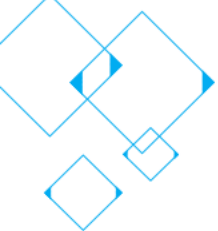


第2章 智能合约概述



重航区块链

CONTENTS

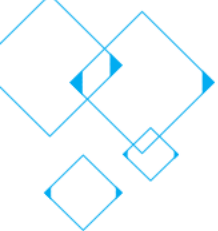
目录

第一章 智能合约概述

第二章 如何编写智能合约

第三章 开发环境搭建





CONTENTS

章节

第一章 智能合约概述

1.1 什么是智能合约

1.2 智能合约的发展史

1.3 智能合约是如何运行的



什么是智能合约

- 合约是经过双方或多方同意，约定立即执行或在将来执行一项交易的**法律文件**。因为合约是法律文件，所以它具有**强制性和可执行性**。
- 合约应用的场景：一个人和保险公司签订合同购买健康险，一个人从另外一个人手里购买一块土地，一个公司出售股权给另外一家公司。



什么是智能合约

- 智能合约是按照用户的需求编写代码，并部署在**区块链上的程序**。智能合约支持创建去信任化协议。这意味着合约双方通过区块链做出承诺，而无需相互了解或信任。双方确定，如果没有达到条件，合约不会执行。此外，使用智能合约不再需要中间机构，从而显著降低运行成本。

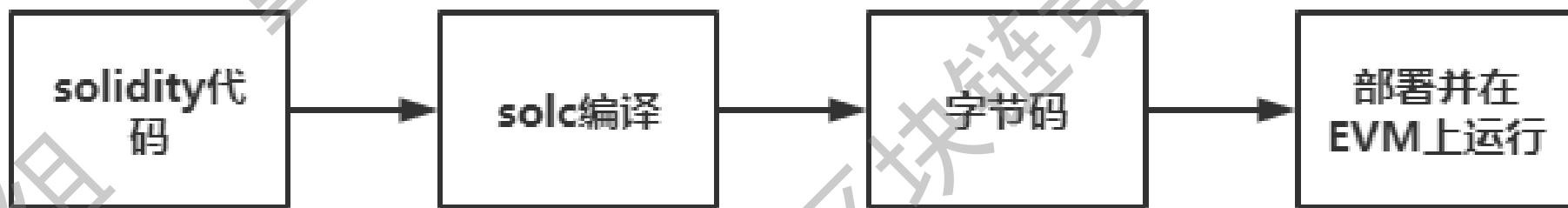


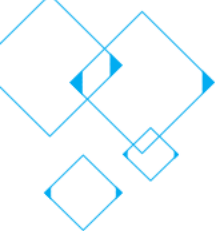
智能合约的发展史

- 智能合约概念于1994年由Nick Szabo首次提出，定义为一种旨在以信息化方式传播、验证或执行合约的**计算机协议**。智能合约允许在没有第三方的情况下进行可信交易，这些交易可追踪且不可逆转。
- 1997年，Nick Szabo对智能合约的定义进行了完善：智能合约是一套以**数字形式指定的承诺**，包括合约参与方可以在上面执行这些承诺的协议。
- 2016年，工信部在《中国区块链技术和应用发展白皮书》上是这样定义智能合约的：智能合约是一段部署在区块链上可**自动运行的程序**，涵盖范围包括编程语言、编译器、虚拟机、时间、状态机、容错机制等。

智能合约是如何运行的

- 智能合约通过EVM（以太坊虚拟机）执行，EVM作为运行智能合约的平台，并不理解Solidity代码，它只能运行称为字节码的低级指令，因此需要编译器将solidity代码转换为字节码，再将字节码部署到区块链上。
- 从代码编写到合约运行的大致流程如下





重航区块链

CONTENTS

章节

第二章 如何编写智能合约

2.1 如何编写智能合约

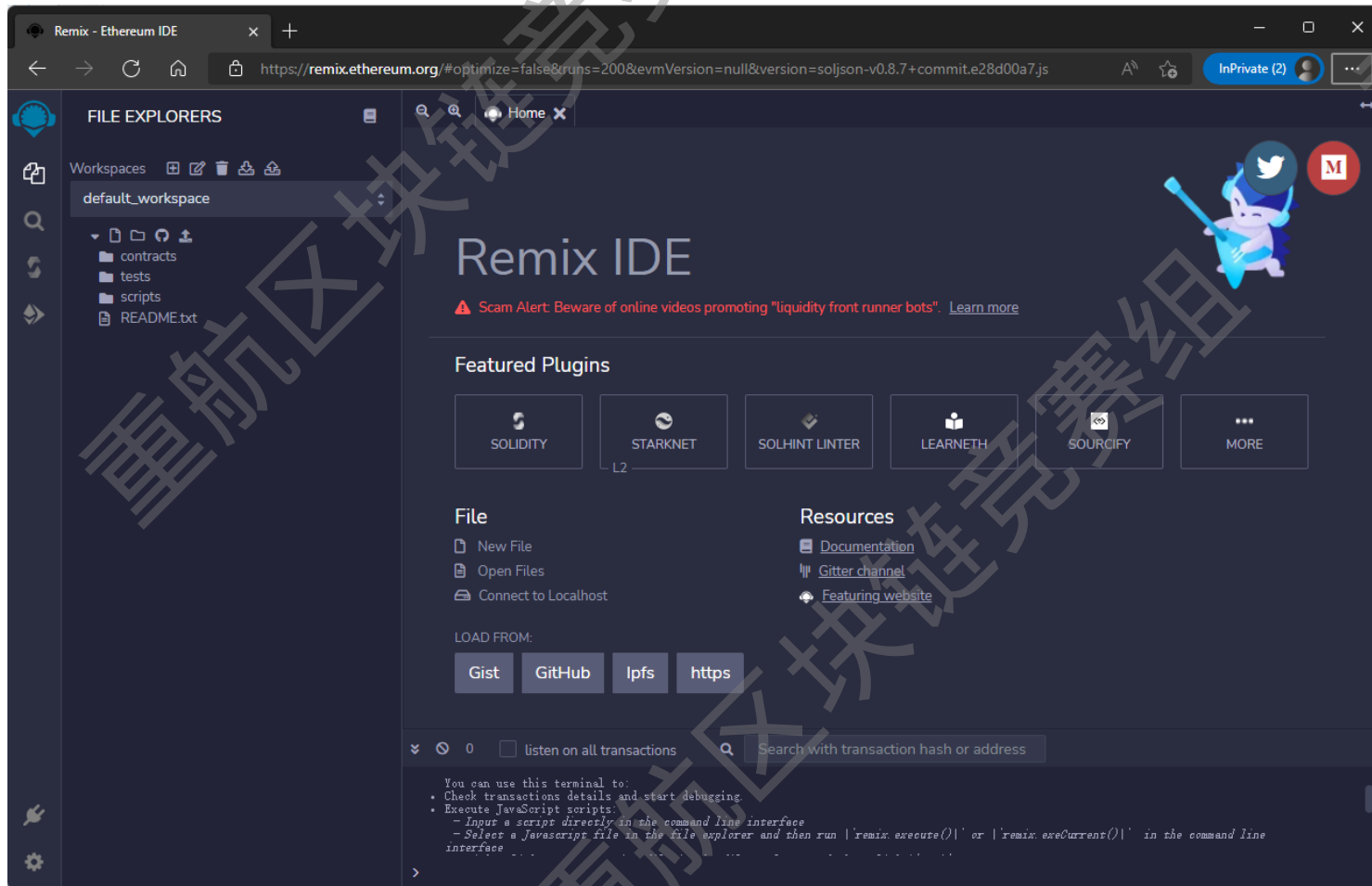


如何编写智能合约

- 编写智能合约就是编写程序，智能合约使用Solidity语言来编写，任何编辑代码的工具都可以用来编写智能合约，如IDEA、Visual Studio Code，甚至是记事本。
- 最简单的方法是使用Remix在线编辑器，通过Remix能够在浏览器上进行智能合约的创建、开发、部署和调试，甚至可以直接将智能合约部署在以太坊区块链网络上。

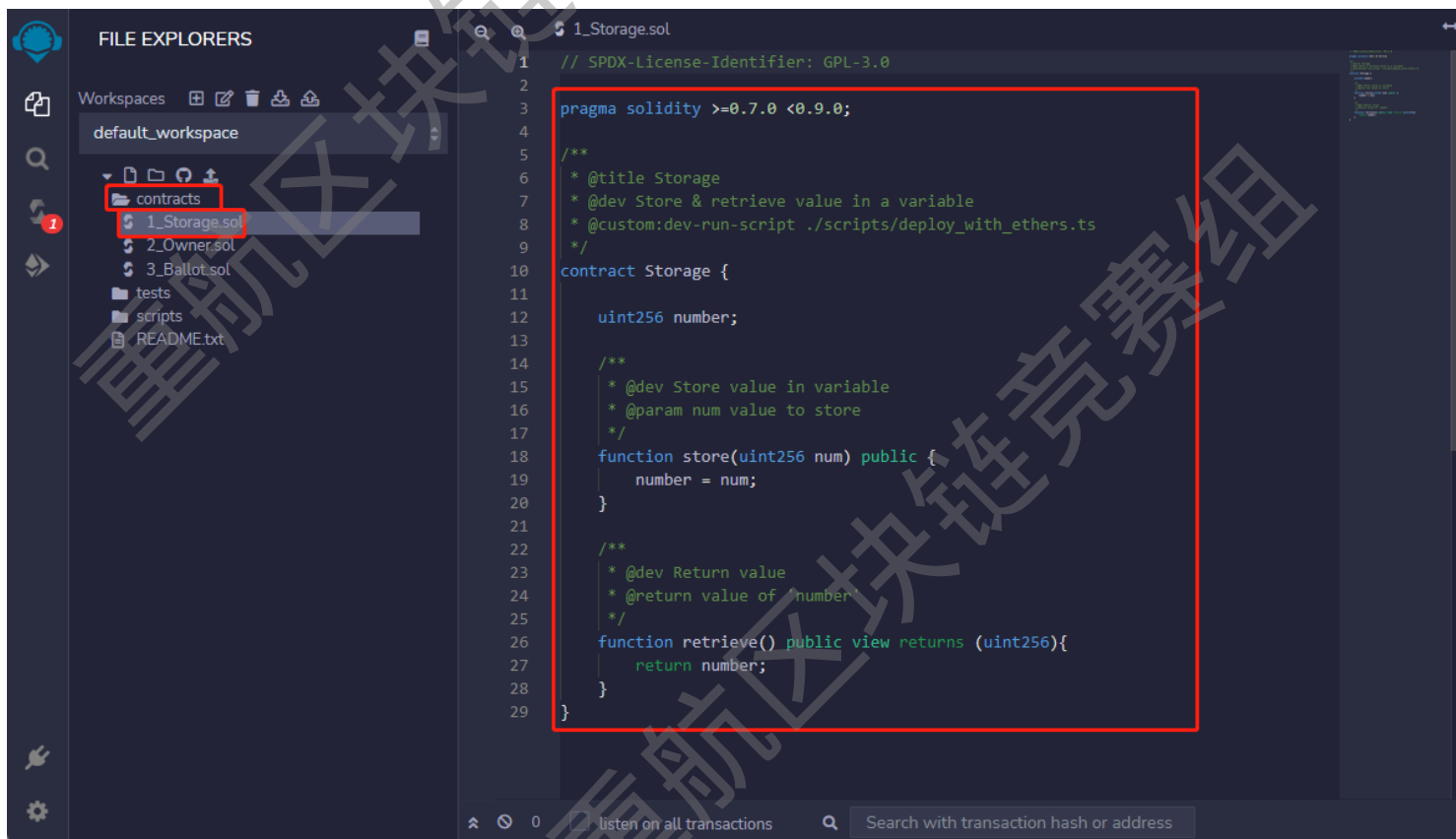
如何编写智能合约

- 在浏览器中访问<https://remix.ethereum.org>进入Remix编辑器



如何编写智能合约

- Remix编辑器中自带三个示例合约，选择“contracts”文件夹下的1_Storage.sol文件，这个合约的功能是保存和读取一个数字

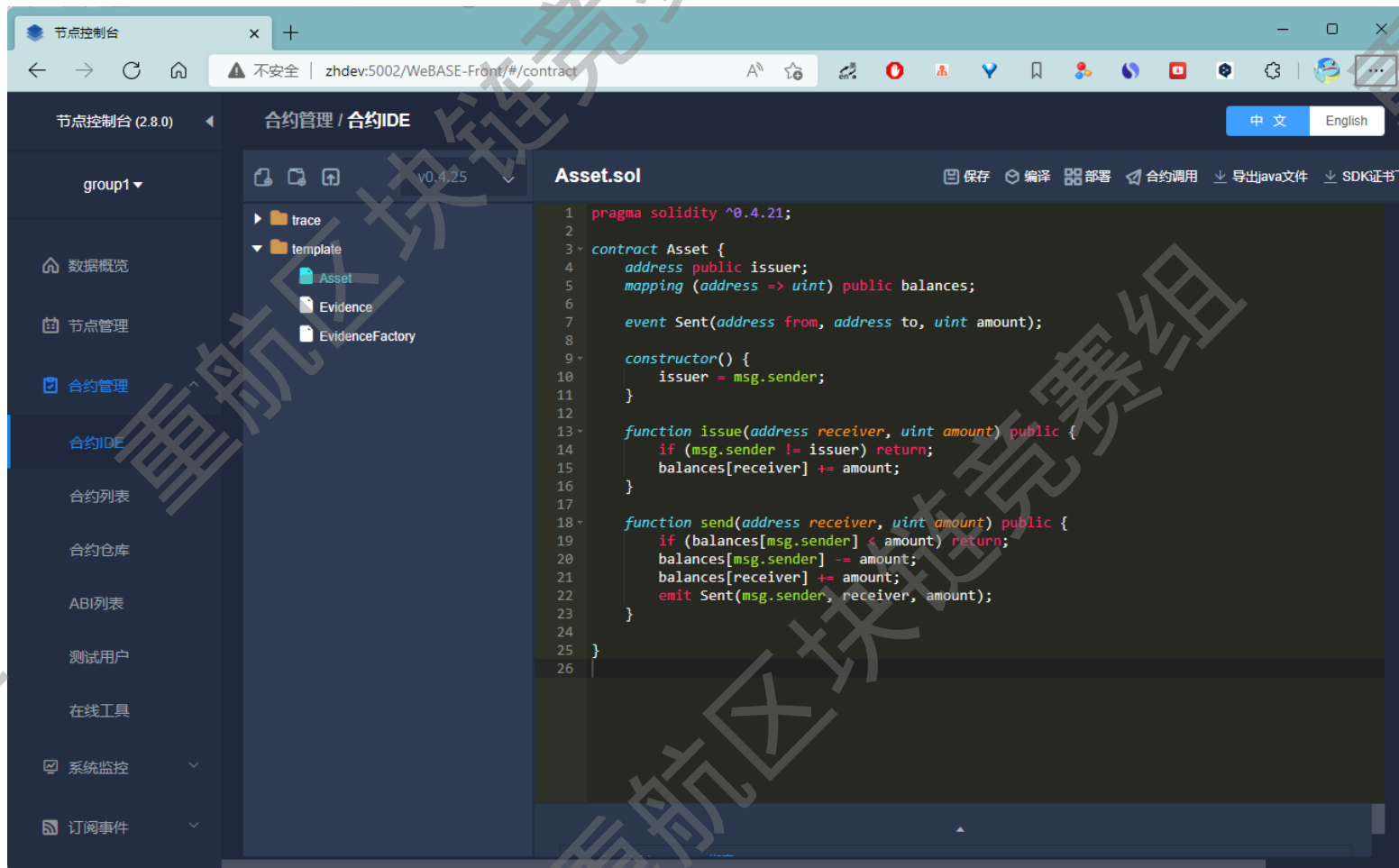


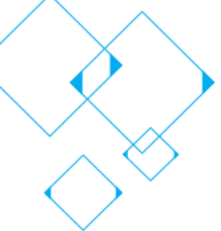
如何编写智能合约

- **Remix**开发环境的服务器在境外，访问速度比较慢，其是基于以太坊区块链网络，不便于学习，因此后续的中教学使用基于**FISCO BCOS**的**WeBASE-Front**工具。
 - **FISCO BCOS**：是由国内企业主导研发、对外开源、安全可控的企业级金融联盟链底层平台，由金链盟开源工作组协作打造，并于**2017**年正式对外开源。
 - **WeBASE-Front**：是和**FISCO-BCOS**节点配合使用的子系统，可通过**HTTP**请求和节点进行通信。另外，具备可视化控制台，可以在控制台上开发智能合约，部署合约和发送交易，并查看交易和区块详情。还可以管理私钥，对节点健康度进行监控和统计。

如何编写智能合约

- WeBASE-Front工具编写智能合约界面，支持中英文。





第三章 开发环境搭建

章节

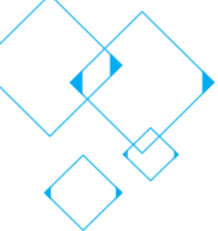
3.1 环境要求

3.2 安装依赖

3.3 创建操作目录，下载安装脚本

3.4 搭建单群组4节点区块链





第三章 开发环境搭建

章节

3.5 启动单群组4节点区块链

3.6 检查区块链进程

3.7 检查日志输出

3.8 WeBASE-Front搭建

CONTENTS



环境要求

- 操作系统：Ubuntu18.04 64位
- CPU：最低1.5GHz，推荐2.4GHz及以上
- 核心数：最低1核，推荐8核及以上
- 内存：最低1G，推荐8G及以上
- 硬盘：剩余50G及以上

安装依赖

- 更新apt源并安装依赖

- apt update && apt -y install openssl curl wget unzip net-tools

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# apt update && apt -y install openssl curl wget unzip  
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease  
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]  
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
```

创建操作目录，下载安装脚本

- 创建并进入fisco文件夹

- `mkdir -p fisco && cd fisco`

```
root@ubuntu:~#  
root@ubuntu:~# mkdir -p fisco && cd fisco  
root@ubuntu:~/fisco#
```

- 使用wget命令下载脚本

- `wget`

`http://res.zhonghui.vip/blockchain/solidity/01/resource/build_chain.sh`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# wget http://res.zhonghui.vip/blockchain/solidity/01/resource/build_chain.sh  
--2022-05-10 02:03:58-- http://res.zhonghui.vip/blockchain/solidity/01/resource/build_chain.sh
```

搭建单群组4节点区块链

● 下载FISCO BCOS二进制文件

- `wget http://res.zhonghui.vip/blockchain/solidity/01/resource/fisco-bcos.tar.gz`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# wget http://res.zhonghui.vip/blockchain/solidity/01/resource/fisco-bcos.tar.gz  
--2022-05-10 02:09:21-- http://res.zhonghui.vip/blockchain/solidity/01/resource/fisco-bcos.tar.gz
```

● 解压缩文件

- `tar -xvf fisco-bcos.tar.gz`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# tar -xvf fisco-bcos.tar.gz  
fisco-bcos
```

搭建单群组4节点区块链

- 使用build_chain.sh创建区块链，执行成功后会输出“All completed”提示

➤ `bash build_chain.sh -l 127.0.0.1:4 -p 30300,30200,8545 -e ./fisco-bcos`

```
root@ubuntu:~/fisco# bash build_chain.sh -l 127.0.0.1:4 -p 30300,20200,8545 -e ./fisco-bcos
Checking fisco-bcos binary...
Binary check passed.
=====
Generating CA key...
=====
Generating keys and certificates ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
=====
Generating configuration files ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
=====
[INFO] FISCO-BCOS Path : ./fisco-bcos
[INFO] Start Port      : 30300 20200 8545
[INFO] Server IP       : 127.0.0.1:4
[INFO] Output Dir      : /root/fisco/nodes
[INFO] CA Path          : /root/fisco/nodes/cert/
=====
[INFO] Execute the download_console.sh script in directory named by IP to get FISCO-BCOS console
e.g.  bash /root/fisco/nodes/127.0.0.1/download_console.sh -f
=====
[INFO] All completed. Files in /root/fisco/nodes
```

启动单群组4节点区块链

- 启动单群组4节点区块链

- `bash nodes/127.0.0.1/start_all.sh`

```
root@ubuntu:~/fisco# bash nodes/127.0.0.1/start_all.sh
try to start node0
try to start node1
try to start node2
try to start node3
node1 start successfully
node3 start successfully
node0 start successfully
node2 start successfully
```

- 有4条node* start successfully提示即为启动成功

检查区块链进程

- 检查4个FISCO BCOS节点进程是否正确运行

➤ `ps -ef | grep -v grep | grep fisco-bcos`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# ps -ef | grep -v grep | grep fisco-bcos  
root      3833      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node1/../../fisco-bcos -c config.ini  
root      3835      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node0/../../fisco-bcos -c config.ini  
root      3837      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node2/../../fisco-bcos -c config.ini  
root      3839      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node3/../../fisco-bcos -c config.ini  
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco#
```

- 4个FISCO BCOS节点进程都在正确运行中

检查日志输出

- 查看node0节点连接的节点数

➤ `tail -f nodes/127.0.0.1/node0/log/log* | grep connected`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# ps -ef | grep -v grep | grep fisco-bcos  
root      3833      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node1/../../fisco-bcos -c config.ini  
root      3835      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node0/../../fisco-bcos -c config.ini  
root      3837      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node2/../../fisco-bcos -c config.ini  
root      3839      0  3 02:24 pts/1    00:00:09 /root/fisco/nodes/127.0.0.1/node3/../../fisco-bcos -c config.ini  
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco#
```

- node0节点正确连接了其他3个节点

检查日志输出

- 查看node0节点是否在共识

➤ `tail -f nodes/127.0.0.1/node0/log/log* | grep +++`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# tail -f nodes/127.0.0.1/node0/log/log* | grep +++  
info|2022-05-10 02:40:56.179393|[g:1][CONSENSUS][SEALER]+++++++ Generating seal on,blkNum=1,tx=0,nodeI  
dx=3,hash=095def7f...  
info|2022-05-10 02:41:00.192599|[g:1][CONSENSUS][SEALER]+++++++ Generating seal on,blkNum=1,tx=0,nodeI  
dx=3,hash=dc4bf801...  
info|2022-05-10 02:41:04.208513|[g:1][CONSENSUS][SEALER]+++++++ Generating seal on,blkNum=1,tx=0,nodeI
```

- 有内容输出即节点在共识中，至此，FISCO BCOS搭建完成

WeBASE-Front搭建

- 配置JAVA环境，安装default-jdk

- apt -y install default-jdk

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# apt -y install default-jdk  
Reading package lists... Done
```

- 配置JAVA_HOME

- echo "export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64" >> /etc/profile

- source /etc/profile

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# echo "export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64" >> /etc/profile  
root@ubuntu:~/fisco# source /etc/profile  
root@ubuntu:~/fisco#
```

WeBASE-Front搭建

● 下载WeBASE-Front程序

- `wget http://res.zhonghui.vip/blockchain/solidity/01/resource/webase-front.zip`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# wget http://res.zhonghui.vip/blockchain/solidity/01/resource/webase-front.zip  
--2022-05-10 02:44:28-- http://res.zhonghui.vip/blockchain/solidity/01/resource/webase-front.zip
```

● 解压缩程序

- `unzip webase-front.zip`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# unzip webase-front.zip  
Archive:  webase-front.zip
```

启动WeBASE-Front服务

- 将节点文件夹中的sdk配置文件复制到webase-front中的conf文件夹

➤ `cp -r nodes/127.0.0.1/sdk/* webase-front/conf/`

```
root@ubuntu:~/fisco#  
root@ubuntu:~/fisco# cp -r nodes/127.0.0.1/sdk/* webase-front/conf/
```

- 进入webase-front文件夹并启动

➤ `cd webase-front`

➤ `bash start.sh`

```
root@ubuntu:~/fisco# cd webase-front  
root@ubuntu:~/fisco/webase-front# bash start.sh  
=====
```

Server com.webank.webase.front.Application Port 5002 ...PID(7185) [Starting]. Please check message through the log file (default path:./log/).

```
=====
```

检查WeBASE-Front服务

- 使用自带的检查脚本检查服务是否启动成功，有“is running”字样即为启动成功

➤ bash status.sh

```
root@ubuntu:~/fisco/webase-front# bash status.sh
=====
Server com.webank.webase.front.Application Port 5002 is running PID(7185)
=====
```

- 检查进程是否存在，由内容输出即为进程存在

➤ ps -ef | grep -v grep | grep webase.front

```
root@ubuntu:~/fisco/webase-front# ps -ef | grep -v grep | grep webase.front
root      7185      0 12 06:36 pts/1    00:00:43 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Djdk.tls.namedGrou
ps=secp256k1 -Dfile.encoding=UTF-8 -Xmx256m -Xms256m -Xmn128m -Xss512k -XX:MetaspaceSize=128m -XX:MaxMetaspaceS
ize=256m -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/root/fisco/webase-front/log/heap_error.log -Djava.li
brary.path=/root/fisco/webase-front/conf -cp conf/:apps/*:lib/* com.webank.webase.front.Application
```

检查WeBASE-Front服务

- 服务默认监听5002端口，检查端口是否监听，有内容返回即为处于监听状态

➤ netstat -tanl | grep 5002

```
root@ubuntu:~/fisco/webase-front#  
root@ubuntu:~/fisco/webase-front# netstat -tanl | grep 5002  
tcp        0      0 0.0.0.0:5002        0.0.0.0:*          LISTEN
```

- 检查服务日志是否记录了运行成功信息，有内容输出即为成功运行

➤ tail log/WeBASE-Front.log |grep "main run success"

```
root@ubuntu:~/fisco/webase-front#  
root@ubuntu:~/fisco/webase-front# tail log/WeBASE-Front.log |grep "main run success"  
2022-05-10 06:36:36.147 [main] INFO  Application() - main run success...
```

检查WeBASE-Front服务

- 在浏览器输入[http://127.0.0.1:5002/WeBASE-Front](http://127.0.0.1:5002/WeBASE-Front#home)即可访问WeBASE-Front服务，有如下界面显示即为搭建成功



作业

1. 什么是合约？
2. 什么是智能合约？
3. EVM是什么，它的功能是什么？
4. 编写智能合约都有哪些工具？
5. WeBASE-Front IDE默认监听在哪个端口上？

谢谢