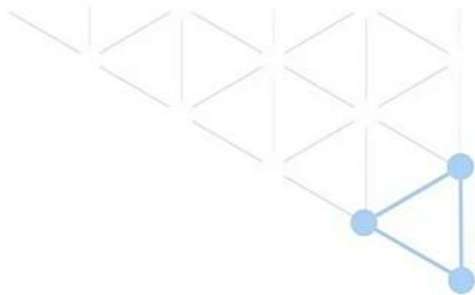


# Solidity 编码规范与关键字

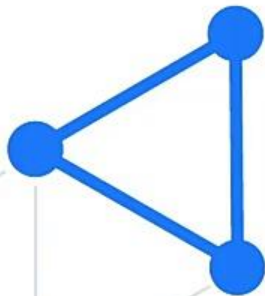
- 蚂蚁链《区块链系统开发与应用》A认证系列课程

## 课程 目标

- 了解Solidity 编码规范
- 了解Solidity 关键字



# 01 Solidity 编码规范



# Solidity 编码规范

## 编码规范是什么？

- 编码规则就是程序编码所要遵循的规则，要注意代码的**正确性、稳定性、可读性**；
- 就像工程建造中需要有工程建造标准一样，一个大型软件开发过程中，涉及到的人员数量是庞大的，这就需要有一套编码规范来对代码进行规约；
- 不同的编程语言，不同的公司，都有自己的编码规范，虽然略有不同，但是设计理念上都是相同的；
- 要避免使用不易理解的数字，用有意义的标识来替代，不要使用难懂的技巧性很高的语句；



“为什么要讲编码规范？”

## 编码规范的重要性

- 规范的编码有助于提高效率。如果没有统一的编码规范，每个人的代码风格迥异，在进行代码审查和修改 bug 的过程中势必会浪费很多时间，效率低下；
- 规范的代码可以减少 bug 的数量。往往一个项目中会出现各种各样的因为没有规范的输入输出，没有规范的错误处理或者没有规范的日志处理而出现大量 傻瓜 的 bug，而代码规范可以避免出现这样的问题；
- 规范代码可以降低维护成本，规范代码可以大大提高程序的可读性，对于后来者维护代码大有裨益；
- 规范代码有助于代码审查，规范的代码可以大大减少在代码审查的时间，提高效率；
- 规范代码也可以让大家轮岗很容易；

# Solidity 编码规范

## 代码结构

- 缩进：使用四个空格；
- 制表符和空格：优先使用空格，避免混用；
- 空行：合约声明之间留出两行空格，函数声明之间流出一行空行；
- 代码行最大长度：79（或99）字符之内（[PEP 8 recommendation](#)）；
- 表达式中的空格：
  - 除单行函数声明外，紧接着小括号，中括号或者大括号的内容应该避免使用空格；
  - 赋值或其他操作符两边各一个空格；
- 用大括号表示一个合约，库、函数和结构，应该：
  - 开括号与声明应在同一行；
  - 闭括号在与之前函数声明对应的开括号保持同一缩进级别上另起一行；
  - 开括号前应该有一个空格；



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

Tweets by @ThePSF

### The PSF

The Python Software Foundation is the organization behind Python. Become a member of the PSF and help advance the software and our mission.

Python >>> Python Developer's Guide >>> PEP Index >>> PEP 8 -- Style Guide for Python Code

## PEP 8 -- Style Guide for Python Code

PEP:	8
Title:	Style Guide for Python Code
Author:	Guido van Rossum <guido at python.org>, Barry Warsaw <barry at python.org>, Nick Coghlan <ncoghlan at gmail.com>
Status:	Active
Type:	Process
Created:	05-Jul-2001
Post- History:	05-Jul-2001, 01-Aug-2013





a00e36c5 - 开放联盟链... 编译

Solidity 0.4.23

- SolidityBasicCourse002
  - CodingAndKeyWords.sol

```
CodingAndKeyWords.sol X
1 pragma solidity ^0.4.20;
2
3 // 编码规范和关键字
4 contract PlatformInterfaceDemo24 {
5     uint testStateVariable;
6
7     /*
8     代码结构：
9     1. 使用四个空格进行缩进
10    2. 回车缩进，默认为四个空格（Cloud IDE 为我们自动完成的）
11    3. 代码行最大长度：79（或99）字符之内
12    4. 空格
13       a. 小括号，中括号，大括号的内容避免使用空格
14       b. 赋值或其他操作符两边各一个空格
15    5. 大括号：
16       a. 开括号应该和方法声明在同一行
17       b. 开括号前应该有个空格
18       c. 闭括号应该另起一行，与开括号所在行缩进相同
19    */
20
21    constructor() public {
22        testStateVariable = 1;
23    }
24
25    function getTestStateVariable() public view returns(uint) {
26        return testStateVariable;
27    }
28
29    function setTestStateVariable(uint newValue) public {
30        testStateVariable = newValue;
31    }
}
```

保存

08:16

a00e36c5 - 开放联盟链... 编译

Solidity 0.4.23

- SolidityBasicCourse002
  - CodingAndKeyWords.sol

```

CodingAndKeyWords.sol X
1 pragma solidity ^0.4.20;
2
3 // 编码规范和关键字
4 contract PlatformInterfaceDemo24 {
5     uint testStateVariable;
6
7     /*
8     代码结构：
9     1. 使用四个空格进行缩进
10    2. 回车缩进，默认为四个空格（Cloud IDE 为我们自动完成的）
11    3. 代码行最大长度：79（或99）字符之内
12    4. 空格
13       a. 小括号，中括号，大括号的内容避免使用空格
14       b. 赋值或其他操作符两边各一个空格
15    5. 大括号：
16       a. 开括号应该和方法声明在同一行
17       b. 开括号前应该有个空格
18       c. 闭括号应该另起一行，与开括号所在行缩进相同
19    */
20
21    constructor() public {
22        testStateVariable = 1;
23    }
24
25    function getTestStateVariable() public view returns(uint) {
26        return testStateVariable;
27    }
28
29    function setTestStateVariable(uint newValue) public {
30        testStateVariable = newValue;
31    }

```



保存

# Solidity 编码规范

## 代码结构

- 缩进：使用四个空格；
- 制表符和空格：优先使用空格，避免混用；
- 空行：合约声明之间留出两行空格，函数声明之间流出一行空行；
- 代码行最大长度：79（或99）字符之内（[PEP 8 recommendation](#)）；
- 表达式中的空格：
  - 除单行函数声明外，紧接着小括号，中括号或者大括号的内容应该避免使用空格；
  - 赋值或其他操作符两边各一个空格；
- 用大括号表示一个合约，库、函数和结构，应该：
  - 开括号与声明应在同一行；
  - 闭括号在与之前函数声明对应的开括号保持同一缩进级别上另起一行；
  - 开括号前应该有一个空格；

# Solidity 编码规范

## 命名规范

- 避免使用：1, l, 0 (1 和 l 容易混淆, 0 和 O 容易混淆)；
- 使用驼峰式命名且首字母大写：合约和库的名称、结构体名称、事件名称、枚举名称；
- 使用驼峰式命名且首字母小写：函数名称、函数参数名称、局部变量和状态变量；
- 命名不应该和关键字（或称保留字）相同；

UserLogin

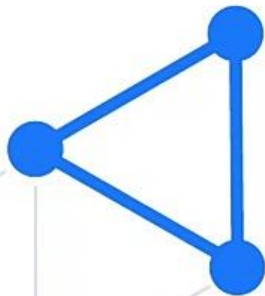
loginName

# Solidity 编码规范

## 代码结构

- 缩进：使用四个空格；
- 制表符和空格：优先使用空格，避免混用；
- 空行：合约声明之间留出两行空格，函数声明之间流出一行空行；
- 代码行最大长度：79（或99）字符之内（[PEP 8 recommendation](#)）；
- 表达式中的空格：
  - 除单行函数声明外，紧接着小括号，中括号或者大括号的内容应该避免使用空格；
  - 赋值或其他操作符两边各一个空格；
- 用大括号表示一个合约，库、函数和结构，应该：
  - 开括号与声明应在同一行；
  - 闭括号在与之前函数声明对应的开括号保持同一缩进级别上另起一行；
  - 开括号前应该有一个空格；

## 02 Solidity 关键字



# Solidity 关键字

## 关键字是什么？

- 关键字是电脑语言里**事先定义的**，有特别意义的标识符，有时又叫**保留字**，还有特别意义的变量；
- 关键字对编程语言来说有特殊意义，他们用来表示一种**数据类型**，或者表示**程序的结构**等；
- 关键字不能用作变量名、方法名、类名、包名和参数；



a00e36c5 - 开放联盟链... 编译

Solidity 0.4.23

SolidityBasicCourse002

CodingAndKeyWords.sol

```
CodingAndKeyWords.sol X
24  /*
25  命名规范:
26  1. 避免使用: 1, I, O (1 和 I 容易混淆, O 和 0 容易混淆);
27  2. 使用驼峰式命名且首字母大写: 合约和库的名称、结构体名称、事件名称、枚举名称;
28  3. 使用驼峰式命名且首字母小写: 函数名称、函数参数名称、局部变量和状态变量;
29  4. 要避免使用不易理解的数字, 用有意义的标识来替代, 不要使用难懂的技巧性很高的语句;
30  5. 命名不应该和关键字 (或称保留字) 相同;
31  */
32
33  /*
34  solidity 关键字:
35  1. 关键字不能用作变量名、方法名、类名、包名和参数
36  */
37
38  struct Person {
39      uint age;
40      string name;
41  }
42
43  enum Size {SMALL, MEDIUM, LARGE}
44
45  constructor() public {
46      testStateVariable = 1;
47  }
48
49  function getTestStateVariable() public view returns(uint returnValue) {
50      uint newValue = testStateVariable + 1;
51      return newValue;
52  }
53
54  function setTestStateVariable(uint newValue) public {
```



# Solidity 关键字

## Solidity 关键字

- 基本数据类型: uint、int、byte、bytes、string...
- 智能合约结构: contract、constructor、pragma、this、super、function、event、struct、mapping...
- 函数权限相关: public、private、internal、external
- 函数修饰相关: modifier、constant、view、pure、Storage、Memory
- 其他保留字: abstract、after、alias、apply、auto、case、catch、copyof、default、in、let、type、switch、of、null、sizeof、try...

# 谢谢