

SPASVO

泽众软件测试

- 软件测试门户
- 软件测试文库
- 开源测试资源
- 软件测试工具
- 软件测试论坛

软件测试微博

测试文库

首页

|

行业资讯

|

业务知识

|

软件测试技术

|

软件测试管理

|

移动平台测试

|

软件测试工具

|

嵌入式测试

|

软件开发专栏

|

软件测试下载

|

软件测试试题



TestCenter 测试管理工具 企业免费版正式推出！

咨询热线 021-61079698-8042

您的位置：软件测试 >> 测试技术 >> 软件开发专栏 >> 数据库

搜索

建立简易金融数据库

作者：[网络转载](#) 发布时间：[\[2015/1/20 13:28:59 \]](#) 推荐标签：[软件开发](#) [数据库](#)

金融数据一直是数据分析的重要数据来源，要做金融数据分析一定要有一个金融数据库，这篇博文就来教大家如何在自己的PC上建立一个简易金融数据库。

“工欲善其事，必先利其器”，建立数据库首先要有一个数据库软件，这里选择的是行业翘楚Oracle。幸运的是，Oracle学微软的那一套，推出了一个免费但功能有限的Oracle Express版本，虽然是功能有限但对基本的数据库操作足够了。为了更轻松的操纵数据库Oracle SQL Developer也是少不了的，读者可以根据网站的介绍下载安装这两款软件，这里不再赘述。

“水有源，树有根”，没有数据源的数据库只是一个空的容器。这里把沪深两市的股票交易数据作为数据源，下面介绍如何获得这些数据。

第一步，获得股票代码。交易所网站是获得股票代码最可靠的来源，这里给出网址，深交所：<http://www.szse.cn/main/marketdata/jypz/colist/>；上交所：<http://www.sse.com.cn/assortment/stock/list/name/>。读者可以将这些代码分别复制保存在两个文件内，这里不再赘述。

第二步，寻找网络数据源。有些大型网站提供股票数据的下载服务，比如163。这里举一个例子，在163官网的股票板块查询浦发银行（600000），可以顺藤摸瓜找到浦发银行的历史交易数据，点击旁边的“下载数据”按键就可以下载数据了，数据以csv表格的形式存储。

第三步，自动化下载数据，这也是最复杂的一步。沪深两市的可交易股票有几千只，这些股票的数据完全由人工点击网页下载是不现实的，需要实现自动化下载。这里演示如何用R语言实现数据自动下载。在第二步点击“下载数据”按键下载数据的过程中可以得到下载数据的网址，这个网址是实现自动化下载的关键。打开第二步中的网页下载数据，如果使用的是360浏览器，在下载工具中可以得到下载链接，如下图，



如果使用的是火狐浏览器，可以在下载管理器中找到下载的文件，右键“复制下载链接”，如下图，

- 测试热门文章
- 最常见的移动App Bug——崩溃的测试用例设计
 - jenkins+testng+ant+webdriver持续集成测试
 - Selenium 2.0 WebDriver 使用指南
 - 移动App测试中的最佳做法
 - 软件外包公司软件测试面试题
 - 常见软件测试工程师面试题
 - 移动app测试中的主要问题
 - 如何用Jmeter做压力测试

免费

自动化测试工具

测试管理工具

手机测试工具

免费

- 技术专题
- 2014年度精彩测试文章回顾
 - 单元测试知多少？
 - WEB安全测试知多少？
 - 云测试(Cloud Testing)
 - 灰盒测试，你知道多少？
 - 什么是群体测试？
 - 测试管理工具 TestCenter5.5 已发布
 - 2013河南省计算机学会首届软件测试技能大赛圆满成功

- 活动专题
- 河南省第二届软件测试技能大赛圆满成功
 - 2014年第八期性能测试(PerformanceRunner)免费在线培训
 - 2014年第七期自动化测试(AutoRunner)免费在线培训
 - 2014年第六期测试管理(TestCenter)免费在线培训
 - 广州大学华软软件学院_第十二届科技学术节之“泽众杯”第十一届软件设计大赛
 - 2014年第五期自动测试(AutoRunner)免费在线培训
 - 2014泽众软件参加第三届中国国防信息化技术与装备展览会圆满成功
 - 2014年第四期测试管理(TestCenter)免费在线培训



获得了下载链接后，下面分析一下链接的组成。刚才获得的链接是：
`http://quotes.money.163.com/service/chddata.html?`
`code=0600000&start=19991110&end=20141231&fields=TCLOSE;HIGH;LOW;TOPEN;LCLOSE;`
`CHG;PCHG;TURNOVER;VOTURNOVER;VATURNOVER;TCAP;MCAP`，关键字一目了然，
600000是股票代码；如果下载一个深市的股票可以发现，股票代码前的0表示沪市，1表示深市；
19991110表示开始日期（下载时可以选择是上市日还是发行日，不过这里推荐用上市日），
20141231表示截止日期；剩下的都是具体的数据项目。

新的问题来了，截止日期可以统一确定，上市日期却不能，必须分别获取股票的上市日期。来到
网页http://quotes.money.163.com/trade/ljsysj_600000.html#01b07，这是查询数据前的状态，
右键“查看网页源代码”，搜索一下“上市日”，可以发现“上市日”前面有一段代
码“`value="1999-11-10"`”，这就是上市日。

```
pe" value="1999-11-10" checked="checked">  
ue" style="width:120px" value="1999-11-10">  
pe" value="1999-11-10">上市日  
</option>
```

根据网页的特征，可以用R语言自动化的分析网页内容，获得上市日数据，代码如下，

```
#下载股票上市日期  
#download the listingdate of one security  
library(RCurl)  
SH <- readLines("SH.txt") #获取证券代码列表  
listing.date <- vector(length = length(SH))  
url.date1 <- "http://quotes.money.163.com/trade/ljsysj_"  
url.date2 <- ".html#01b07"  
for (i in 1:length(SH))  
{  
  #解析网页，得到listingdate  
  cat(i, ' ')  
  url.date <- paste(url.date1, SH[i], url.date2, sep="")  
  xx <- getURL(url.date)  
  posi <- regexpr("上市日", xx)  
  listing.date[i] <- substring(xx, posi[1]-13, posi[1]-4)  
}  
listing.date.tab <- data.frame(code=SH, listingdate=listing.date, stringsAsFactors=FALSE)  
#输出  
write.table(listing.date.tab, file="xxx.txt", sep="", quote=FALSE, row.name=FALSE)  
把下载好的数据按照市场分开，分别保存到txt文件即可，这里不再赘述。保存好的数据要稍微处理一下，日期  
的格式调整为yyyymmdd，write.table会把数据框的“列名”打印出来，列名也是要去掉的。  
有了股票代码和上市日期数据就可以自动化下载数据了，最好深市沪市分开进行，存在不同的文件夹下，R代码  
如下，  
#下载股票数据  
library(RCurl)  
#http://quotes.money.163.com/service/chddata.html?code=0600030&start=20030106&end=20140920&f  
ields=TCLOSE;HIGH;LOW;TOPEN;LCLOSE;CHG;PCHG;TURNOVER;VOTURNOVER;VATURNOVER;TCAP;MCAP  
url1 <- "http://quotes.money.163.com/service/chddata.html?code="   
market <- "1" # 1:深市, 0:沪市  
code <- "000003"  
url2 <- "&start="   
start <- "19900101"  
url3 <- "&end="   
end <- "20140920"  
url4 <- "&fields=TCLOSE;HIGH;LOW;TOPEN;LCLOSE;CHG;PCHG;TURNOVER;VOTURNOVER;VATURNOVER;T  
CAP;MCAP"  
#文件的存放路径  
file.path <- "F:/download/SS/"  
#股票代码+发行日期，格式：CODE制表符yyyymmdd  
security <- readLines("SS.txt")  
code <- vector(length = length(security))  
listingdate <- vector(length = length(security))
```

```
security.tab <- data.frame(code, listingdate, stringsAsFactors=FALSE)
for (i in 1 : length(security))
{
  security.tab[i, ] = strsplit(security[i], " ")
}
for (i in 1 : dim(security.tab)[1])
{
  code <- security.tab$code[i]
  start <- security.tab$listingdate[i]
  cat(i, " ---", code, " ")
  url <- paste(url1, market, code, url2, start, url3, end, url4, sep=" ")
  file <- paste(file.path, code, ".csv", sep=" ")
  download.file(url, destfile=file, method="auto", quiet=T)
}
```



关键词阅读

- Android SQLite数据库版本升级原理解析
- DataTable获取列名 DataTable批量更新至数据库
- 使用SQL Server Audit记录数据库变更
- 数据库范式到底是怎么个回事
- DB2数据库中的各数据类型
- Android之单元测试及数据库操作
- 一个备份MySQL数据库的简单Shell脚本
- 不用json文件不用数据库存储实现三级联动
- 将RichTextBox的内容直接写入数据库
- 玩转ASP.NET 5：初始化数据库
- 在Windows Phone 8.1中使用Sqlite数据库
- 配置管理之数据库版本控制策略
- 大数据高性能数据库Redis在Windows上的使用教程
- SQL 2005数据库转到SQL 2000的步骤
- 把实体从一个数据库复制到另一个数据库
- 修复Domino数据库视图索引的两种方法

0条评论

最新 最早 最热

还没有评论，沙发等你来抢

社交帐号登录： 微博 QQ 人人 豆瓣 更多»



说点什么吧...

发布

spasvo正在使用多说