**University of Pittsburgh**

**BUSBIS 1605**

Professor David Eargle

Final Examination

Spring 2017

By submitting this exam you acknowledge that the completed exam is entirely your own work and you have complied with the Academic Honesty program of the University of Pittsburgh.

Submitted exams are analyzed to identify similarities in content and structure.

You may consult your notes and textbook and use your computer for testing SQL queries or drawing data models. You can even search the web for general help with queries, syntax, etc. But you may not consult anyone else.

Please submit this exam on CourseWeb as **several** PDFs.

* One PDF for the SQL portion, and
* One knitr’d PDF for **each** of the R questions, **or** one big pdf for all R questions. Your choice.

# Team Evaluation (1 point)

Before taking this final exam, please submit a team evaluation form to CourseWeb.

# CourseWeb Quiz (4 points)

Before doing anything else, complete the short quiz on CourseWeb called “Final – short quiz.”

One PDF for the SQL portions 1 and 2 combined

# SQL 1 -- 15 points (3 questions @ 5 points each)

1. For customers in the United States, how many customers are in each state? Sort by the count from greatest to smallest.
2. Which offices have telephone numbers with a '+1' at the **start**? Use REGEXP.  
   Hint: You can match a '+' character in REGEXP by putting it in square brackets, like this: '[+]'
3. What is the average value of orders sold for each month?

# SQL 2 -- 40 points (4 questions @ 10 points each)

1. Which Employees do not act as sales representatives for any customer?
2. What is the total value of each order with status 'Disputed'? At least report the order number and the order value. Format the value to two decimal places. Sort by order value, descending.
3. How many employees have at least three employees directly reporting to them? List each boss' full name, job title, and number of direct subordinates.
4. Write a stored procedure to update all customer's credit limit by a given percentage if they have placed a given minimum number of orders. To demonstrate that your query works, do this:
   1. Show your procedure code
   2. Show the starting point: SELECT the customer number and credit limit of customers with a minimum number of orders
   3. Provide a test CALL to your procedure
   4. Show that your procedure worked: SELECT the customer number and credit limit of customers with a minimum number of orders again

* Note: You will need to execute this on your own database so that you can create the stored procedure.

# R -- 40 points (4 questions @ 10 points each)

One knitr’d PDF for **each** of the R questions, **or** one big pdf for all R questions. Your choice.

1. Show on a Google map the top 10 customers in the USA, ranked by total number of orders they have placed.

* Hint: Form the query in MySQL Workbench first.
* Hint: You could select the top 10 by using The SQL `LIMIT` statement, or by using the `head()` function (see `?head`).
* Hint: The data.frame that you use for `marker` in the call to get\_googlemap must only have two columns (see `?get\_googlemap`.) Remember that you can use `select()` to create new data.frames from old ones.

1. Graphing
   1. Using the ClassicModels database, prepare a line graph with appropriate labels showing total payments for each month in 2004.
   2. Using the ClassicModels database, prepare a bar chart showing the number of products in each product line.
2. Using a dataset of Facebook post metrics for a company (<https://daveeargle.com/class/data/dataset_Facebook.csv>), use mapreduce to calculate the minimum, mean, maximum, and standard deviation of Total Interactions for each day of the week.

* Note: The data is related to posts published during the year of 2014 on the Facebook page of a renowned cosmetics brand. (https://archive.ics.uci.edu/ml/datasets/Facebook+metrics)
* Note: “Total Interactions” is defined as “The sum of “likes,” “comments,” and “shares” of the post.”
* Note: The file is delimited using semicolons, and the column names have spaces in them. You can clean up the column names by running:

colnames(<your dataset>) <- make.names(colnames(<your dataset>))

* Original data source:
  + <https://archive.ics.uci.edu/ml/datasets/Facebook+metrics>
  + <http://www.sciencedirect.com/science/article/pii/S0148296316000813>

1. A file on broccoli production for each state in the USA has been cleaned up for use in R (<https://daveeargle.com/class/data/broccoli.csv>). It shows the total acres of broccoli produced by each state over a number of years. Calculate the total number of acres produced for each state, sorted from greatest to least. Show `head()` for your sorted dataset.

* Note: Original data source: <http://usda.mannlib.cornell.edu/MannUsda/viewDocumentInfo.do?documentID=1816>