eth.accounts                        # 枚举用户
eth.getBalance()                    # 查看账户余额
eth.getBlock()                      # 获取区块
admin.addPeer()                     # 添加其他节点
miner.start(1)                      # 挖矿
miner.stop()                        # 停止
```

## 复现

树状区块链需要使用修改后以太坊geth客户端：

- 将新的geth文件（设置了树状区块链的文件）替换本地原有的geth位置

```
[ifpop@arch ~]$ whereis geth
geth: /usr/bin/geth
[ifpop@arch ~]$ sudo cp /usr/bin/geth /usr/bin/geth1
[ifpop@arch ~]$ sudo cp ./geth /usr/bin/geth
```

这样操作就可以使用两种 `geth`，测试是否更换成功：

```
[ifpop@arch ~]$ geth version
Geth
Version: 1.9.12-stable
Git Commit: fe8bdfb728b96fd445511ed71932ef67229e2738
Git Commit Date: 20210425
Architecture: amd64
Protocol Versions: [65 64 63]
Go Version: go1.13.9
Operating System: linux
GOPATH=
GOROOT=/usr/local/go
[ifpop@arch ~]$ geth1 version
Geth
Version: 1.10.3-stable
Git Commit: 991384a7f6719e1125ca0be7fb27d0c4d1c5d2d3
Git Commit Date: 20210505
Architecture: amd64
Go Version: go1.16.3
Operating System: linux
GOPATH=
GOROOT=go
```

## 部署合约之前先创建节点并初始化余额

- 为什么要有余额呢？因为在区块链的操作是会消耗余额的，所以先初始化足够的余额
- 在 `genesis.json` 所在文件夹下，初始化和启动移动区块链（如果启动报错（bind: invalid argument），是因为文件夹的路径太深了，剪切到浅一点的文件夹就好了）

```
# 初始化区块链
$ geth --identity "MyEth" --rpc --rpcaddr 127.0.0.1 --rpcport "8545" --rpccorsdomain "*" --datadir gethdata --port "30303" --no-discover --rpcapi "eth,net,personal,web3" --networkid 91036 init genesis.json
# 启动区块链
$ geth --identity "MyEth" --rpcaddr 127.0.0.1 --rpc --rpcport "8545" --rpccorsdomain "*" --datadir gethdata --port "30303" --no-discover --rpcapi "eth,net,personal,web3" --networkid 91036 --allow-insecure-unlock --dev.period 1 console
```

```
# 初始化
$ geth --identity "MyEth" --rpc --rpcaddr 127.0.0.1 --rpcport "8546" --rpccorsdomain "*" --datadir gethdata --port "30304" --nodiscover --rpcapi "eth,net,personal,web3" --networkid 91036 init genesis.json

# 启动
$ geth --identity "MyEth" --rpcaddr 127.0.0.1 --rpc --rpcport "8546" --rpccorsdomain "*" --datadir gethdata --port "30304" --nodiscover --rpcapi "eth,net,personal,web3" --networkid 91036 --allow-insecure-unlock --dev.period 1 console
```

```
Welcome to the Geth JavaScript console!

instance: Geth/MyEth/v1.9.12-stable-fdd6b536/linux-amd64/go1.13.9
coinbase: 0x87e2d88cd788e8b8be2682e356cb1f81bec3116e
at block: 0 (Thu Jan 01 1970 08:00:00 GMT+0800 (CST))
datadir: /home/ubuntu/桌面/db-udgraduate/undergraduate-dbin2021/src/MapStore_VN/gethdata
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

>
```

```
> personal.newAccount("123456")
```

```
INFO [05-29|10:34:46.970] Your new key was generated
WARN [05-29|10:34:46.971] Please backup your key file!
WARN [05-29|10:34:46.971] Please remember your password!
"0x87e2d88cd788e8b8be2682e356cb1f81bec3116e"
address=0x87E2D88cd788E8B8BE2682E356CB1f81BEC3116E
path=/home/ubuntu/桌面/db-udgraduate/undergraduate-dbin2021/src/Maps
```

下一步，退出Geth控制台（注意不是关闭窗口），参考命令：

[illegible]



## 树状区块链启动和创建节点

```

Qual = QualContract.new({
  from: web3.eth.accounts[0],
  data: bytecode,
  gas: '1400000',
  position: "w25111111111111",
  txtime: 277001
}, function (e, contract) {
  console.log(e, contract);
  if (!e) {
    if (!contract.address) {
      console.log("Contract transaction send: TransactionHash: " + contract.transactionHash + " waiting to be mined...");
    } else {
      console.log("Contract mined! Address: " + contract.address);
      console.log(contract);
    }
  }
});

```

- **StoreMap 部署**

## 部署StoreMap在移动区块链上

[illegible]

[illegible]

然后miner.start(), 挖着挖着就会返回合约的address, 保存好, 这是合约调用时的所需的信息。