

Assignment 2

Sept 20th

Kyle Klassy

Bio

- Kyle Klassy
- 2nd year CS masters student
- Email: klassy@cs.wisc.edu
- Office: 4243 CS

Announcements

- Assignment 1 will be graded by Sept 23rd (Monday)
 - Send email to both Ruohui and Zhihan with grade questions
 - Joint office hour next week (Ruohui's office 9/24 2:15-3:15) to answer any further questions in person
 - Please wait 24 hours before emailing
- Piazza
 - Please try to search piazza, read entire write-up before posting a question
 - If it doesn't need to be private, don't make it private






Assignment 2: SQL

- Goal:
 - Give you practice writing SQL queries on a real database
- Description:
 - Load db file, write queries to solve a specified set of problems.
 - 5 queries to write
- Full write-up:
 - <https://kyle-klassy.github.io/cs564-fall19/assignments/p2/assignment2.pdf>

Database File

- TPC-H database
 - Benchmark dataset used to compare query times across engines.
 - “data populating the database have been chosen to have broad industry-wide relevance while maintaining a sufficient degree of ease of implementation”
 - 8 tables with primary and foreign key constraints
 - ~120 MB database file
 - http://pages.cs.wisc.edu/~klassy/courses/564/P2_SQL/TPC-H.db

3,000 GB Results

Rank	Company	System	QphH	Price/QphH	Watts/KQphH	System Availability	Database	Operating System	Date Submitted	Cluster
1		Dell PowerEdge R6415	6,053,020	.07 USD	NR	07/09/19	EXASOL 6.2	CentOS 7.6	07/09/19	Y
2	 Hewlett Packard Enterprise	HPE Proliant DL380 Gen10	1,244,450	.38 USD	NR	04/02/19	Microsoft SQL Server 2017 Enterprise Edition	SUSE Linux Enterprise Server 15	04/01/19	N
3	 Hewlett Packard Enterprise	HPE Proliant DL380 Gen10	1,090,716	.46 USD	NR	09/14/18	Microsoft SQL Server 2017 Enterprise Edition	Microsoft Windows Server 2016 Standard Edition	09/14/18	N
4	 CISCO	Cisco UCS C240 M5	1,029,593	.47 USD	NR	06/12/18	Microsoft SQL Server 2017 Enterprise Edition	Red Hat Enterprise Linux Server 7.3	06/12/18	N
5	 Hewlett Packard Enterprise	HP Proliant DL380 Gen10	1,014,374	.53 USD	NR	10/31/17	Microsoft SQL Server 2017 Enterprise Edition	Microsoft Windows Server 2016 Standard Edition	10/31/17	N

Query 1

- Produce a list of the 20 largest orders in the database, where size is defined as Σ (quantity * (extended price - discount)). Order the output by decreasing order size.
- ~~Please **DO NOT** use a “limit” clause to select the first 20 rows. Instead, think of other ways to filter out the bottom rows.~~

Column Order: Customer name, Order number, Order size

Query 2

- Produce an alphabetical list of all nations in the database, along with the total order volume within each nation (supplier nation equal customer nation).
- The total order volume can be defined as the sum of order sizes.
- Write **two** queries, one using a nested query and one without a nested query.

Column order: Country name, Order volume

Query 3

- Produce a list of all nations in the database. In your result, include the name of the seller nation, each nation's total order volume from nations within their own region, and the total order volume from nations in a different region.
- Sort the nations by their total order volume within their own region in descending order.

Column order: Seller country name, total order volume within region, total order volume from different region

Query 4

- Find a list of all suppliers with orders from more than 615 different customers. Order the rows by decreasing customer count.

Column order: Supplier name, customer count

Development Tools

- SQLite3 is the DBMS we're going to use
 - Installed on CS lab machines already
 - Free for you to install on your personal machines
- Download for Windows/Mac:
 - <https://www.sqlite.org/download.html>
 - Can use `apt-get install sqlite3` for Linux
 - Easiest way is probably just to use CS lab machines
- Jupyter notebook template [here](#) if you prefer
 - Need Python 3, Jupyter Notebook, and ipython-sql package to run (installation instructions [here](#)).
 - **Completely optional**, can easily do this assignment without it.

SQLite Command Line

- `sqlite3 TPC-H.db`
 - Loads the database file and starts the SQLite process
- Run a single query and quit
 - `sqlite3 TPC-H.db < query.txt`
- SQLite commands
 - `.tables`
 - Lists all of the loaded tables
 - `.schema`
 - Displays the CREATE statements for the loaded database
 - `.read <filename>`
 - Execute SQL commands in <filename>
 - `.quit`
 - `.help`

Query Syntax

- Most of the syntax we've talked about in class is present in SQLite
 - Some of the more complex SQL statements may differ between DBMSs
 - <https://www.sqlitetutorial.net/> provides a complete listing of all essential SQLite commands

Submission

1. Put each query in its own text file
 - Ex: Query1.txt, Query2.txt, etc.
 - Add your student ID number as a SQL comment in each file
 - Use a single query for each problem
 - i. “with” clauses are allowed when convenient
2. Copy your query results from all queries into a file called `results.txt`
3. Put all query files and `results.txt` into a directory, tar it up, submit on Canvas