**Unity3D读取Csv文件**

Posted on 2013年07月29日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script) /被围观 24 次

调用LoadCsv，传入csv文件路径，返回内容Table。

//csv解析 // 去掉字符串左空白 local function trim\_left(s) return string.gsub(s, "^%s+", ""); end // 去掉字符串右空白 local function trim\_right(s) return string.gsub(s, "%s+$", ""); end // 解析一行 local function parseline(line) local ret = {}; local s = line .. ","; // 添加逗号,保证能得到最后一个字段 while (s ~= "") do //print(0,s); local v = ""; local tl = true; local tr = true; while(s ~= "" and string.find(s, "^,") == nil) do //print(1,s); if(string.find(s, "^\"")) then local \_,\_,vx,vz = string.find(s, "^\"(.−)\"(.\*)"); //print(2,vx,vz); if(vx == nil) then return nil; // 不完整的一行 end // 引号开头的不去空白 if(v == "") then tl = false; end v = v..vx; s = vz; //print(3,v,s); while(string.find(s, "^\"")) do local \_,\_,vx,vz = string.find(s, "^\"(.−)\"(.\*)"); //print(4,vx,vz); if(vx == nil) then return nil; end v = v.."\""..vx; s = vz; //print(5,v,s); end tr = true; else local \_,\_,vx,vz = string.find(s, "^(.−)([,\"].\*)"); //print(6,vx,vz); //Unity3D教程手册：www.unitymanual.com if(vx~=nil) then v = v..vx; s = vz; else v = v..s; s = ""; end //print(7,v,s); tr = false; end end if(tl) then v = trim\_left(v); end if(tr) then v = trim\_right(v); end ret[table.getn(ret)+1] = v; //print(8,"ret["..table.getn(ret).."]=".."\""..v.."\""); if(string.find(s, "^,")) then s = string.gsub(s,"^,", ""); end end return ret; end //解析csv文件的每一行 local function getRowContent(file) local content; local check = false local count = 0 while true do local t = file:read() if not t then if count==0 then check = true end break end if not content then content = t else content = content..t end //Unity3D教程手册：www.unitymanual.com local i = 1 while true do local index = string.find(t, "\"", i) if not index then break end i = index + 1 count = count + 1 end if count % 2 == 0 then check = true break end end if not check then assert(1~=1) end return content end //解析csv文件 function LoadCsv(fileName) local ret = {}; local file = io.open(fileName, "r") assert(file) local content = {} while true do local line = getRowContent(file) if not line then break end table.insert(content, line) end for k,v in pairs(content) do ret[table.getn(ret)+1] = parseline(v); end file:close() return ret end //test //local t= LoadCsv("csvtesttxt.csv") //for k,v in pairs(t) do // local tt = v // local s = "" // for i,j in pairs(tt) do // s = string.format("%s,%s",s,j) // end // print ("",s) //end