**华东师范大学数据科学与工程学院实验报告**

|  |  |  |
| --- | --- | --- |
| **课程名称**： | **年级**： | **上机实践成绩**： |
| **指导教师**：张召 | **姓名**：赵煜硕 |  |
| **上机实践名称**：区块链第五次 | **学号**：10195501415 | **上机实践日期**： |
| **上机实践编号**： | **组号**： | **上机实践时间**： |

1. **实验目的**

编写Fabric练码

1. **实验任务**

**1.**

1. 使用Java编写e2e链码
2. 链码部署与调用
3. 熟悉e2e案例的工作原理

**2.**

（1）存储账单数据< id, amount > 根据id查询某一条账单的金额；

（2）当前的账单总额；

（3）金额最大的一笔账单的信息（比如返回其id与amount）；

（4）金额最小的一笔账单的信息（比如返回其id与amount）；

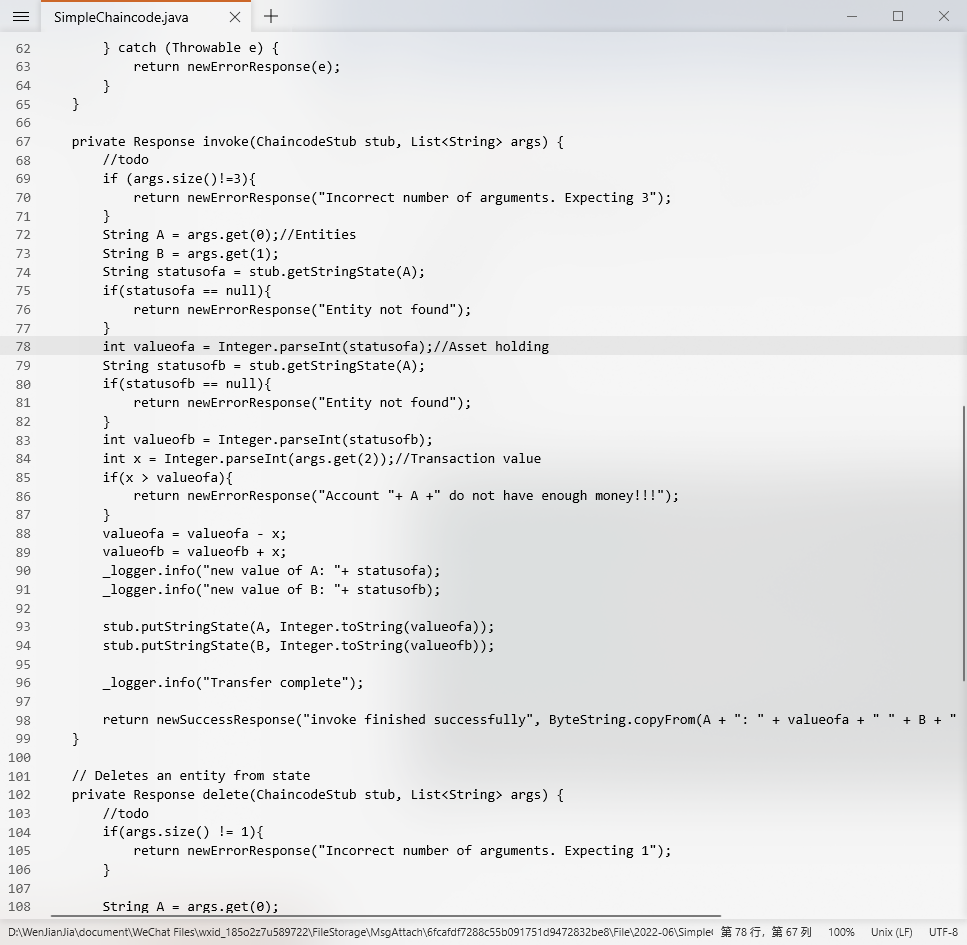
1. **使用环境**

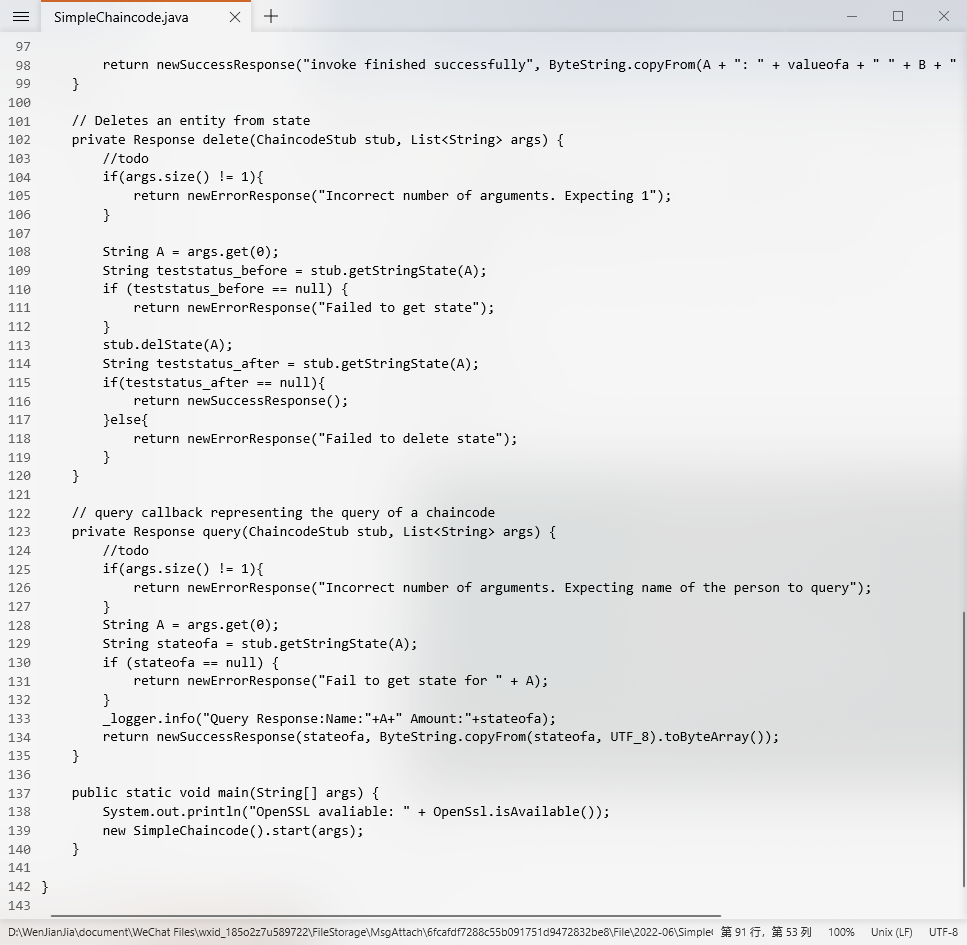
**Fabric1.1**

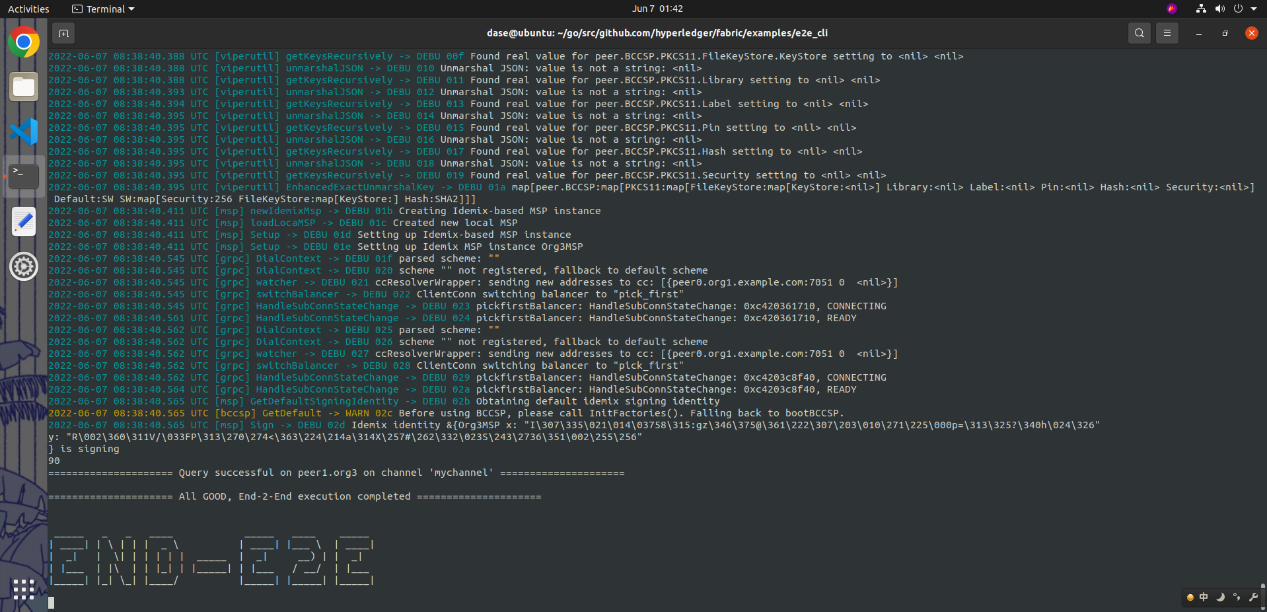
**Fabric1.3**

1. **实验过程**

Java编写练码

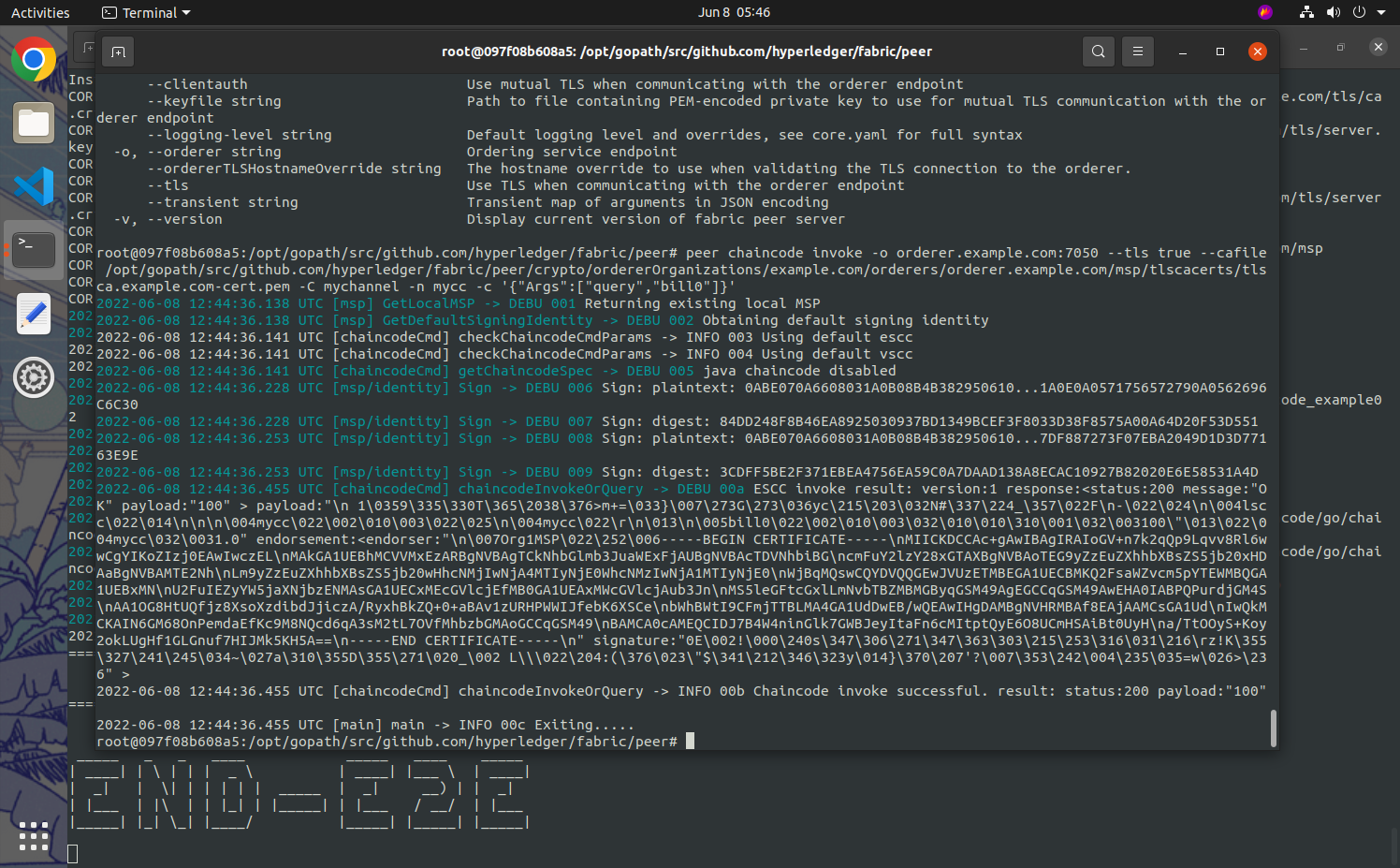




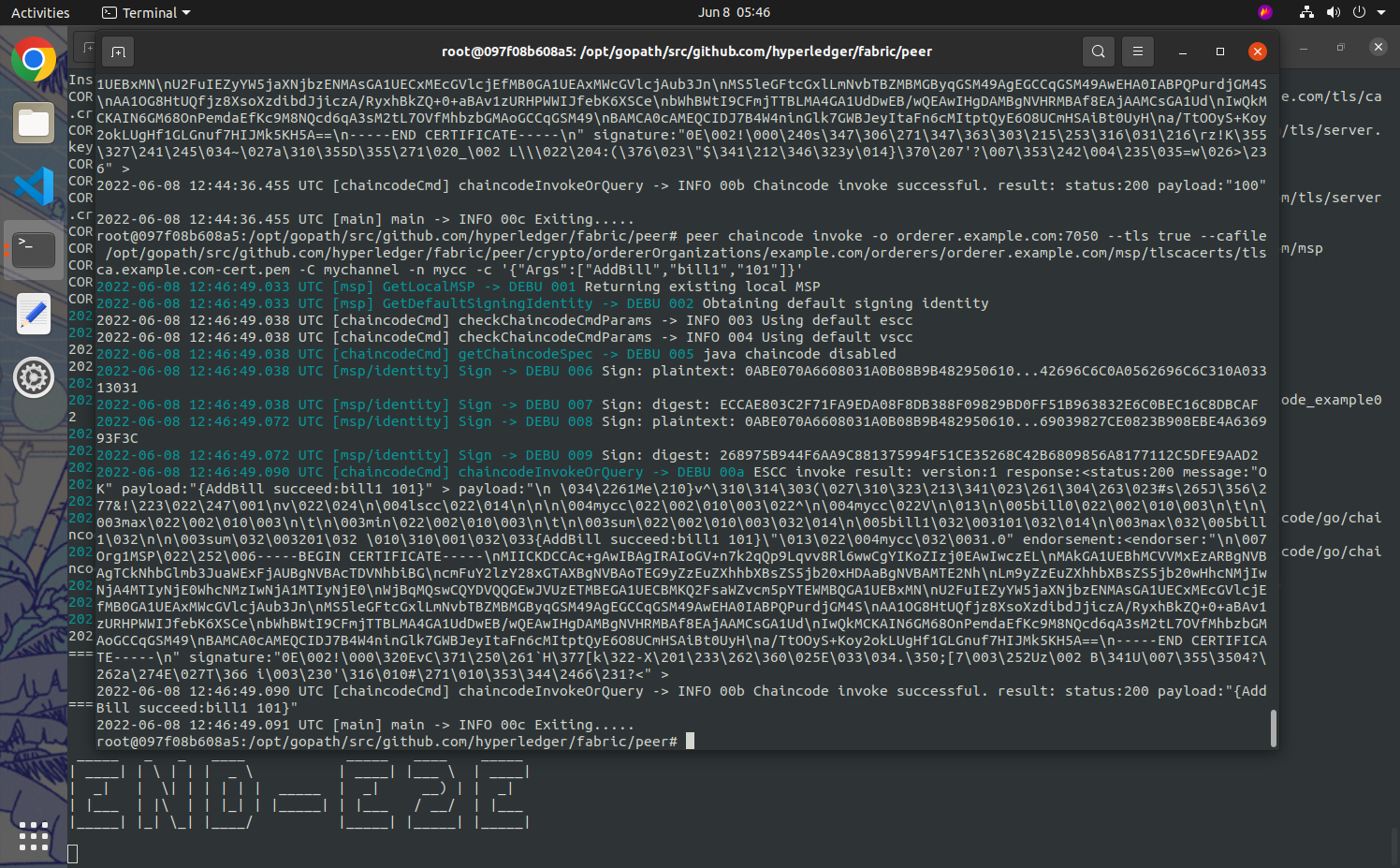


编写chaincode\_example02

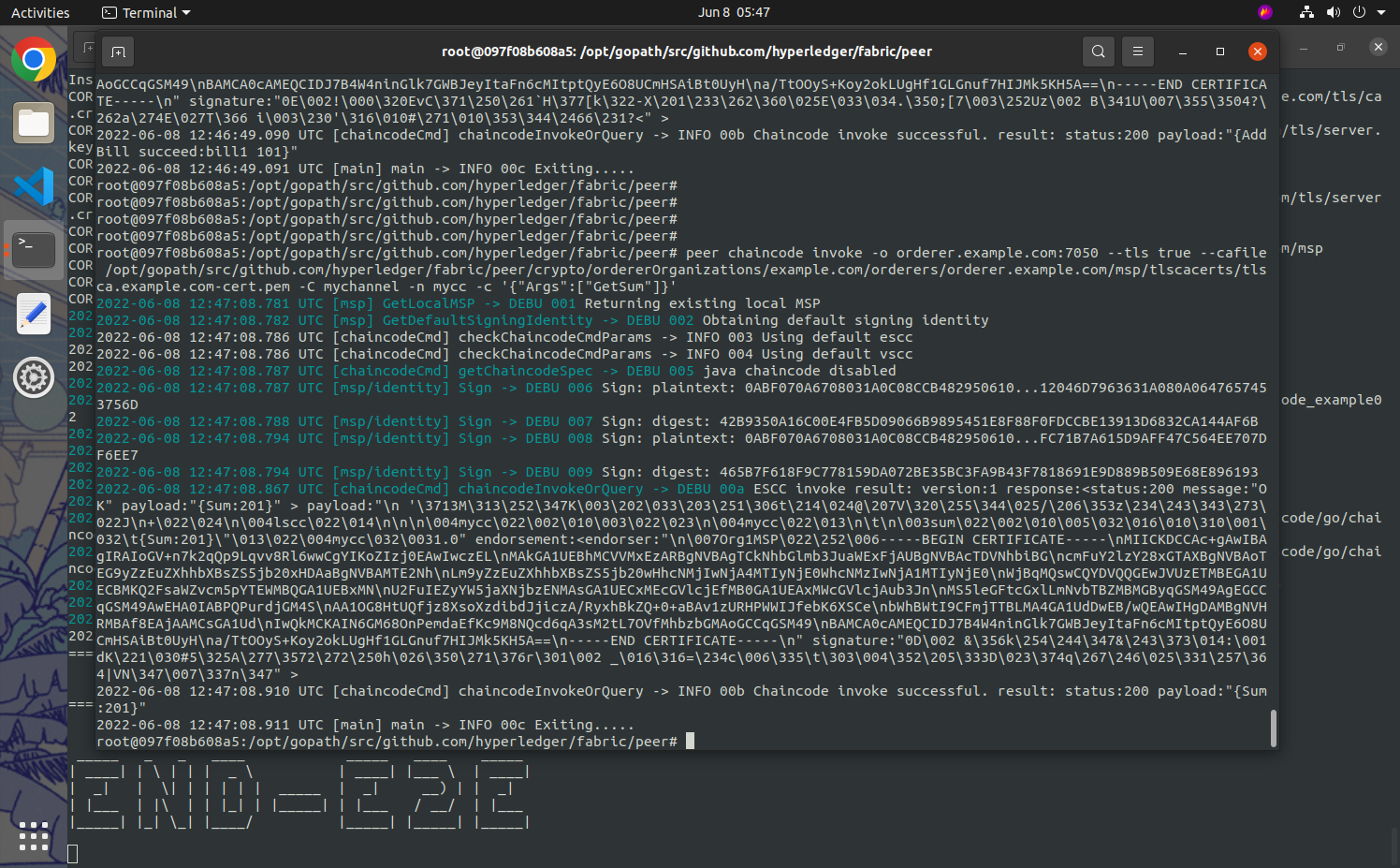
peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n mycc -c '{"Args":["query","bill0"]}'



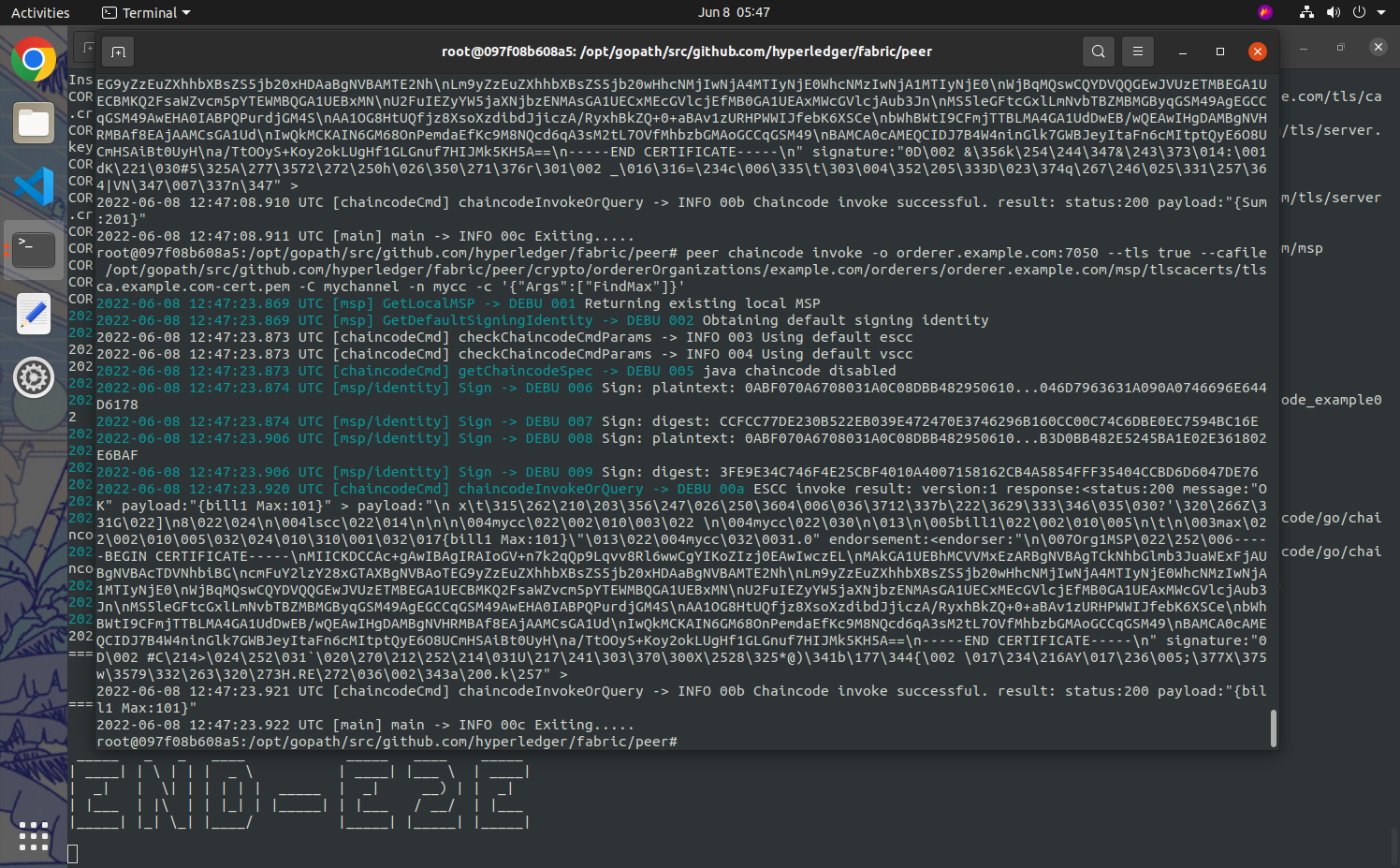
peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n mycc -c '{"Args":["AddBill","bill1","101"]}'



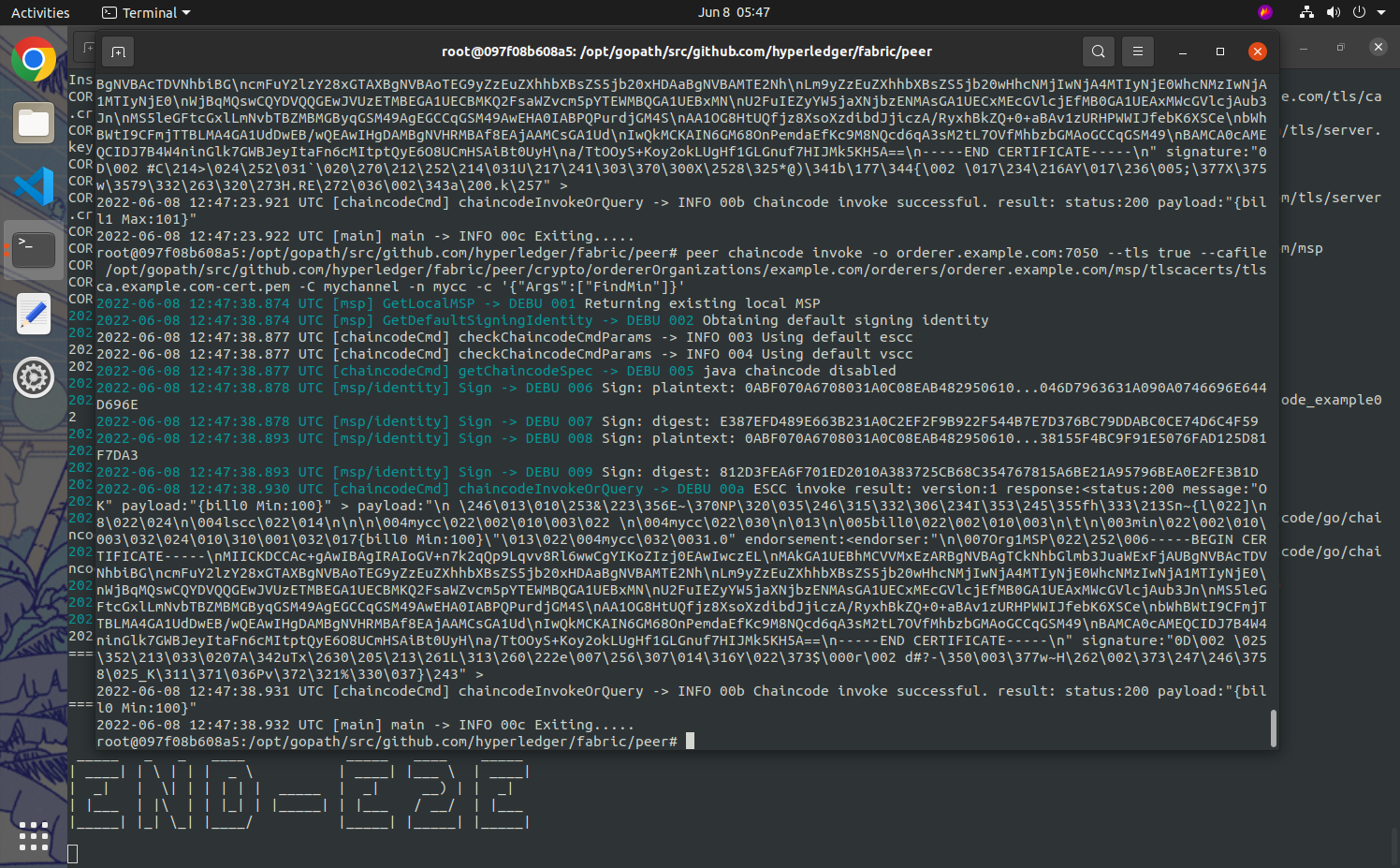
peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n mycc -c '{"Args":["GetSum"]}'



peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n mycc -c '{"Args":["FindMax"]}'



peer chaincode invoke -o orderer.example.com:7050 --tls true --cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C mychannel -n mycc -c '{"Args":["FindMin"]}'



1. **总结**

**1.** Fabric链码的种类该怎么划分？

用户维护的链码，系统链码

ESCC,

CSCC,ESCC,LSCC,QSCC,VSCC

2. 链码的生命周期有哪些？

打包，安装，实例化，升级，签名

3. Fabric chaincode和以太坊智能合约有什么不同？

多个智能合约能够被定义在同一个链码内，每一个智能合约都通过链码内的名字而被唯一标识。

Fabric采用背书策略

以太烦无授权操作模式，牺牲了性能，可扩展性。

Fabric 有授权，采用BFT和细粒度访问控制解决了性能可扩展性和隐私问题

1. 简单地说明一下客户端申请调用链码函数进行后fabric的运行流程

客户端注册

客户端发起交易提案到背书结点

背书结点收到后，验证后模拟执行智能合约，将结果和证书签名发给客户端

客户端收到后，判断提案结果是否一致，以及是否按照背书策略执行；若不是，中止；若是，将数据打包并签名，然后发给order

Order收到后做共识排序，根据区块生成策略；将一批交易打包成新的区块，发送给提交结点

提交结点校验每笔交易，完成后将区块追加到本地的链上