# Business Data Management
# Lab Exercise

Instructor: Periklis A. Papakonstantinou (periklis.research@gmail.com)
Teaching assistant: Nikolas Melissaris-Papanikolaou

**Due date:** Wednesday, November 13 at 11.59pm. We will set up Blackboard to receive your assignment reports (check the Blackboard announcements).

The exercise consists of three parts. I am posting only the first one for now (because this is all I have taught until next Monday) and will update the exercise document later.

**Remark:** You are encouraged to start doing the Relational Algebra queries right away. Also, please make sure that you have installed (1) Python, (2) MySQL and (3) Apache on your machine and that you (i) were able to write a simple Python script to connect to MYSQL and (ii) you were able to run a CGI script from Apache. **All of these should be done before next Monday's lecture.**

## Part 1 (30 points)

Devise relational algebra queries for the following schema of a database storing information about daily stock prices and basic transactions made by a trading firm. You should define domains so as to make the schema complete.

- STOCK (ticker, exchange)

  ticker: the stock's ticker symbol; e.g. GOOG, AAPL, GE
  exchange: the exchange where the ticker is listed; e.g. NYSE, NASDAQ

- PRICE (ticker, date, close)

  ticker: the stock's ticker symbol
  date: the date of the price information
  close: the closing price of the stock

- BUYnSELL (buy_or_sell,ticker, date, timestamp, value, num_of_shares)

  buy_or_sell: 'BUY' or 'SELL'
  ticker: the stock's ticker symbol
  date: the date of the price information
  timestamp: time of the transaction
  value: the price of a single share
  num_of_shares: number of shares (bought or sold)

Express the following as relational algebra expressions.

i. Find the tickers and all closing prices of all stocks exchanged in 2017.

ii. Find all tickers (i.e. for all dates) whose closing price is both higher than 'IBM' on '9/20/2018' and *no higher* than 'GOOG' on '9/20/2018'.

iii. Find the tickers of all stocks that closed at the highest price on '9/20/2018'.
(we are asking for "all stocks" since there may be more than one with the same "highest price")

iv. Find the tickers of all stocks in 'NYSE' whose closing price on '9/20/2018' was either strictly below \$20 or strictly above \$100

v. Find all tickers in 'NYSE' of the stocks whose closing price showed the highest increase between '9/20/2018' and '9/21/2018' in 'NYSE' and whose closing price was (in 'NYSE') strictly above \$100 for the entire 2018
(we are asking for "all stocks" since there may be more than one with the same increase. Recall that Relational Algebra does NOT support MAX, MIN, AVG operations.)

# Example of database entries

In your report you should present your results in the following database instance. You must also add at least 5 new entries that make the queries you wrote meaningful.

(remark: when you are experimenting with your code you should change the database)

STOCK

| ticker | exchange |
|---|---|
| AAPL | NASDAQ |
| GOOG | NASDAQ |
| MSFT | NASDAQ |
| IBM | NYSE |

**Remark:** If you wish you can modify the above table by *adding* some new entries.

PRICE

| ticker | date | close |
|---|---|---|
| AAPL | 9/20/2018 | $100 |
| AAPL | 9/21/2018 | $101.5 |
| AAPL | 9/22/2018 | $106.5 |
| GOOG | 9/20/2018 | $100 |
| GOOG | 9/21/2018 | $130 |
| GOOG | 9/22/2018 | $110 |
| MSFT | 9/20/2018 | $184.5 |
| MSFT | 9/21/2018 | $188.5 |
| MSFT | 9/22/2018 | $210 |
| IBM | 9/20/2018 | $72 |
| IBM | 9/21/2018 | $70 |
| IBM | 9/22/2018 | $10 |

**Remark:** You must modify the above table by *adding* indicative entries such that the answers to the queries of Part 1 and 2 appear to be somewhat typical. Do not add more than 5 new entries. Present all added entries in your report.

BUYnSELL

| ticker | buy_or_sell | date | timestamp | price | num_of_shares |
|---|---|---|---|---|---|
| IBM | BUY | 9/20/2018 | 11:55:00 | $273 | 1100 |
| IBM | BUY | 9/21/2018 | 10:45:00 | $271 | 2400 |
| IBM | SELL | 9/22/2018 | 12:09:00 | $270.5 | 2500 |
| GOOG | BUY | 9/20/2018 | 12:22:00 | $86 | 2200 |
| GOOG | SELL | 9/20/2018 | 14:00:00 | $87 | 1000 |
| GOOG | SELL | 9/21/2018 | 10:22:00 | $87.5 | 1000 |
| GOOG | BUY | 9/21/2018 | 13:28:00 | $87 | 800 |
| GOOG | SELL | 9/22/2018 | 11:45:00 | $86 | 500 |
| AAPL | BUY | 9/20/2018 | 10:01:00 | $99 | 1000 |
| AAPL | BUY | 9/20/2018 | 11:22:00 | $99.5 | 1000 |
| AAPL | BUY | 9/21/2018 | 14:22:00 | $100 | 1000 |
| AAPL | SELL | 9/22/2018 | 14:42:00 | $103 | 3000 |
| MSFT | BUY | 9/20/2018 | 11:45:00 | $186 | 1500 |
| MSFT | SELL | 9/21/2018 | 10:45:00 | $188 | 1000 |
| MSFT | BUY | 9/22/2018 | 12:03:00 | $187 | 5000 |

**Remark:** You must modify the above table by *adding* indicative entries such that the answers to the queries of Part 1 and 2 appear to be somewhat typical. Do not add more than 5 new entries. Present all added entries in your report.