Lab 8: Database Connectivity, Data Analysis

Due date: Wednesday, June 14 11:59pm. This is a hard deadline.

Assignment Preparation

This is a *pair programming assignment*. You choose your own partner. It can be a new partner. There are 62 students enrolled in the class. Everyone must wind up with a partner.

Lab Assignment

nyse Database

In this assignment you will be working with a new database, `nyse` created for your use on our MySQL server. The database contains information about six seven years worth of NYSE (New York Stock Exchange) securities trading. The database contains three tables: `Securities`, `Prices`, and `AdjustedPrices`.

The `Securities` table contains the description of each security (company) traded on NYSE, keyed to the NYSE ticker symbol (such as `GOOG` or `IBM`).

The `Prices` and `AdjustedPrices` tables contain information about the day-to-day prices of the securities. For each day and each security, the tables report the opening and closing prices, the highest and the lowest prices for the day, and the trading volume.

The `Prices` tables contains pricing information in terms of the actual share prices on the day of trading. The `AdjustedPrices` table contains the same information but adjusts the price of securities that underwent stock splits to show the value of the initial share of the security before any stock splits occurred. For securities that never underwent stock splits both tables should record the same data (although I have not checked).

The `CREATE TABLE` statements for these tables are shown below.

```
CREATE TABLE Securities(
  Ticker CHAR(8) PRIMARY KEY,  -- NYSE ticker symbol
  Name   Varchar(40),          -- Name of the company/security
  Sector Varchar(30),          -- Sector in which the company operates
  Industry Varchar(50),        -- Sub Industry in which the company operates
  City   Varchar(36),          -- City where the company is headquarterd
  State  Varchar(15),          -- Either US state where the company is headquartered
  Country Varchar(30),         -- Country where the company is headquartered
  StartDate DATE               -- Date the company was listed for the first time
);


-- Stock Information

CREATE TABLE Prices(
  Day DATE,            -- date for which the price data is reported
  Ticker CHAR(8),      -- ticker for which the price data is reported
  Open FLOAT,          -- price at opening
  Close FLOAT,         -- price at closing
  Low   FLOAT,         -- lowest price of the day
  High  FLOAT,         -- highest price of the day
  Volume FLOAT,        -- volume of trading
  PRIMARY KEY(Ticker, Day),
  FOREIGN KEY(Ticker) REFERENCES Securities(Ticker)
);


CREATE TABLE AdjustedPrices(
  Day DATE,            -- date for which the price data is reported
  Ticker CHAR(8),      -- ticker for which the price data is reported
  Open FLOAT,          -- price at opening
  Close FLOAT,         -- price at closing
  Low   FLOAT,         -- lowest price of the day
  High  FLOAT,         -- highest price of the day
  Volume FLOAT,        -- volume of trading
  PRIMARY KEY(Ticker, Day),
  FOREIGN KEY(Ticker) REFERENCES Securities(Ticker)
);
```

The database was constructed from the NYSE stock market dataset released on Kaggle:

https://www.kaggle.com/dgawlik/nyse

Some modifications were made to the `Securities` table: a number of columns was omitted, and the original address column was split into `City`, `State`, and `Country` columns. Please note, that there may be some outlier data in those columns regarding the country of origin of a non-US company.

For this lab you will be working witht this database. All of you have been granted `SELECT` access to the database. If you want to create additional tables or views of the data to help your analysis, you will do so in a different database.

The Task

You are acting out a role of a stock market analyst well-versed in software development and SQL. Your goal for this assignment is to prepare a number of analytical reports looking into the behavior of stock market in general, and individual securities. Each person in the class gets an individual set of securities to track (see the sheet of paper distributed to you with the assignment).

You shall write a Java program that builds the reports you need. The program can take some input (e.g., the stock ticker for the security for which the report needs to be generated, or the file name containing the list of stock tickers).

For each security assigned to your team your program shall produce one output file called `<Ticker>.html` (for example, for `GOOG` stock ticker, the filename becomes `GOOG.html`). The output shall be a well-formatted HTML file. The file shall be loadable and readable by any Web browser/HTML browser. The file shall include plain HTML markup of the information your report must convey.

To show the results of your queries (or any other data collected in a tabular form), your output shall use the `<table>` environment. For other parts of the report, you can use any reasonable formatting that plain HTML provides. If you want to automatically generate and include HTML5 graphs documenting the behavior of your stocks, it is allowed, but it is **not required**. Your output shall be a readable report with headings, explanations, and any other text that makes a typical report readable (i.e., if you simply dump a bunch of HTML tables, with no explanations, you will lose points.)

The standard is this: a person who have not read this assignment shall be able to load your HTML file into a browser and read and understand the information provided in your report.

Analytical Report

Every analytical report you generate will contain two parts: general stock market analytical data, and individual stock data. The general stock market analytical data part shall be the same in all reports your team generates. The individual data shall differ from stock ticker to stock ticker.

General Stock Market Analytical Data. This section shall report the following information:

1. Report the total number of securities traded at the start of 2016, total number of securities traded at the end of 2016, total number

of securities whose prices saw increase between the end of 2015 and the end of 2016, and the total number of securities whose prices saw decrease between the end of 2015 and the end of 2016.

2. Report the top 10 stocks that were most heavily traded in 2016.

3. For each year, report top five highest performing stocks in terms of the *absolute price increase* and top five highest performing stocks in terms of relative price increase.

4. Select 10 stocks to watch in 2017 based on stock market performance in 2016 (and possibly before). You can determine how to select these 10 stocks based on the available information.

5. Provide some general assessment of the performance of different sectors of the stock market (there are 11 sectors total), you can ignore Telecommunication Services as it only has five securities traded) in 2016. The methodology you select is up to you, your characteristics of the performance can go anywhere from `Tanking` to `Significantly outperforming`, through a number of other possible values (e.g., `Showing resilience`).

Individual Stock data. For each stock ticker you are assigned, report the following information:

1. Range of dates for which the pricing data is available[1]

2. Stock performance for every year: increase/decrease in prices year-over-year, volume of trading, average closing price in a given year, average trade volume per day.

3. For 2016 (or, for the last year for which full data exists about the stock), show the average closing price, the highest and the lowest price, the average daily trading volume by month.

4. Determine the month of best performance for your stock for each of the years. Explain the criteria used to determine the month of best performance in your HTML text, and provide the results[2]

5. For each of the following dates (for which the stock data is available): January 1, 2015, June 1, 2015, October 1, 2015, January 1, 2016, May 1, 2016, October 1, 2016 determine your position on the stock. The position may be Buy, Hold, or Sell. You can devise your own methodology for determining the position, *but this methodology cannot use future data*, i.e., in order to determine your positon for date X you cannot look at the stock performance on days X and later. You can assume

[1] Some securities/stock tickers were added to the NYSE in the middle of the time interval we are studying.

[2] This is one of the questions where each team can develop its own methodology in responding to.

that your position needs to be determined *before* the trading starts on the specified day (or the next day on which the trading happens).

6. For each of the days above, examining the next three-to-six months of available data determine if your position was correct. You can devise your own methodology for determining this, but it must fit the general understanding of what Buy, Hold, and Sell positions are:

   - Buy position is for stocks that you believe are currently underappreciated and will considerably rise in price.
   - Hold position is for stocks that do not experience significant price fluctuation at the moment, but were subject to a price increase some time in the past (i.e., they have appreciated in the past, and are not projected to lose their value in the future).
   - Sell position is for stocks that you believe are overpriced, are at the top of their price, or are experiencing price volatility without the upside of bringing their holders significant returns.

   You can take either a day-trading view of the stock value, or the long-term investor view of the stock value.

7. Compare your stock with the top performing stocks (up to five) in 2016 (if data is available, otherwise -compare for the last year for which full year of data exists). Compare the change in prices month-to-month and the volume of trading.

8. Compare your stock with another stock (other stocks) assigned to your team (same data as above). Determine which of the stocks is performing better throughout 2016 (or the last year both stocks are traded together).

Java Program

Some notes about your Java program are in order. Your program shall, at the very least, take one input parameter. This parameter shall either be a single ticker symbol, or a name of a file containing a list of ticker symbols for which reports need to be generated, or some other way of conveying what report or reports to generate. You can choose how to do this, but you must document it properly in your `README` file.

Additionally, your program shall access a `credentials.in` file located in the same directory as your main executable (if you have to move it somewhere else, *please* document it in the `README` file. The `credentials.in` file will have the following format:

```
<userId>
<password>
<database>
```

The first line of the file shall contain the MySQL loginId of the user under whose credentials the program will be accessing the MySQL server. The second line of the file shall contain the password (in plain text - I know it is not very secure, but you WILL NOT be submitting your `credentials.in` file). The third line shall contain the name of the database in which any temporary tables/views shall be created. Note, that you do not have to use the temporary database, but this is option is available to you, and the credentials file will indicate a database for the use of your program.

For example, a user `alex1` with password `abc123` wanting to use database `test1` yields the following `credentials.in` file:

```
alex1
abc123
test1
```

Submission Instructions

Submit the following:

- Your code. All of it.
- `README` file. It must, at the very least, contain the names of the students in your team, their email addresses, and any instructions for running your Java program. Additionally, I suggest that you include notes on your stock evaluation methodology (where you are given freedom to do so) in the `README` file, *even if you are providing a through explanation* in your HTML outputs.
- All `<Ticker>.html` files for the stocks assigned to you produced by your code. (Note: we will run your code, and may do so on other tickers as well, but we want you to submit the reports assigned to you).

Submit all files using `handin` as shown below.

```
handin dekhtyar lab08 <files>
```

Each Java file must contain a comment block at the top listing all members of the group.