一些网友收集的excel vba ，特别是如果抓取网页股票数据的文章。

http://blog.sina.com.cn/s/articlelist\_2127818045\_10\_1.html

http://blog.sina.com.cn/s/articlelist\_1576572380\_12\_1.html

目前市场上有很多股票行情交易软件,各种软件提供了丰富的分析和展示功能，而且基本上是免费的。

但有时我们还是想取得股票历史数据自己用Excel等工具进行查看或分析。如何获得股票历史数据呢？

最复杂也是最灵活的方法是编程实现，就是用程序到新浪搜狐的财经频道或其他财经类网站抓取并分析网页，这种方法对于大部分来说可操作性不强。

## 方法一

最直观的方法是直接到网站查看股票历史数据然后复制下来，比如到谷歌财经（http://www.google.com.hk/finance）输入相应股票代码，点击“历史价格”，就可以看到该股票历史数据。

如何快速取得股票交易历史数据

接下来可以直接复制数据了。缺点是每页最多显示200行数据，需要手工翻页。

## 方法二

最方便快捷省时省力的方法就是雅虎财经网站，它提供的接口可以直接把股票历史数据导成Excel,真实太方便了！

直接在浏览器地址中数据网址即可

http://table.finance.yahoo.com/table.csv?s=股票代码

上证股票是股票代码后面加上.ss，深证股票是股票代码后面加上.sz

深市数据链接：http://table.finance.yahoo.com/table.csv?s=000001.sz

上市数据链接：http://table.finance.yahoo.com/table.csv?s=600000.ss

另外，上证综指代码：000001.ss，深证成指代码：399001.SZ，沪深300代码：000300.ss，上证30指数代码

例如查询中国石油的历史数据，直接在浏览器中输入：http://table.finance.yahoo.com/table.csv?s=601857.ss

网站自动返回一个csv格式的文件，保存到本地即可。可以直接用Excel打开分析，也可以导入SAS、SPSS等软件进行分析。

得到的文件包括如下几个字段：

Date Open High Low Close Volume Adj Close

分别是：日期、开盘价、最高价、最低价、收盘价、收盘价、成交量、复权收盘价

好了，有时间我将在以后的文章介绍如何进行股票价格的回归分析。

【例子】 取 2012年1月1日 至 2012年4月19日的数据

http://table.finance.yahoo.com/table.csv?a=0&b=1&c=2012&d=3&e=19&f=2012&s=600000.ss

private string CreatUrl(string \_Symbol)

{

DateTime dt = DateTime.Today.AddYears(-1);

dt.AddDays(-10);

int a = dt.Month -1 ;

int b = dt.Day;

int c = dt.Year;

int d = DateTime.Today.Month -1 ;

int e = DateTime.Today.Day;

int f = DateTime.Today.Year;

return @"http://table.finance.yahoo.com/table.csv?a=" + a + @"&b=" + b + @"&c=" + c + @"&d=" + d + @"&e=" + e + @"&f=" + f + @"&s=" + \_Symbol + @"&y=0&g=d&ignore=.csv";

}

通过Yahoo Finance API获取股票数据

最近要做的一个网站需要在首页显示世界各大股票交易场所的综合指数，于是搜集了点这方面的资料，今天把研究的结果放出来。

提供财经方面讯息，做的比较好的应该是Google财经和Yahoo财经了，综合考虑还是Yahoo的接口比较好用

API使用方法比较简单，举个例子：

http://finance.yahoo.com/d/quotes.csv?s=^XAU&f=snd1l1c6

返回数据：

﻿”^XAU”,”PHLX Gold/Silver “,”11/24/2010″,213.51,”+1.01″

很典型的CSV数据，比较好处理。“s”指的是股票的代码（Symbol），“f”是要获取的数据选项，上面例子返回的数据分别是“代码”，“股票名称”，“最后更新时间”，“指数”，“变动”。完整的参数列表参见下面表格（对财经方面知道不多，所以不翻译了，以免误导别人）：

a Ask a2 Average Daily Volume a5 Ask Size

b Bid b2 Ask (Real-time) b3 Bid (Real-time)

b4 Book Value b6 Bid Size c Change & Percent Change

c1 Change c3 Commission c6 Change (Real-time)

c8 After Hours Change (Real-time) d Dividend/Share d1 Last Trade Date

d2 Trade Date e Earnings/Share e1 Error Indication (returned for symbol changed / invalid)

e7 EPS Estimate Current Year e8 EPS Estimate Next Year e9 EPS Estimate Next Quarter

f6 Float Shares g Day’s Low h Day’s High

j 52-week Low k 52-week High g1 Holdings Gain Percent

g3 Annualized Gain g4 Holdings Gain g5 Holdings Gain Percent (Real-time)

g6 Holdings Gain (Real-time) i More Info i5 Order Book (Real-time)

j1 Market Capitalization j3 Market Cap (Real-time) j4 EBITDA

j5 Change From 52-week Low j6 Percent Change From 52-week Low k1 Last Trade (Real-time) With Time

k2 Change Percent (Real-time) k3 Last Trade Size k4 Change From 52-week High

k5 Percebt Change From 52-week High l Last Trade (With Time) l1 Last Trade (Price Only)

l2 High Limit l3 Low Limit m Day’s Range

m2 Day’s Range (Real-time) m3 50-day Moving Average m4 200-day Moving Average

m5 Change From 200-day Moving Average m6 Percent Change From 200-day Moving Average m7 Change From 50-day Moving Average

m8 Percent Change From 50-day Moving Average n Name 股票名称 n4 Notes

o Open p Previous Close p1 Price Paid

p2 Change in Percent p5 Price/Sales p6 Price/Book

q Ex-Dividend Date r P/E Ratio r1 Dividend Pay Date

r2 P/E Ratio (Real-time) r5 PEG Ratio r6 Price/EPS Estimate Current Year

r7 Price/EPS Estimate Next Year s Symbol 股票代码 s1 Shares Owned

s7 Short Ratio t1 Last Trade Time t6 Trade Links

t7 Ticker Trend t8 1 yr Target Price v Volume

v1 Holdings Value v7 Holdings Value (Real-time) w 52-week Range

w1 Day’s Value Change w4 Day’s Value Change (Real-time) x Stock Exchange

y Dividend Yield

上面接口虽然好用，不过根据项目需求，在进行了进一步封装，自己写了个Widget类，方便在模板里调用

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74

class FinanceWidget { public $stocks = array(); public $htmlOptions = array(); public$data = array(); public function \_\_construct($data = array(),$htmlOptions = array()){$this->stocks = $data; $this->htmlOptions = $htmlOptions; } private function\_renderHead($htmlOptions){ $options = ''; foreach($htmlOptions as $option => $value){$options .= "$option=\"$value\" "; } echo "

$options>\n"; } private function\_renderBody($data){ foreach($data as $row){ echo '

'.$row[3].''.$row[1].''.$row[2].'

'."\n"; } } private function\_renderFoot(){ echo "

\n"; } private function getAPI(){ $IDs = join(',',$this->stocks); return "http://finance.yahoo.com/d/quotes.csv?s=$IDs&f=nl1p2"; } publicfunction run(){ $this->getStocks(); $this->\_renderHead($this->htmlOptions); $this->\_renderBody($this->data); $this->\_renderFoot(); } public function getStocks(){ $row =0; $stocks\_name = array\_keys($this->stocks); if (($handle = fopen($this->getAPI(), 'r'))!== FALSE) { while (($data = fgetcsv($handle, 1000, ",")) !== FALSE) {array\_push($data,$stocks\_name[$row]); array\_push($this->data,$data); $row++; }fclose($handle); } } }

调用的时候只需要定义你需要展示的股票，传给FinanceWidget类：

1 2 3 4 5 6 7 8 9 10 11 12 13

$stock = array( '上证指数'=>'000001.ss', '深圳成指'=>'399001.sz', '香港恒生'=>'0011.hk', '日经指数'=>'^N225', '英国FTSE'=>'^FTSE', '法国CAC'=>'^FCHI', '德国DAX'=>'^GDAXI', '纳斯达克'=>'^IXIC','道琼指数'=>'INDU', ); $widget = new FinanceWidget($metals); $widget->run();

输出结果如下：

XAU指数213.51+0.48%

HUI指数548.979+0.01%

JSE黄金2701.89+0.11%

GOX指数246.62+0.31%

TSX指数411.840.00%

当然这个类支持指定ul的html选项，作为第二个参数传递给Widget类即可，这里不多介绍，大家根据自己需要扩展即可

分享：