这是学习lua脚本语言所做的笔记，在学习任何知识的时候，在项目中，或者平时学习中遇到的问题以及解决方法以及学习到的方法都应该做好笔记，这就是学习。

在学习这门语言的时候。不能只看，不能偷懒，应该跑下代码，去改代码，实践出真知，经历模仿，改造，自己创造的过程，要喜欢上敲键盘

A = “a”

B = “b”

print(“string”,A..B)

在这种情况下，逗号分隔8个空格，两个点好分割两个空格

..一般用于给字符串做连接，而逗号和..都可以用于给字符串做连接

# 关键字用来计算字符串变量的长度

.. 用于连接字符串连接时中间间隔2个空格，而两边连接的都是数值变量时中间不空格

，则都是间隔8个字符

|  |  |  |
| --- | --- | --- |
| and | 逻辑与操作符。 若 A 为 false，则返回 A，否则返回 B。 | (A and B) 为 false。 |
| or | 逻辑或操作符。 若 A 为 true，则返回 A，否则返回 B。 | (A or B) 为 true。 |
| not | 逻辑非操作符。与逻辑运算结果相反，如果条件为 true，逻辑非为 false。 | not(A and B) 为 true。 |

|  |  |  |
| --- | --- | --- |
| == | 等于，检测两个值是否相等，相等返回 true，否则返回 false | (A == B) 为 false。 |
| ~= | 不等于，检测两个值是否相等，相等返回 false，否则返回 true | (A ~= B) 为 true。 |
| > | 大于，如果左边的值大于右边的值，返回 true，否则返回 false | (A > B) 为 false。 |
| < | 小于，如果左边的值大于右边的值，返回 false，否则返回 true | (A < B) 为 true。 |
| >= | 大于等于，如果左边的值大于等于右边的值，返回 true，否则返回 false | (A >= B) 返回 false。 |
| <= | 小于等于， 如果左边的值小于等于右边的值，返回 true，否则返回 false | (A <= B) 返回 true。 |

**运算符优先级**

从高到低的顺序：

^

not - (unary)

\* /

+ -

..

< > <= >= ~= ==

and

or

除了^和..外所有的二元运算符都是左连接的。

a+i < b/2+1 <--> (a+i) < ((b/2)+1)

5+x^2\*8 <--> 5+((x^2)\*8)

a < y and y <= z <--> (a < y) and (y <= z)

-x^2 <--> -(x^2)

x^y^z <--> x^(y^z)

# 函数

optional\_function\_scope function function\_name( argument1, argument2, argument3..., argumentn)

    function\_body

    return result\_params\_comma\_separated

end

解析：

* **optional\_function\_scope:** 该参数是可选的制定函数是全局函数还是局部函数，未设置该参数默认为全局函数，如果你需要设置函数为局部函数需要使用关键字 **local**。
* **function\_name:**指定函数名称。
* **argument1, argument2, argument3..., argumentn:**函数参数，多个参数以逗号隔开，函数也可以不带参数。
* **function\_body:**函数体，函数中需要执行的代码语句块。
* **result\_params\_comma\_separated:**函数返回值，Lua语言函数可以返回多个值，每个值以逗号隔开。

## 可变参数

Lua 函数可以接受可变数目的参数，和 C 语言类似，在函数参数列表中使用三点 **...**表示函数有可变的参数。

function add(...)

local s = 0

for i, v in ipairs{...} do --> {...} 表示一个由所有变长参数构成的数组

s = s + v

end

return s

end

print(add(3,4,5,6,7)) --->25

我们也可以通过 select("#",...) 来获取可变参数的数量:

* **select('#', …)** 返回可变参数的长度
* **select(n, …)** 用于访问 **n** 到 **select('#',…)** 的参数