Solidity Basic

Solidity 基础语法 · 小学生也能懂

什么是 Solidity ?

一句话给小学生

Solidity 是：**写给以太坊用的程式语言**。

Solidity 的特色

- 面向对象（Object-Oriented）
- 静态型别（Statically Typed）
- 专门写智慧合约
- 跑在 **EVM（Ethereum Virtual Machine）**上

小测验

Solidity 跑在哪一个虚拟机上?

- JVM
- **EVM（正确）**
- KVM

第一个 Solidity 合约

HelloWorld 合约

```
pragma solidity ^0.8.19;

contract HelloWorld {

}
```

解释

- `pragma solidity`：指定编译器版本
- `contract`：定义一个智慧合约

变量的三种类型

一句话记忆

变量看位置，不看值。

三种变量

- Local：函数内，用完就没
- State：合约内，存到区块链
- Global：EVM 自动给的资讯

变量示例（中文注释）

```
pragma solidity ^0.8.19;

contract Variables {

    // -------- 状态变量（存在区块链） --------
    uint8 public u8 = 10;
    uint public u256 = 600;
    int public i = -123;

    address public addr =
      0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c;

    bool public defaultBoo2;   // false
    uint public defaultUint;   // 0
    address public defaultAddr;// 0x000...000

    function doSomething() public {
        // -------- 局部变量 --------
        uint ui = 456;

        // -------- 全域变量 --------
        uint time = block.timestamp;
        address sender = msg.sender;
    }
}
```

观念题

State variable 是哪一种?

- 宣告在函数内
- **宣告在函数外，存在区块链上**
- 提供链资讯

Functions / If / Loop

条件与循环

```
pragma solidity ^0.8.19;

contract Conditions {
    uint public num;

    function set(uint _num) public {
        num = _num;
    }

    function get() public view returns (uint) {
        return num;
    }

    function foo(uint x) public returns (uint) {
        if (x < 10) return 0;
        else if (x < 20) return 1;
        else return 2;
    }

    function loop() public {
        for (uint i = 0; i < 10; i++) {
            if (i == 3) continue;
            if (i == 5) break;
        }
    }
}
```

关键字理解

- `public`：内外都能呼叫
- `view`：不能改状态

Array 与 String

数组与字串

```
pragma solidity ^0.8.19;

contract Array {
    string public greet = "Hello World!";
    uint[] public arr;
    uint[10] public fixedArr;

    function get(uint i) public view returns (uint) {
        return arr[i];
    }

    function getArr(uint[] memory _arr)
        public view returns (uint[] memory) {
        return _arr;
    }

    function doStuff(uint i) public {
        arr.push(i);
        arr.pop();
        uint len = arr.length;
    }
}
```

选择题

如何取得 array 长度?

- len(arr)
- **arr.length**
- arr.size()

DYOR：你一定要知道的全域变量

DYOR

- msg.sender 是什么？→ **呼叫者地址**
- block.coinbase 是什么？→ **出块矿工（验证者）地址**

本章总结

一页背完 Solidity Basic

- Solidity 跑在 EVM
- 变量分 Local / State / Global
- view 不改状态
- array 用 arr.push / arr.length
- msg.sender 是呼叫者