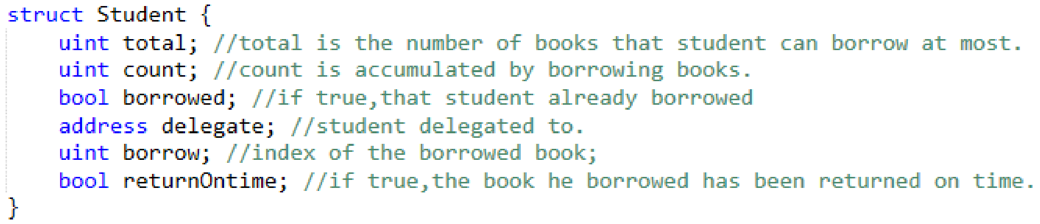
合约结构及使用说明

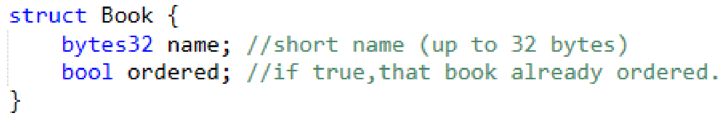
结构体声明：

学生（具体变量含义见注释）



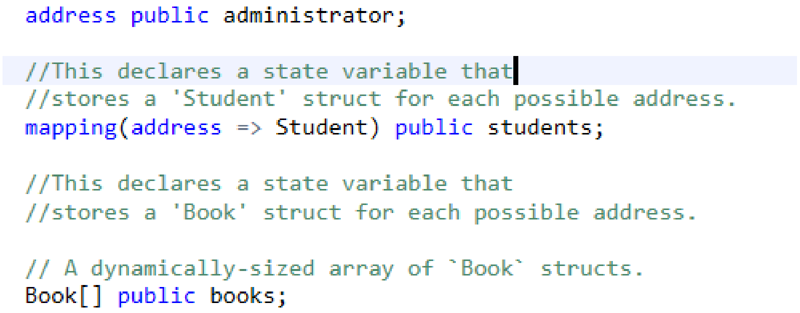
```
struct Student {
    uint total; //total is the number of books that student can borrow at most.
    uint count; //count is accumulated by borrowing books.
    bool borrowed; //if true,that student already borrowed
    address delegate; //student delegated to.
    uint borrow; //index of the borrowed book;
    bool returnOntime; //if true,the book he borrowed has been returned on time.
}
```

书籍



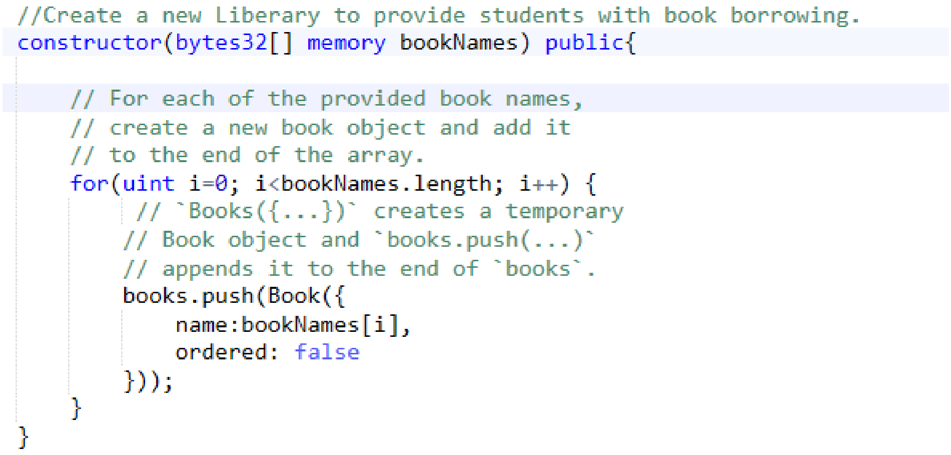
```
struct Book {
    bytes32 name; //short name (up to 32 bytes)
    bool ordered; //if true,that book already ordered.
}
```

映射关系及地址声明：



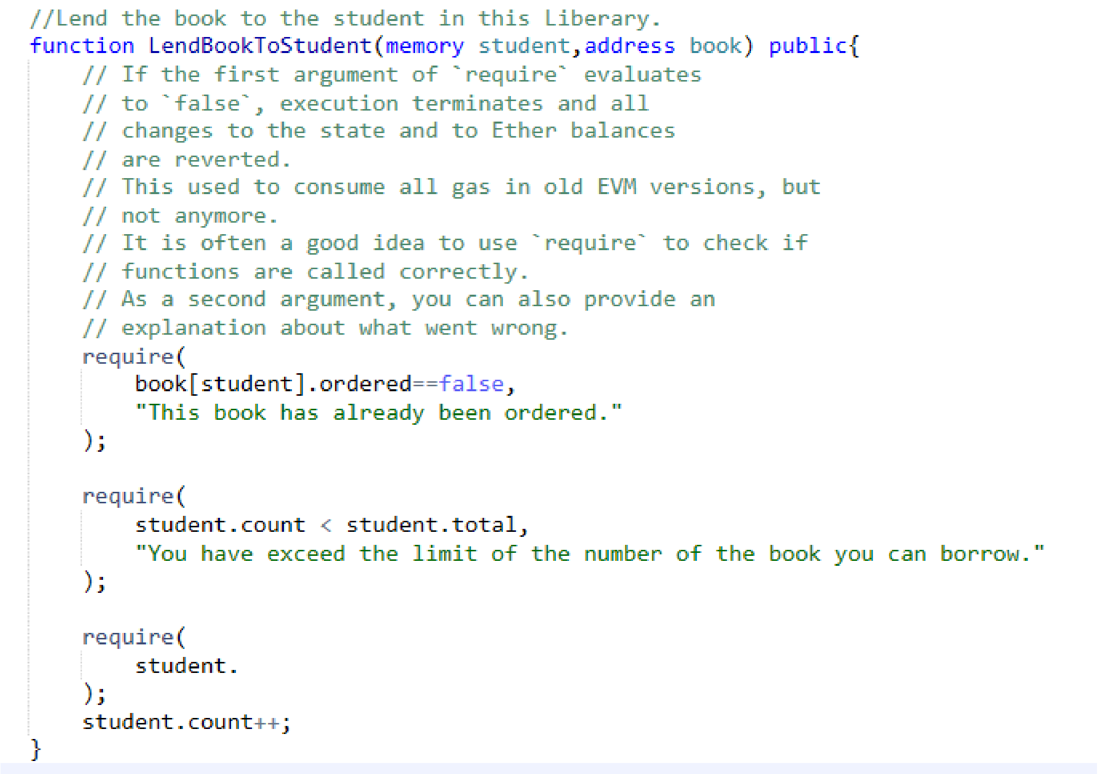
```
address public administrator;

//This declares a state variable that
//stores a 'Student' struct for each possible address.
mapping(address => Student) public students;

//This declares a state variable that
//stores a 'Book' struct for each possible address.

// A dynamically-sized array of `Book` structs.
Book[] public books;
```

Liberary 构造函数：



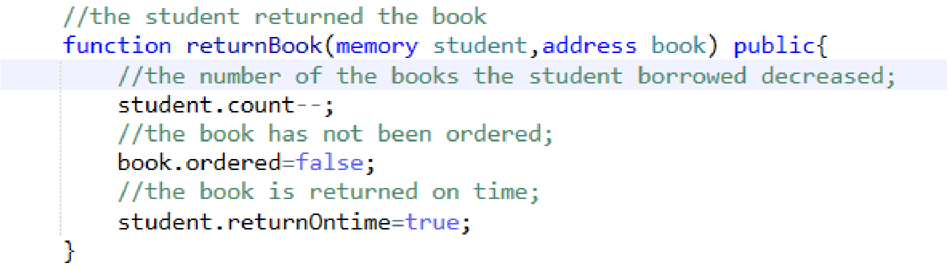
```
//Create a new Liberary to provide students with book borrowing.
constructor(bytes32[] memory bookNames) public{

    // For each of the provided book names,
    // create a new book object and add it
    // to the end of the array.
    for(uint i=0; i<bookNames.length; i++) {
        // `Books({...})` creates a temporary
        // Book object and `books.push(...)`
        // appends it to the end of `books`.
        books.push(Book({
            name:bookNames[i],
            ordered: false
        }));
    }
}
```

管理员进行学生图书借阅的函数：



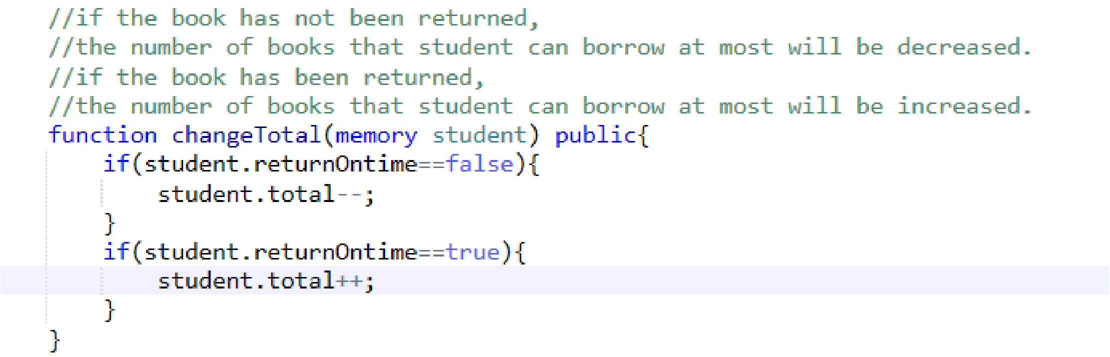
```
//Lend the book to the student in this Liberary.
function LendBookToStudent(memory student,address book) public{
    // If the first argument of `require` evaluates
    // to `false`, execution terminates and all
    // changes to the state and to Ether balances
    // are reverted.
    // This used to consume all gas in old EVM versions, but
    // not anymore.
    // It is often a good idea to use `require` to check if
    // functions are called correctly.
    // As a second argument, you can also provide an
    // explanation about what went wrong.
    require(
        book[student].ordered==false,
        "This book has already been ordered."
    );

    require(
        student.count < student.total,
        "You have exceed the limit of the number of the book you can borrow."
    );

    require(
        student.
    );
    student.count++;
}
```

图书归还:



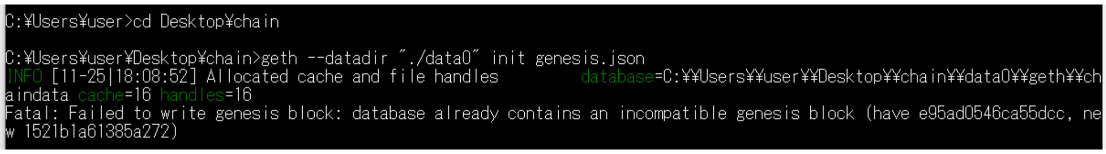
```
//the student returned the book
function returnBook(memory student,address book) public{
    //the number of the books the student borrowed decreased;
    student.count--;
    //the book has not been ordered;
    book.ordered=false;
    //the book is returned on time;
    student.returnOntime=true;
}
```

学生信用等级变化导致可借阅图书数目变化:

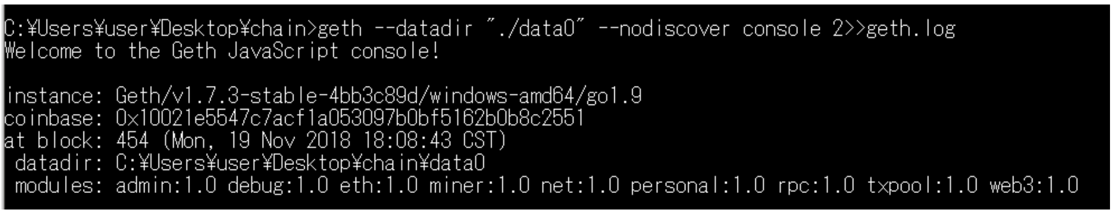


```
//if the book has not been returned,
//the number of books that student can borrow at most will be decreased.
//if the book has been returned,
//the number of books that student can borrow at most will be increased.
function changeTotal(memory student) public{
    if(student.returnOntime==false){
        student.total--;
    }
    if(student.returnOntime==true){
        student.total++;
    }
}
```

私有链搭建:

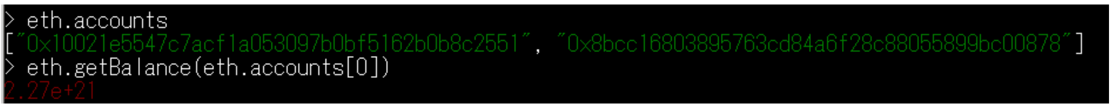


```
C:¥Users¥user>cd Desktop¥chain

C:¥Users¥user¥Desktop¥chain>geth --datadir "./data0" init genesis.json
INFO [11-25|18:08:52] Allocated cache and file handles         database=C:¥¥Users¥¥user¥¥Desktop¥¥chain¥¥data0¥¥geth¥¥ch
aindata cache=16 handles=16
Fatal: Failed to write genesis block: database already contains an incompatible genesis block (have e95ad0546ca55dcc, ne
w 1521b1a61385a272)
```



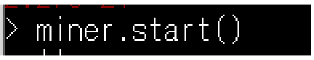
```
C:¥Users¥user¥Desktop¥chain>geth --datadir "./data0" --nodiscover console 2>>geth.log
Welcome to the Geth JavaScript console!

instance: Geth/v1.7.3-stable-4bb3c89d/windows-amd64/go1.9
coinbase: 0x10021e5547c7acf1a053097b0bf5162b0b8c2551
at block: 454 (Mon, 19 Nov 2018 18:08:43 CST)
 datadir: C:¥Users¥user¥Desktop¥chain¥data0
 modules: admin:1.0 debug:1.0 eth:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
```

查看账户余额：

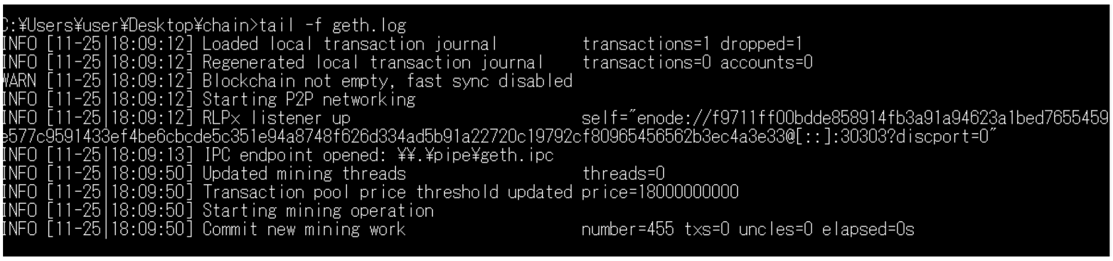


```
> eth.accounts
["0x10021e5547c7acf1a053097b0bf5162b0b8c2551", "0x8bcc16803895763cd84a6f28c88055899bc00878"]
> eth.getBalance(eth.accounts[0])
2.27e+21
```

开启挖矿：

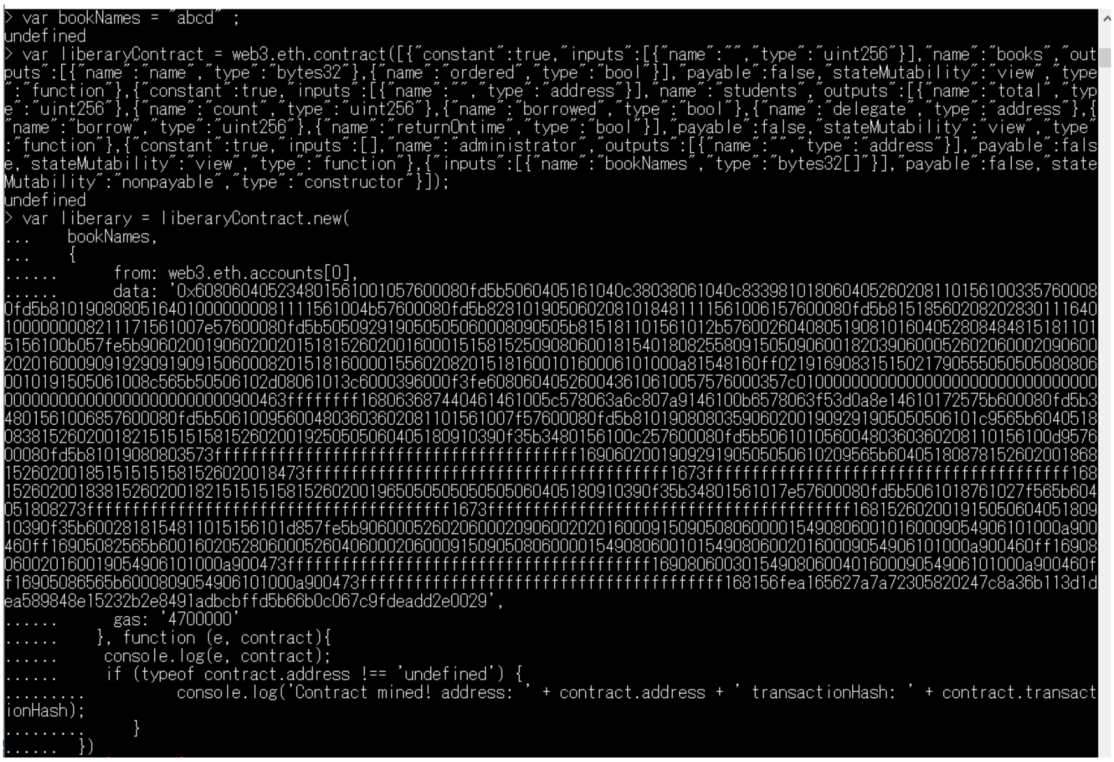


```
> miner.start()
```



```
C:¥Users¥user¥Desktop¥chain>tail -f geth.log
INFO [11-25|18:09:12] Loaded local transaction journal         transactions=1 dropped=1
INFO [11-25|18:09:12] Regenerated local transaction journal    transactions=0 accounts=0
WARN [11-25|18:09:12] Blockchain not empty, fast sync disabled
INFO [11-25|18:09:12] Starting P2P networking
INFO [11-25|18:09:12] RLPx listener up                         self="enode://f9711ff00bdde858914fb3a91a94623a1bed7655459
e577c9591433ef4be6cbcde5c351e94a8748f626d334ad5b91a22720c19792cf80965456562b3ec4a3e33@[::]:30303?discport=0"
INFO [11-25|18:09:13] IPC endpoint opened: ¥¥.¥pipe¥geth.ipc
INFO [11-25|18:09:50] Updated mining threads                   threads=0
INFO [11-25|18:09:50] Transaction pool price threshold updated price=18000000000
INFO [11-25|18:09:50] Starting mining operation
INFO [11-25|18:09:50] Commit new mining work                   number=455 txs=0 uncles=0 elapsed=0s
```

合约部署：



```
> var bookNames = "abcd" ;
undefined
> var liberaryContract = web3.eth.contract([{"constant":true,"inputs":[{"name":"","type":"uint256"}],"name":"books","out
puts":[{"name":"name","type":"bytes32"},{"name":"ordered","type":"bool"}],"payable":false,"stateMutability":"view","type
":"function"},{"constant":true,"inputs":[{"name":"","type":"address"}],"name":"students","outputs":[{"name":"total","typ
e":"uint256"},{"name":"count","type":"uint256"},{"name":"borrowed","type":"bool"},{"name":"delegate","type":"address"},{
"name":"borrow","type":"uint256"},{"name":"returnOntime","type":"bool"}],"payable":false,"stateMutability":"view","type"
:"function"},{"constant":true,"inputs":[],"name":"administrator","outputs":[{"name":"","type":"address"}],"payable":fals
e,"stateMutability":"view","type":"function"},{"inputs":[{"name":"bookNames","type":"bytes32[]"}],"payable":false,"state
Mutability":"nonpayable","type":"constructor"}]);
undefined
> var liberary = liberaryContract.new(
...    bookNames,
...    {
......      from: web3.eth.accounts[0],
......      data: '0x608060405234801561001057600080fd5b5060405161040c38038061040c83398101806040526020811015610033576008
0fd5b81019080805164010000000081111561004b57600080fd5b82810190506020810184811115610061576000080fd5b815185602082028301111640
10000000082111715610007e57600080fd5b505092919050505060008090505b815181101561012b57600260408051908101604052808484815181101
5156100b057fe5b9060200190602002015181526020016000151581525090806001815401808255809150509060018203906000526020600020906000
2020160009091929091909150600082015181600001556020820151816001016000610100000a81548160ff021916908315150217905550505050808060
0010191505061008c565b50506102d08061013c6000396000f3fe608060405260043610610057576000357c0100000000000000000000000000000000
00000000000000000000000000900463ffffffff1680636874404614610005c578063a6c807a9146100b6578063f53d0a8e14610172575b600080fd5b3
4801561006857600080fd5b506100956004803603602081101561007f57600080fd5b81019080803590602001909291905050506101c9565b60405 18
08381526020018215151515158152602001925050506040518091039 0f35b3480156100c257600080fd5b5061010560048036036020811015610 0d9576
00080fd5b81019080803573ffffffffffffffffffffffffffffffffffffffff1690602001909291905050506102 09565b6040518087815260200186 8
15260200185151515151581526020018473ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff168
1526020018381526020018215151515158152602001965050505050505050604051809103 90f35b34801561017e57600080fd5b5061018761027f565b604
051808273ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff168152602001915050604051809
10390f35b60028181548110151561 01d857fe5b906000526020600020906002020160009150905080600001549080600101600090549061 01000a900
460ff16905082565b6001602052806000526040600020600091509050806000015490806001015490806002016000905490610 1000a900460ff16908
060020160019054906101000a900473ffffffffffffffffffffffffffffffffffffffff16908060030154908060040160009054906101000a900460f
f16905086565b600080905490610 1000a900473ffffffffffffffffffffffffffffffffffffffff168156fea165627a7a72305820247c8a36b113d1d
ea589848e15232b2e8491adbcbffd5b66b0c067c9fdeadd2e0029',
......      gas: '4700000'
......    }, function (e, contract){
......     console.log(e, contract);
......     if (typeof contract.address !== 'undefined') {
.........         console.log('Contract mined! address: ' + contract.address + ' transactionHash: ' + contract.transact
ionHash);
........       }
......   })
```

总结：

这部分相对于初期的智能合约增加了图书归还和改变信用等级导致可借阅图书数目变化，完善了图书管理系统的各项功能。总体来说，这次利用区块链智能合约实现的图书管理系统大部分依照学校图书馆的借还功能，部署在私有链上，改变了传统图书馆的中心化。