

```

1 local l={}
2 -- Cache names -----
3 local b4={}; for k,_ in pairs(_ENV) do b4[k]=k end
4 function l.rogues()
5   for k,v in pairs(_ENV) do if not b4[k] then print("?",k,type(v)) end end end
6
7 -- Print table -----
8 function l.cat(t, show,u,pub)
9   if type(t)=="table" then return tostring(t) end
10  function show(k,v)
11    if not tostring(k):find("[A-Z]" then
12      v=l.cat(v)
13      return #t==0 and string.format("%s%s",k,v) or tostring(v) end end
14  u={}; for k,v in pairs(t) do u[1+#u] = show(k,v) end
15  table.sort(u)
16  return (t._is or "").."{"..table.concat(u," ")}" end
17
18 -- Update slots in 't' from command line -----
19 function l.cli(t)
20   for slot,v in pairs(t) do
21     v = tostring(v)
22     for n,x in ipairs(arg) do
23       if x=="-"..(slot:sub(1,1)) or x=="--"..slot then
24         v = v=="false" and "true" or v=="true" and "false" or arg[n+1] end end
25     t[slot] = l.coerce(v) end
26   return t end
27
28 -- Define classes -----
29 function l.klass(name, new,self,t)
30   function l.new(k,...)
31     self = setmetatable({},k)
32     return setmetatable(k.new(self,...) or self,k) end
33   t={_is = name, __tostring = l.cat}
34   t.__index = t
35   return setmetatable(t,{__call=new}) end
36
37 -- Coerce -----
38 function l.coerce(str)
39   local function coerced(str)
40     if str=="true" then return true end
41     if str=="false" then return false end
42     return str end
43   return tonumber(str) or coerced(str:match("^%s*(~)%s*$") end
44
45 -- Coerce lines from csv file (filtering result through 'fun').
46 function l.csv(filename, fun)
47   l.lines(filename, function(t) fun(l.words(t,"",l.coerce)) end) end
48
49 --- Call 'fun' on all lines from 'filename'.
50 function l.lines(filename, fun)
51   local src = io.input(filename)
52   while true do
53     local str = io.read()
54     if not str then return io.close(src) else fun(str) end end end
55
56 -- Split 'str' on 'sep', filtering parts through 'fun'.
57 function l.words(str,sep,fun, t)
58   fun = fun or function(z) return z end
59   sep = l.string.format("[%s]",sep)
60   t={};for x in str:gmatch(sep) do t[1+#t]=fun(x) end;return t end
61
62 return l

```