

Lua 数据库访问

本文主要为大家介绍 Lua 数据库的操作库：[LuaSQL](#)。他是开源的，支持的数据库有：ODBC, ADO, Oracle, MySQL, SQLite 和 PostgreSQL。

本文为大家介绍MySQL的数据库连接。

LuaSQL 可以使用 [LuaRocks](#) 来安装可以根据需要安装你需要的数据库驱动。

LuaRocks 安装方法：

```
$ wget http://luarocks.org/releases/luarocks-2.2.1.tar.gz
$ tar xzpf luarocks-2.2.1.tar.gz
$ cd luarocks-2.2.1
$ ./configure; sudo make bootstrap
$ sudo luarocks install luasocket
$ lua
Lua 5.3.0 Copyright (C) 1994-2015 Lua.org, PUC-Rio
> require "socket"
```

Window 下安装 LuaRocks：<https://github.com/keplerproject/luarocks/wiki/Installation-instructions-for-Windows>

安装不同数据库驱动：

```
luarocks install luasql-sqlite3
luarocks install luasql-postgres
luarocks install luasql-mysql
luarocks install luasql-sqlite
luarocks install luasql-odbc
```

你也可以使用源码安装方式，Lua Github 源码地址：<https://github.com/keplerproject/luasql>

Lua 连接MySql 数据库：

```
require "luasql.mysql"

--创建环境对象
env = luasql.mysql()

--连接数据库
conn = env:connect("数据库名", "用户名", "密码", "IP地址", 端口)

--设置数据库的编码格式
conn:execute"SET NAMES UTF8"

--执行数据库操作
```

```
cur = conn:execute("select * from role")

row = cur:fetch({}, "a")

-- 文件对象的创建
file = io.open("role.txt", "w+");

while row do
    var = string.format("%d %s\n", row.id, row.name)

    print(var)

    file:write(var)

    row = cur:fetch(row, "a")
end

file:close() -- 关闭文件对象
conn:close() -- 关闭数据库连接
env:close()  -- 关闭数据库环境
```

[← Lua 面向对象](#)**1 篇笔记****写笔记**

5.2 版本之后，require 不再定义全局变量，需要保存其返回值。

```
require "luasql.mysql"
```

需要写成:

```
luasql = require "luasql.mysql"
```

四重人格 1年前 (2017-10-18)