Homework 6: Server-side Scripting

1. Objectives

- Get experience with PHP programming language
- Get experience with Yahoo! Finance Web Services APIs
- Get experience using an XML parser in PHP.

2. Description

In this exercise, you are asked to create a webpage that allows users to search for stock quotes and financial company news using the Yahoo! Finance Web Services and Yahoo! Company News RSS feed APIs, and the results will be displayed in tabular format.
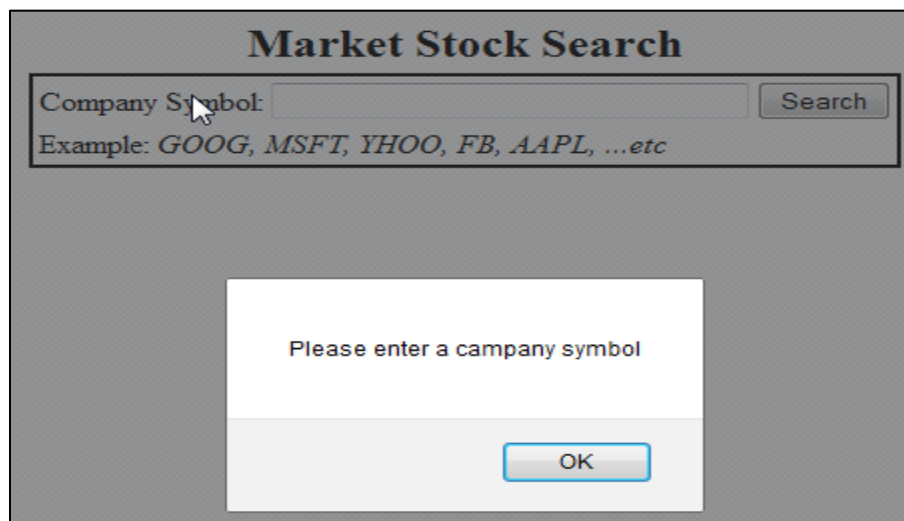
A user will first open a page, called **search_stock.php (or any valid page name)**, where he/she can enter a company symbol search about their stock quotes and their financial news. An example is shown in Figure 1.

Figure 1: Initial Screen to Enter Company Names

The user should enter at least one of the company names. If the user did not enter anything clicked on "Search", then an alert message should be shown with an appropriate message prompting the user inputting a company symbol. An example of the alert message is shown in Figure 2, and an example of valid input is shown in Figure 3.

Figure 2: Alert Message

Figure 3: Enter a company symbol and Click the "Search" Button

When the input is valid, clicking on the Search button will send a request to your server for **search_stock.php (or whatever your valid page name is)** with the form data (You can use either GET or POST in the form action). This script will grab the data sent to it and send the search information to Yahoo! Finance Web Services and Yahoo! Company News RSS feed.

For company stock quote information, the PHP script will construct a web service URL to query the Yahoo! Finance Web service using the company symbol appropriately. The following URL retrieves Google stock financial information through its company symbol (GOOG).

- http://query.yahooapis.com/v1/public/yql?q=select%20Name%2C%20Symbol%2C%20LastTradePriceOnly%2C%20Change%2C%20ChangeinPercent%2C%20PreviousClose%2C%20DaysLow%2C%20DaysHigh%2C%20OpenYearLow%2C%20YearHigh%2C%20Bid%2C%20Ask%2C%20AverageDailyVolume%2C%20OneyrTargetPrice%2C%20MarketCapitalization%20from%20yahoo.finance.quotes%20where%20symbol%3D%22**GOOG**%22&env=store%3A%2F%2Fdatatables.org%2Falltableswithkeys
- Note that URL contains the query:
    - **select** *Name, Symbol, LastTradePriceOnly, Change, ChangeinPercent, PreviousClose, DaysLow, DaysHigh, OpenYearLow, YearHigh, Bid, Ask, AverageDailyVolume, OneyrTargetPrice, MarketCapitalization **from yahoo.finance.quotes where symbol="GOOG"***.
- **GOOG** should be replaced with the value coming from the company symbol textbox

For company news, another PHP script will construct another web service URL to query the Yahoo! Company News RSS feed API using the company symbol. The following URL retrieves Google news through its company symbol (GOOG).
- http://feeds.finance.yahoo.com/rss/2.0/headline?s=**GOOG**&region=US&lang=en-US
- **GOOG** should be replaced with the value coming from the company symbol textbox

Each web service URL retrieves an XML response and your PHP script parses the returned XML files and extracts the necessary information to display data in a tabular format below the search form. A sample output is shown in Figure 4.

Figure 4: Search Result

If the entered company symbol is not correct so your page should show the error message "*Stock Information Not Available*". For example when entering the symbol ""LLLL", a sample output is shown in Figure 5.

Figure 5: Example Output for an Incorrect Company Symbol

4 RETURN XML FILES

The returned XML from Yahoo Finance contains a set of information. Figure 5 shows an example of a returned XML from Yahoo Finance. We will guide you to extract specific information to fill up the necessary data in the result data. The top part of result table displays data extracted from Yahoo! Finance web service. The headline of top part shows company name and the company symbol and three numbers. Table 1 shows the mapping between the Yahoo! Finance XML File and the result table. The value of Change started either with + or - operator. If the value starts with + you should display the second and the third numbers in the headline in green and should show the UP green arrow. Otherwise, the second and the thirds numbers should be displayed in red and DOWN arrow should also be display. The arrow images are available at

- *http://www-scf.usc.edu/~csci571/2014Spring/hw6/down_r.gif*
- *http://www-scf.usc.edu/~csci571/2014Spring/hw6/up_g.gif*

All numbers should display the thousand operator (1500 → 1,500). If the number has decimal points it should displays only two decimal points (1500.562 → 1,500.56).

```
This XML file does not appear to have any style information associated with it. The document tree is shown below.

-<query yahoo:count="1" yahoo:created="2014-02-13T21:29:00Z" yahoo:lang="en-US">
  -<results>
    -<quote>
        <Bid>1198.01</Bid>
        <Change>+13.2101</Change>
        <DaysLow>1179.66</DaysLow>
        <DaysHigh>1199.90</DaysHigh>
        <YearHigh>1191.87</YearHigh>
        <MarketCapitalization>403.0B</MarketCapitalization>
        <LastTradePriceOnly>1199.90</LastTradePriceOnly>
        <Name>Google Inc.</Name>
        <PreviousClose>1186.6899</PreviousClose>
        <ChangeinPercent>+1.11%</ChangeinPercent>
        <Symbol>GOOG</Symbol>
        <OneyrTargetPrice>1316.95</OneyrTargetPrice>
        <Ask>1198.49</Ask>
        <AverageDailyVolume>2014080</AverageDailyVolume>
    </quote>
  </results>
</query>
<!-- total: 95 -->
<!-- engine8.yql.gq1.yahoo.com -->
```

Figure 5: Example of a returned XML file from Yahoo! Finance Web Service

Result Table	Tags in Yahoo! Finance's XML
Company Name	Name
Company Symbol	Symbol
First Number in the headline	LastTradePriceOnly
Second Number in the headline	Change
Third Number in the headline	ChangeinPercent
Prev. Close	PreviousClose
Day's Range	DaysLow - DaysHigh
Open	Open
52wk Range	YearLow - YearHigh
Bid	Bid
Ask	Ask
Avg Vol	AverageDailyVolume
1y Target Est	OneyrTargetPrice
Market Cap	MarketCapitalization

Table 1: Mapping between Yahoo Finance XML File and Result Table

The returned XML from Yahoo! News RSS Feed contains <item> tags. Figure 5 shows an example of a returned XML from Yahoo! News RSS Feed. Each Item contains <title> and <link> tags. The result table shows the company news in the bottom part of the table. The news are listed in an unordered list. Each list item should display the value of <title> tag which points to the link mentioned in the <link> tag. In the unordered list, when clicking on any news item, an new tab is opened.

```
This XML file does not appear to have any style information associated with it. The document tree is shown below.

▼<rss version="2.0">
  ▼<channel>
      <title>Yahoo! Finance: GOOG News</title>
    ▼<copyright>
        Copyright (c) 2014 Yahoo! Inc. All rights reserved.
      </copyright>
      <link>http://finance.yahoo.com/q/h?s=goog</link>
      <description>Latest Financial News for Google Inc.</description>
      <language>en-US</language>
      <lastBuildDate>Thu, 13 Feb 2014 21:11:03 GMT</lastBuildDate>
    ▼<image>
      ▼<url>
          http://l.yimg.com/a/i/brand/purplelogo/uh/us/fin.gif
        </url>
        <title>Yahoo! Finance: GOOG News</title>
        <link>http://finance.yahoo.com/q/h?s=goog</link>
        <width>144</width>
        <height>45</height>
      </image>
    ▼<item>
      ▼<title>
          Google Play Music just launched a new visualization mode for Chromecast
        </title>
      ▼<link>
          http://us.rd.yahoo.com/finance/news/rss/story/*http://finance.yahoo.com/news/google-play-music-just-launched-211103627.html
        </link>
      ▼<description>
          [paidContent.org] - Here's something cheesy but fun for all you Chromecast users out there: Google Play Music now lets you stre
          TV while you cast your music. "Just in time for Valentine's Day, ...
        </description>
        <guid isPermaLink="false">yahoo_finance/3376025616</guid>
        <pubDate>Thu, 13 Feb 2014 21:11:03 GMT</pubDate>
      </item>
    ▼<item>
        <title>Motorola chief departs for Dropbox</title>
      ▼<link>
          http://us.rd.yahoo.com/finance/news/rss/story/*http://uk.finance.yahoo.com/news/motorola-chief-departs-dropbox-205200632.html
        </link>
      </link>
```

Figure 6: Example of a returned XML file from Yahoo! News RSS Feed

5. Hints

5.1. Parsing XML files in PHP

You are free to choose any XML parsing library but we recommend *SimpleXML* API.The SimpleXML library is a simple way of getting an XML element's name, attributes, and text. As of PHP 5, the SimpleXML functions are part of the PHP core. No installation is required to use these functions. The following two tables show a set of functions which you may use. For more detailed information, please read:

- http://www.w3schools.com/php/php_xml_simplexml.asp
- http://php.net/manual/en/book.simplexml.php
- http://www.w3schools.com/php/php_ref_simplexml.asp

PHP 5 SimpleXML Functions

Function	Description
__construct()	Creates a new SimpleXMLElement object
addAttribute()	Adds an attribute to the SimpleXML element
addChild()	Adds a child element the SimpleXML element

asXML()	Formats the SimpleXML object's data in XML (version 1.0)
attributes()	Returns attributes and values within an XML tag
children()	Finds the children of a specified node
count()	Counts the children of a specified node
getDocNamespaces()	Returns the namespaces DECLARED in document
getName()	Returns the name of the XML tag referenced by the SimpleXML element
getNamespaces()	Returns the namespaces USED in document
registerXPathNamespace()	Creates a namespace context for the next XPath query
saveXML()	Alias of asXML()
simplexml_import_dom()	Returns a SimpleXMLElement object from a DOM node
simplexml_load_file()	Converts an XML file into a SimpleXMLElement object
simplexml_load_string()	Converts an XML string into a SimpleXMLElement object
xpath()	Runs an XPath query on XML data

PHP 5 SimpleXML Iteration Functions

Function	Description
current()	Returns the current element
getChildren()	Returns the child elements of the current element
hasChildren()	Cheks whether the current element has children
key()	Return the current key
next()	Moves to the next element
rewind()	Rewind to the first element
valid()	Check whether the current element is valid

4. Files to submit

On your course homework page, you should update HW6 link to refer your new PHP page. Also, Submit your files (likely only a .php file) electronically to the csci571 account so that they can be graded and compared to all other students' code via the MOSS code comparison tool.

**IMPORTANT:
All discussion and explanation in Piazza related to this homework are part of the homework description. So please review all Piazza threads before finishing the assignment.