Covariant Script语法参考

## 说明

本文档格式为：正文内容 是代码，*斜体内容* 是说明，**加粗内容** 是解释

## 引入文件

import *文件目录（文字常量）* **引入一个文件**

## 变量定义

var *变量名* **定义一个变量，初始值为0**

var *变量名*=*表达式* **定义一个变量，初始值为表达式的值**

var *变量名* as *类型* **定义一个变量为指定类型**

const var *变量名*=*表达式* **定义一个常量，其值为表达式的值**

## 作用域与命名空间

block

*语句块*

end

**定义一个临时作用域**

block *命名空间名*

*语句块*

end

**定义一个命名空间**

## 分支语句

if *逻辑表达式*

*语句块*

end

**逻辑表达式的值为真则执行语句块**

if *逻辑表达式*

*语句块1*

else

*语句块2*

end

**逻辑表达式的值为真则执行语句块1**

**逻辑表达式的值为假则执行语句块2**

switch *表达式*

case *常量标签*

*语句块*

end

default

*语句块*

end

end

**执行与表达式的值相等的常量标签对应的case中的语句块**

**当无匹配的常量标签时会执行default中的语句块，如default未找到则跳出**

## 循环语句

while *逻辑表达式*

*语句块*

end

**当逻辑表达式的值为真时循环执行语句块**

loop

*语句块*

until *逻辑表达式*

end

**直到逻辑表达式的值为假时跳出循环**

loop

*语句块*

end

**循环执行语句块直到用户手动跳出**

for *变量名*=*表达式* to *表达式*

*语句块*

end

**定义一个变量在闭区间中遍历，步长为1**

for *变量名*=*表达式* to *表达式* step *表达式*

*语句块*

end

**定义一个变量在闭区间中遍历，步长为用户指定的值**

for *变量名* iterate *表达式*

*语句块*

end

**表达式的值必须是一个支持for遍历的容器**

**定义一个变量正序遍历容器**

## 函数定义

function *函数名*(*参数列表（可选）*)

*语句块*

end

**定义一个函数**

**参数列表中的参数只能指定名称，参数名不可重复，各参数之间以逗号分隔，如：**

function test(a0,a1,a2)

## 控制语句

break **跳出循环**

continue **进入下一轮循环**

return **结束函数并返回0**

return *表达式* **结束函数并返回表达式的值**

## 结构定义

struct *结构名*

*结构体*

end

**结构定义后结构名就可以作为类型名使用**

**结构体中只允许变量定义和函数定义**

## 作用域访问

*变量名* **从最上层作用域开始向下查找变量**

current.*变量名* **查找当前作用域中的变量**

global.*变量名* **查找全局作用域中的变量**

this.*变量名* **查找当前结构体中的变量**

*名称空间名*. *变量名* **查找名称空间中的变量**

*变量名*.*变量名* **查找结构体或扩展中的变量**

**注意：对于最后一种访问方法，仅变量类型为结构或支持扩展的类型时可用，如访问的是扩展中的函数，将会把点运算符左边的变量作为函数的第一个参数传入。**

**也就是说：** types.char.isspace(ch) **等价于** ch.isspace()