```
15
16  //원본 MasterContainer
17  MasterContainerClass MasterContainer;
18  //두 번째 MasterContainer 생성
19  MasterContainerClass MasterContainer2;
20
21  UserInfoListClass UserInfoList;
22  map<string, string> NodeInfoList;
23  string MyNodeName = "Node2";
24
25  int main()
26  {
27      /*채굴 중간에도 마스터 컨테이너 복사가능*/
28
29      //NodeInfoList초기화
30      NodeInfoList["Node1"] = "localhost:1111";
31      NodeInfoList["Node2"] = "localhost:2222";
32      NodeInfoList["Node3"] = "localhost:3333";
33
34      //원본 MasterContainer에 블럭체인을 전파할전파할 노드인포리스트와 노드이름 설정
35      MasterContainer.SetNodeInfoList(&NodeInfoList, "");
36
37      //원본 MasterContainer에 파일해쉬 5개 삽입 채굴이 진행된다.
38      MasterContainer.InsertFileHash("1");
39      MasterContainer.InsertFileHash("2");
40      MasterContainer.InsertFileHash("3");
41      MasterContainer.InsertFileHash("4");
42      MasterContainer.InsertFileHash("5");
43
44      //원본 MasterContainer 출력
45      cout << MasterContainer.toJSON().dump(3) << endl;
46
47      //원본MasterContainer에서 출력한 JSON 을 가지고 MasterContainer2 초기화, 바로 채굴이 시작된다.
48      MasterContainer2.MakeContainerByJSON(MasterContainer.toJSON(), "", &NodeInfoList);
49
50      //채굴이 진행중인 MasterContainer2의 내용물 확인
51      cout << MasterContainer2.toJSON().dump(3) << endl;
52      std::this_thread::sleep_for(3600s);
53
54  }
55
```

MasterContainer에서 MasterContainer2를 생성하는 코드입니다.

45 : MasterContainer.toJSON() 매서드는 마스터컨테이너의 내용물을 JSON 형태로 출력합니다.

48 : MasterContainer.MakeContainerByJSON 매서드는 첫번째 인자로 위의 45줄에서 출력된 JSON 을 받고 두번째 인자로는 노드의 이름(ex Node2), 세번째 인자로는 NodeInfoList 주솟값, 세번째 인자는 생략가능 합니다만 이 채굴기의 주인(채굴 보상금을 받을 클라이언트, ex Client1) 을 지정합니다.

위의 예제에서는 5개의 파일 해쉬를 넣어서 채굴을 돌리고 그 상태에서 MasterContainer2를 생성했습니다. MasterContainer2도 48줄에서 초기화된 이후로 바로 채굴을 시작하는 것을 확인할 수 있습니다.

toJSON 매서드는 MasterContainer를 전송할 때 사용하고 MakeContainerByJSON 매서드는 수신된 JSON으로 MasterContainer를 초기화 할 때 사용하시면 될 것 같습니다.

변동 사항은 BlockChain.hpp 파일 하나만 수정했습니다. 이 부분은 여기서 확인 가능합니다.

https://github.com/CSID-DGU/2018-2-OSSP-FileEncryptionBlockChain/tree/master/BlockChain%26Buffer/%5B4%5DMasterContainer%EC%97%90JSON%EA%B8%B0%EB%8A%A5%EC%B6%94%EA%B0%80

위의 예제 실행 모습


C:¥Users¥CDM¥source¥repos¥Node2¥Debug¥Node2.exe
Finding Merkle Root....
{
"BlockCount": 0,
"BufferList": {
"data": [
[
"1",
"2",
"3",
"4",
"5"
]
],
"length": 1
},
"HashCount": 5,
"blockChain": {
"data": [
{
"data": [
"Genesis Block!"
],
"hash": "003d9dc40cad6b414d45555e4b83045cfde74bcee6b09fb42536ca25000
87fd9",
"index": 0,
"nonce": "46",
"previousHash": "00000000000000"
}
],
"length": 1
}
}
Mining start
Finding Merkle Root....
{
"BlockCount": 0,
"BufferList": {
"data": [
[
"1",
"2",
"3",
"4",
"5"
]
],
"length": 1
},
"HashCount": 5,
"blockChain": {
"data": [
{
"data": [
"Genesis Block!"
],
"hash": "003d9dc40cad6b414d45555e4b83045cfde74bcee6b09fb42536ca25000
87fd9",
"index": 0,
"nonce": "46",
"previousHash": "00000000000000"
}