------Start----primary grammar---------------------------

print("Hello,World!");

print("Hello,xiaolong!");

print("Hello,SeanWu!");

print("Show in sublime!");

print("Hello, Yuyongbing!");

function norm (x,y)

return (x^2 + y^2)^0.5

end

print("(3^2+4^2)^0.5 is equal to " .. norm(3,4));

function twice (x)

return 2\*x;

end

print("twice of 7 is :", twice(7));

colorArray = {

"red",

"yellow",

"green"

};

print(colorArray[0]);

print("color Array from here:\r\n");

for

i = 1,3 do

print(colorArray[i]);

end

dataArray = {10,9,12};

print(5+3);

print("90" + 12);

print("90" .. 12);

print("abc",tostring(139));

local k= 1;

while k<9 do

print(k);

if(dataArray[k]) then

print(dataArray[k]);

end

k=k+1;

end

------End----primary grammar-----------------------------

------Start----loop structure---------------------------

temTable = {1,3,5,7,9,110,119,200,1000};

for i,v in ipairs(temTable) do

print(i,v)

end

function values(t)

local i = 0;

return function () i = i + 1; return t[i];end

end

for element in values(temTable) do

print(element);

end

for i=1,10 do

print(i.."out");

for i=1,3 do

print(i.."in");

end

end

------End---- loop structure -----------------------------

------Start----Table-------------------------

testTable = {

["hello"] = "directory1",

["sky"] = "directory2",

["who"] = "directory3",

["sea"] = "directory4",

[30] = 9,

[31] = 12,

}

print(testTable["hello"].."\r\n");

print(testTable["sky"].."\r\n");

print(testTable["who"].."\r\n");

print(testTable["sea"].."\r\n");

print(testTable[30].."\r\n");

print(testTable[31].."\r\n");

testMultiTable = {

{30,"A",12},

{31,"B",13},

{40,"K",15},

}

local keyIdLetterRelayMap = {

-- KeyID,Letter,Relay

{"Key\_Wave",53,144},

{"Key\_1",30,146},

{"Key\_2",31,149},

{"Key\_3",32,147},

{"Key\_4",33,145},

{"Key\_5",34,160},

{"Key\_6",35,166},

{"Key\_7",36,168},

{"Key\_8",37,161},

{"Key\_9",38,187},

{"Key\_0",39,162},

{"Key\_Minus",45,176},

{"Key\_Equal",46,198},

{"Key\_Del",42,197},

{"Key\_Tab",43,148},

{"Key\_Q",20,159},

{"Key\_W",26,158},

{"Key\_E",8,153},

{"Key\_R",21,151},

{"Key\_T",23,163},

{"Key\_Y",28,164},

{"Key\_U",24,165},

{"Key\_I",12,184},

{"Key\_O",18,177},

{"Key\_P",19,178},

{"Key\_L\_Bracket",47,199},

{"Key\_R\_Bracket",48,200},

{"Key\_BK\_slash",49,192},

{"Key\_Caps",57,150},

{"Key\_A",4,152},

{"Key\_S",22,154},

{"Key\_D",7,155},

{"Key\_F",9,170},

{"Key\_G",10,191},

{"Key\_H",11,190},

{"Key\_J",13,185},

{"Key\_K",14,180},

{"Key\_L",15,181},

{"Key\_Semicolon",51,182},

{"Key\_Quota",52,179},

{"Key\_Return",40,201},

{"Key\_L\_Shift",225,156},

{"Key\_Z",29,175},

{"Key\_X",27,172},

{"Key\_C",6,167},

{"Key\_V",25,171},

{"Key\_B",5,183},

{"Key\_N",17,188},

{"Key\_M",16,186},

{"Key\_Comma",54,193},

{"Key\_Point",55,194},

{"Key\_FW\_Slash",56,195},

{"Key\_R\_Shift",229,196},

{"Key\_Globe",3,157},

{"Key\_L\_Ctrl",224,173},

{"Key\_L\_Alt",226,174},

{"Key\_L\_Command",227,169},

{"Key\_Space",44,189},

{"Key\_R\_Command",231,202},

{"Key\_R\_ALt",230,203},

{"Key\_L\_Arrow",80,204},

{"Key\_U\_Arrow",82,205},

{"Key\_D\_Arrow",81,207},

{"Key\_R\_Arrow",79,206}

}

for i = 1,64 do

for j = 1,3 do

print(keyIdLetterRelayMap[i][j]);

end

print("\r\n");

end

------End----Table-----------------------

------Start----file operation---------------------------

function ReadFile(file\_path)

local ret = nil;

local path = file\_path;

local f = io.open(path, "r");

if f == nil then

return nil, "failed to open file at: "..path;

else

ret = f:read("\*all");

f:close();

return ret;

end

end

function folder\_check(par)

--check the folder,whether it is exist or not.

local folder = par;

local ret = nil;

ret = os.execute("cd "..folder);

if tonumber(ret) > 0 then

--the folder is not exist

os.execute("mkdir "..folder);

--create the folder

end

end

testString = ReadFile("/Leo/Debug/Learn\_Lua/test.txt");

if testString == nil then

testString = "This is demo data when the txt file not found!";

end

print(testString);

if string.find(testString,"test") then

print("Test found!");

else

print("Test Not Found");

end

------End---------------------------------

------Start----String---------

sn = "ABCDEF1234567890G";

print("SN TO 0x Data");

print("sn is :"..sn);

array = {};

pstr1 = "";

for i = 1,17 do

-- print(string.byte(sn,i));

array[i] = string.byte(sn,i);

pstr1 = pstr1..array[i].." ";

end

print(pstr1);

print("Data To SN String:");

pstr2 = "";

newArray = {};

for i=1,#array do

-- print(array[i]);

newArray[i] = (string.format("%#x ",array[i]));

pstr2 = pstr2..newArray[i];

end

print(pstr2);

pstr3 = "";

for i=1,#newArray do

pstr3 = pstr3..string.format("%d",newArray[i]).." ";

end

print(pstr3);

num = string.format("%d",tonumber("7500C201000000",16));

if #num > 10 then

print(num);

end

print(string.format("%d",tonumber("7500C201000000",16)));

------End----String---------

------Start---- ---------------------------

place = string.find("hello,world!,ok","%w+,%w+,%w+");

print(place);

place = string.find("hello,worldoka1231,ok","%w+,%w+,%w+");

print(place);

place = string.find("0.3,45623,998.91","%d+%.\*%d\*,%d+%.\*%d\*,%d+%.\*%d\*");

print(place);

place = string.find("0,45623,998","%d+%.\*%d\*,%d+%.\*%d\*,%d+%.\*%d\*");

print(place);

place = string.find("01.91,45623.01,998.08","%d+%.\*%d\*,%d+%.\*%d\*,%d+%.\*%d\*");

print(place);

function \_\_split(str, reps)

local r = {};

if (str == nil) then return nil; end

string.gsub(str, "[^"..reps.."]+", function(w) table.insert(r, w) end);

return r;

end

temTable = \_\_split("12,34,56,786,90",",");

for i=1,#temTable do

print(temTable[i]);

end

------End---- ---------------------------

------Start----os and math library----------------------

print(os.date("%x"));

print(os.date("%X"));

fulltime = os.date("%Y-%m-%d%%20%H:%M:%S");

print("current full time is :"..fulltime);

print(os.date("%c"));

panelSN = "bBCDEF1234567890G";

local ProductType = "C100";

if string.char(string.byte(panelSN,1)) ~= "A" and string.char(string.byte(panelSN,1)) ~= "a" then

    ProductType = "C101";

end

print("type is "..ProductType);

trynum = math.huge;

print(trynum);

trysum = trynum + 90;

print(trysum);

trysum = trynum /190;

print(trysum);

trysum = 100/trynum;

print(trysum);

------End---- os and math library ------------------------

------Start----debug---------------

function abc()

print("abcdefg");

print(debug.getinfo(1).name);

ownName = debug.getinfo(1).name;

return ownName;

end

getName = abc();

print(getName);

------End----debug-----------------

Baidu Button BEGIN