EDU

CSDN学院　IT实战派

快速掌握Solidity语法

内建对象介绍和随机数生成方法

讲师：高野

本节目标

1. 掌握内建对象使用
2. 掌握随机数生成方法

内建对象

- block 在调用某个方法的时候，solidity 会提供一个block 的变量，把当前块的信息返回。

```
block.blockhash(uint blockNumber) returns (bytes32) 给定块的哈希 - 仅适用于256个不包括当前最新块
block.coinbase (address) 当前块矿工地址
block.difficulty (uint) 当前块难度
block.gaslimit (uint) 当前块gaslimit
block.number (uint) 当前数据块号
block.timestamp (uint) 当前块时间戳从unix纪元开始为秒
```

内建对象

- msg 在调用某个方法的时候，会给方法传递一个msg的属性，用来传递消息

```
msg.data (bytes) 完整的 calldata
msg.gas (uint) 剩余gas
msg.sender (address) 该消息（当前呼叫）的发送者
msg.sig (bytes4) 呼叫数据的前四个字节（即功能标识符）
msg.value (uint) 发送的消息的数量
```

- now (uint) 当前块时间戳（block.timestamp的别名）

内置函数（数学）

- addmod(uint x, uint y, uint k) returns (uint);
- mulmod(uint x, uint y, uint k) returns (uint);
- keccak256(...) returns (bytes32);

内置函数（调试）

- assert(bool condition);
- require(bool condition);
- revert();

1. assert是比较自信的做法，断言此事不会发生，如果发生则认罚：扣光所有gas
2. require是比较温和的做法，如果条件不满足，退回剩余的gas
3. revert是主动退回gas，用于if/else判断后使用
4. 建议多用require，少用assert
5. Assert多用于判断非状态变量
6. Assert可以考虑放在函数结尾部分用于验证之前的操作结果正确

内置函数

- solidity是面向对象的，内部有this
- selfdestruct(address) 销毁合约，address是收益人

随机数

使用keccak256函数：

```
uint random = uint(keccak256(block.difficulty,now));
```

使用blockhash：

```
uint renadom= uint(block.blockhash(block.number-1));
```

小结

1. Solidity也是面向对象的
2. 应灵活使用内建对象

布置作业

- 必做内容
- 编写获取随机数的智能合约

EDU

CSDN学院　IT实战派