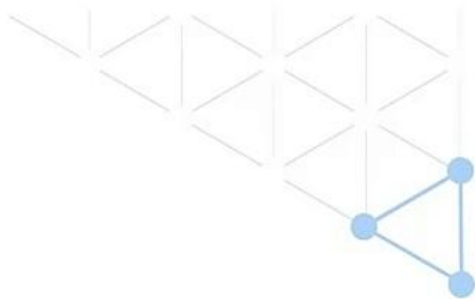


Solidity 其他类型-结构体

- 蚂蚁链《区块链系统开发与应用》A认证系列课程

课程 目标

- 了解结构体



思考一个问题

思考一个问题

- 我们已经学习了 Solidity 的一些基本数据类型，比如整型、布尔型、字符串、数组等，我们可以使用这些基本类型来定义我们的变量，例如我们想定义一个数字，可以用整型，我们想要定义一组数字，可以使用整型数组；
- 但是，无论是日常生活中，还是程序逻辑中，我们想要定义的变量，往往不能够用这些基本数据类型直接表示出来，比如一个车子，一个人；
- 所以，我们就有了一个问题：

如果我们想要定义一个自定义类型，
如 Person 类型，我们该怎么办？

结构体

结构体

- 就像C语言一样，Solidity 允许我们使用结构体来创建自定义类型（复合型数据）；
- 结构体使用 struct 关键字来定义；
- 结构体中可以存放任意类型的值；
- 结构体是引用类型；
- 结构体的定义如右侧所示：

```
struct struct_name {  
    type1 type_name_1;  
    type2 type_name_2;  
    type3 type_name_3;  
}
```

结构体

使用结构体来定义一个 Person

- 一个人可以包含如下的一些属性：
 - 名字 => string
 - 年龄 => uint
 - 身高 => uint
 - 体重 => uint
 - 性别 => boolean
 - ...



```
struct Person {  
    string name;  
    uint age;  
    uint height;  
    uint weight;  
    byte gender;  
    // ...  
}
```

a00e36c5 · 开放联盟链...

编译

Solidity

0.4.23

SolidityBasicCourse

IdentityDemo10.sol

IntDemo07.sol

BoolAndEnum08.sol

StringDemo09.sol

DataLocation11.sol

ArrayDemo12.sol

MappingDemo13.sol

StructDemo14.sol

StructDemo14.sol X

```
1 pragma solidity ^0.4.20;
2
3 contract StructDemo14 {
4
5 }
```

编译

编译详情

合约分析

调试详情

交易详情

IntDemo07.sol:5:5: The shadowed declaration is here:
uint24 y;
^-----^

1

DataLocation11.sol:15:9: Warning: Variable is declared as a storage pointer. Use an explicit "storage" keyword to silence this warning.
uint[] y = a;
^-----^

1

DataLocation11.sol:16:9: Warning: Variable is declared as a storage pointer. Use an explicit "storage" keyword to silence this warning.
uint[] y = a;
^-----^

保存

function person1

tx hash
0x22e5c44aa7b4764ab11a4ad2961c...

input

output
string: Li Si
uint256: 28
uint256: 162
uint256: 51
bytes1: 0x46

log

编译

```
StructDemo14.sol X
1 pragma solidity ^0.4.20;
2
3 contract StructDemo14 {
4     // 我们使用结构体来定义一个新的类型: Person
5     struct Person {
6         string name;
7         uint age;
8         uint height;
9         uint weight;
10        byte gender; // 'F' 表示女性; 'M' 表示男性;
11    }
12
13    Person public person = Person("Zhang San", 18, 177, 74, 'M');
14    Person public person1 = Person("Li Si", 28, 162, 51, 'F');
15 }
```

function person

tx hash
0xb991c62e339103748c2385ad57ba...

input

output
string: Zhang San
uint256: 18
uint256: 177
uint256: 74

编译

编译详情 合约分析 调试详情 交易详情

Filter

> txhash: 0xb63954ceb88b02b8ee6b4d063958b0e7594f33408ff94bc13f39334d461b1051

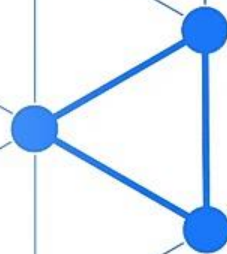
> txhash: 0x9b0ef81d2095b00a6d78483d5c6422a913fef5298368a18a564cafb07f22605b

> txhash: 0x5eed686a380fca7339e37d31bde87e307ea413a9316cf939f13ac66de5ac813

> txhash: 0x4118c9864b1008e6abe384928608cb3125d845dfeaa3b44eb579ffa99b572d98

保存

谢谢



蚂蚁集团
ANT GROUP



蚂蚁链
ANTCHAIN