

『0006』 - Solidity值类型与引用类型

孔壹学院：国内区块链职业教育领先品牌

作者：黎跃春，区块链、高可用架构工程师

微信：liyc1215 QQ群：348924182 博客：<http://liyuechun.org>

在前几节中我们初步认识了简单的以太坊智能合约的结构，生命周期以及如何通过 Ethereum Wallet 进行合约部署。

也许对于很多有开发经验的童鞋来说，大家会以为Solidity语法非常简单，其实不然，在我自己写合约的过程中，还是存在很多和其他语言不一样的坑，接下来我们将通过接下来的几篇文章为大家讲解Solidity的相关语法细节以及注意事项。

由于 Solidity 是一个 静态类型 的语言，所以编译时需明确 指定变量的类型（包括 本地变量 或 状态变量），Solidity 编程语言提供了一些基本类型(elementary types)可以用来组合成复杂类型。

我们先来看看有哪些类型属于值类型，哪些属于引用类型。

值类型(Value Type)

值类型 包含：

- 布尔(Booleans)
- 整型(Integer)
- 地址(Address)
- 定长字节数组(fixed byte arrays)
- 有理数和整型(Rational and Integer Literals, String literals)
- 枚举类型(Enums)
- 函数(Function Types)

有其他语言开发经验的童鞋都知道，值类型传值时，会临时拷贝一份内容出来，而不是拷贝指针，当你修改新的变量时，不会影响原来的变量的值。

例如：

```
int a = 100; // a == 100
int b = a;   // b == 100, a == 100
b = 300;     // b == 300, a == 100
```

由上面的数据看，执行 `b = a` 时，会将 `a` 的值临时拷贝一份传给 `b`，所以当你修改 `b` 时，其与 `a` 没关系。

引用类型(Reference Types)

引用类型 包含：

- 不定长字节数组 (bytes)
- 字符串 (string)
- 数组 (Array)
- 结构体 (Struts)

引用类型，赋值时，我们可以 值传递，也可以 引用 即地址传递，如果是值传递，和上面的案例一样，修改新变量时，不会影响原来的变量值，如果是 引用 传递，那么当你修改新变量时，原来变量的值会跟着变化，这是因为新就变量同时指向同一个地址的原因。

引用类型中如何类比值传递？

值传递伪代码（以iOS中可变字符串NSMutableString为例子）：

```
// 创建一个可变的字符串name
NSMutableString *name = [@"liyuechun" mutableCopy]; // name == "liyuechun"

NSMutableString *name1 = [name copy]; //name1 == "liyuechun", name == "liyuechun"

// PS: liyc1215 是我微信号，添加我微信拉你进区块链技术交流群
name1 = "liyc1215"; //name1 == "liyc1215", name == "liyuechun"
```

引用类型中如何类比引用传递？

```
// 创建一个可变的字符串name
NSMutableString *name = [@"liyuechun" mutableCopy]; // name == "liyuechun"

NSMutableString *name1 = name; //name1 == "liyuechun", name == "liyuechun"

// PS: liyc1215 是我微信号，添加我微信拉你进区块链技术交流群
name1 = "liyc1215"; //name1 == "liyc1215", name == "liyc1215"
```

小结

在本节中，主要是让大家知道Solidity编程语言中有哪些是**值类型**，哪些是**引用类型**，以及**值类型**和**引用类型**的简单区别。【PS：Solidity值类型中，赋值时我们始终记住传的是值，改变新变量，不会影响原来的边来干值，而引用类型就有两种可能，下一小结中，我们将重点讲解Solidity编程语言中引用类型中的**memory**和**storage**的使用，以及如何去深入理解Solidity语言中**状态变量**和**局部变量**之间的关系，以及如何去正确使用**memory**和**storage**】。

技术交流

- 区块链技术交流QQ群：348924182
- 「区块链部落」官方公众号



长按，识别二维码，加关注