Assignment 5 (100 points)

Implement a software system, which periodically extracts the information about the most active stocks on NYSE (The New York Stock Exchange) and saves/updates the information in a database. The system also provides a web service to serve the information on a web page when a user requests such information through a web browser.

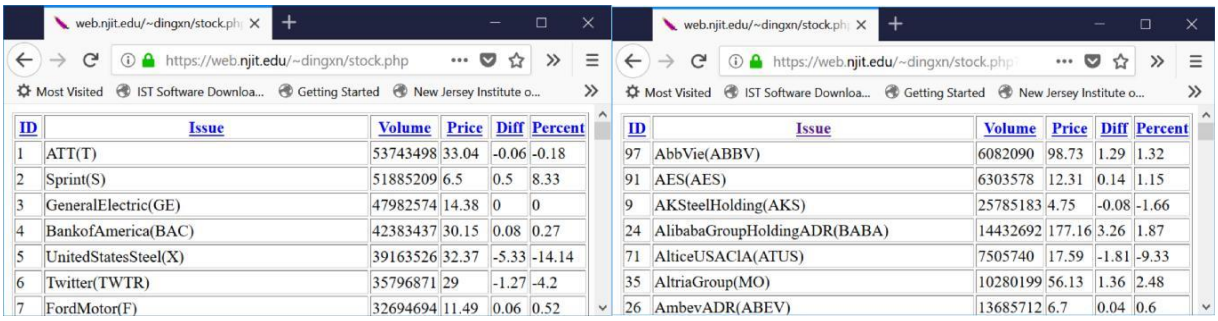
1. Objectives

1) To learn how to write a shell script.
2) To learn how to parse web pages and extract data from web.
3) To learn how to manage and access data in programs using database.
4) To learn how to generate dynamic web pages using PHP.
5) To get further understanding on HTML.

2. Overview

The system has the following five components:

1) ***A bash script*** that periodically 1) downloads web pages containing the data about the most active stocks on NYSE from ***http://online.wsj.com/mdc/public/page/2_3021-activnyse-actives.html?mod=mdc_mstactv***, and 2) executes a python program (the second component) to extract and save/update the data.
2) ***A python program*** that extracts the required data from a web page and saves the data into a mysql database (the third component).
3) ***A mysql database*** that saves the data and provides the data when it is queried.
4) ***A PHP script*** that generates the web page to display the data. To generate the web page, it needs to talk to the mysql database to obtain the required data.
5) ***A web server*** that supports dynamic web pages and PHP. It is to accept requests, runs the PHP script to generate the web page dynamically, and serve the web page to satisfy the requests.



ID	Issue	Volume	Price	Diff	Percent
1	ATT(T)	53743498	33.04	-0.06	-0.18
2	Sprint(S)	51885209	6.5	0.5	8.33
3	GeneralElectric(GE)	47982574	14.38	0	0
4	BankofAmerica(BAC)	42383437	30.15	0.08	0.27
5	UnitedStatesSteel(X)	39163526	32.37	-5.33	-14.14
6	Twitter(TWTR)	35796871	29	-1.27	-4.2
7	FordMotor(F)	32694694	11.49	0.06	0.52

ID	Issue	Volume	Price	Diff	Percent
97	AbbVie(ABBV)	6082090	98.73	1.29	1.32
91	AES(AES)	6303578	12.31	0.14	1.15
9	AKSteelHolding(AKS)	25785183	4.75	-0.08	-1.66
24	AlibabaGroupHoldingADR(BABA)	14432692	177.16	3.26	1.87
71	AlticeUSAClA(ATUS)	7505740	17.59	-1.81	-9.33
35	AltriaGroup(MO)	10280199	56.13	1.36	2.48
26	AmbevADR(ABEV)	13685712	6.7	0.04	0.6

You can use the mysql database server and/or web server maintained by NJIT. You may also setup your own servers. But, no matter which servers you use, you need to develop components 1, 2, and 4, and ensure that they interact with the servers correctly and the five components can work together to provide up-to-date stock data through the web interface.

As shown in the screenshots above, with the PHP script, a user should be able to sort the data by clicking the column title on the header row.

3. Programming Instructions

1) The Bash script

It is a very simple script. In an infinite loop, it first downloads the page following the link. You may use tools such as *wget* or *curl* to download the page. Then, it executes the python program. If you use pymysql module on an afs server, make sure that you use python3. Finally, it should execute sleep to wait for 3 minutes before the next round of iteration. So, as long as the script is running, the data in the database can be updated every 3 minutes.

2) The python program

To extract the data from the web pages, the program needs to use *re* and *minidom* module. To save/update the data in the database, the program needs to use the MySQLdb module or the PyMySQL module. You may refer to the skeleton python code provided in Moodle for this assignment. Note that the skeleton code is not a complete program. You need to understand the skeleton code and fill in the missing parts. Try to stay within the framework.The skeleton code uses PyMySQL module. If you choose to use MySQLdb module, you need to change the code accordingly. But you don't need to change the data extraction part. Refer to the slides and other online tutorials on how to use these modules.

3) The PHP program

Refer to the last example in the DB and Dynamic Web Pages. The example generates a page containing a table which is sortable if a user clicks the column titles. Your PHP program should do the similar thing. The PHP program should be put into a specific directory (e.g., ~/public_html on an afs server) if you use the NJIT web server.

Submission and Grading:

Submit 1) the Bash script, 2) the python script, 3) the PHP script, and 4) screen shots that show that your system really works.

Name your source files in the pattern SECTION#_NJITID#.{sh, py, php}. SECTION# is the three-digit section number of the CS288 section you registered (e.g., 001, 101, don't miss the leading 0s). NJITID# is the eight-digit NJIT ID (Not your UCID, Rutgers students also have NJIT IDs). The Bash script should have a file extension of .sh. The python script should have a file extension of .py. The PHP script should have a file extension of .php.

For the screen shots, open two browsers. Show in one browser the original web page that contains the data while in the second browser the page you obtained by doing all of the above. Juxtapose the two. You need to submit two screen shots. In one screen shot, the data is sorted based on ID. In the other screen shot, the data is sorted based on Issue. Name the first screenshot in the pattern SECTION#_NJITID#_ID.png. Name the first screenshot in the pattern SECTION#_NJITID#_Issue.png.

10 points for submitting the Bash script

20 points for submitting the python script

20 points for submitting the PHP script

30 points for submitting the first screenshot

20 points for submitting the second screenshot

Total 100 points.