Homework 1

Due Tuesday, Oct 15th at 11:59pm

Objectives:

Learn how to use a shared-nothing, relational database management system (DBMS) offered as a cloud service. We will use Amazon Redshift on the Amazon Web Services (AWS) cloud. In the assignment, we will set-up Redshift clusters, ingest data, and execute simple queries.

Assignment tools:

- Amazon Redshift
- Amazon Web Services

Amazon Redshift is a very expensive cloud service. For that reason, the assignment will not use datasets nor clusters that are very large. Additionally, we will use the free trial of the service.

To activate the free trial:

- Go to: https://aws.amazon.com/redshift/
- Click on "Get started with Redshift"
- On the right-hand side, click on "Free Trial"
- Read the instructions

IMPORTANT The Amazon Redshift free trial is limited to the *DC2.Large* node type. Please use ONLY those instance types in this assignment. Since we are using Redshift Spectrum to query data directly from s3 in this assignment use the *us-west-2* region. This restriction of region does not apply if you are doing the alternative assignment using RDS (see below).

ALTERNATIVE If you already tried Redshift before and no longer qualify for the free tier or if you do not wish to create your own AWS account but want to stick with using AWS Educate Classroom, you may complete the assignment using RDS instead of Redshift. Go here for directions on alternative assigment using RDS.

What to turn in:

You will turn in: - SQL for the queries - Runtime for each query - Number of rows returned - First two rows from the result set (or all rows if a query returns fewer than 2 rows) - A brief discussion of the query runtimes that you observed in different settings

Submit everything as a single pdf or Markdown file.

How to submit the assignment:

In your GitLab repository you should see a directory called `Homeworks/hw1`. Put your report in that directory. Remember to commit and push your report to GitLab (`git add && git commit && git push`)!

You should add your report early and keep updating it and pushing it as you do more work. We will collect the final version after the deadline passes. If you need extra time on an assignment, let us know. This is a graduate course, so we are reasonably flexible with deadlines but please do not overuse this flexibility. Use extra time only when you truly need it.

Assignment Details

In this Assignment you will be required to deploy a Redshift cluster, ingest data, and run some queries on this data.

1. Setting up Amazon Redshift (0 points)

- Follow the instructions above to activate your Redshift trial
- Download and install SQLWorkbenchJ
- Create a myRedshift IAM role
- Create a Redshift security group
- Deploy Redshift cluster
- Connect to the Redshift cluster

2. Ingest Data into Amazon Redshift (0 points)

- Ingest data into Redshift

3. Run Queries

Run each query listed below multiple times. Plot the average and either min/max or standard deviation. Use the warm cache timing, which means you discard the first time the query is run. Go to the Query tab on the AWS web console for your redshift cluster to view the runtime of the queries.

1. Write and run queries on the *1GB* dataset and a 2-node cluster (15 points)

- What is the total number of parts offered by each supplier? The query should return the name of the supplier and the total number of parts.
- What is the cost of the most expensive part by any supplier? The query should return only the price of that most expensive part. No need to return the name.
- What is the cost of the most expensive part for each supplier? The query should return the name of the supplier and the cost of the most expensive part but you do not need to return the name of that part.
- What is the total number of customers per nation? The query should return the name of the nation and the number of unique customers.
- What is number of parts shipped between 10 oct, 1996 and 10 nov, 1996 for each supplier? The query should return the name of the supplier and the number of parts.

2. Run queries from (1) on the *10GB* dataset (15 points)

You can remove the 1GB data by executing `DELETE FROM table_name` for each table. Note that you will run more queries on the 1GB dataset below. You may want to do those questions first. After you delete the 1GB dataset, load the 10GB dataset.

The lineitem table is the largest table for the 10 GB dataset. Load this table from parallel file segments instead of the single file. The data for this table is divided into 10 segments. They are named `lineitem.tbl.1`, `lineitem.tbl.2`, ..., `lineitem.tbl.10` in the bucket `s3://uwdb/tpch/uniform/10GB-split/`

3. Run queries from (1) on the 10 GB dataset but this time increase the cluster size from 2 to 4 nodes. To do this, you can either create a new, larger cluster or you can use the cluster resize feature available from the AWS console.

4. A customer is considered a *Gold* customer if they have orders totalling more than $1,000,000.00. Customers with orders totalling between $1,000,000.00 and $500,000.00 are considered *Silver*. Write a SQL query to compute the number of customers in these two categories. Try different methods of

writing the query (only SQL or use a UDF or a View to categorize a user). Discuss your experience with the various methods to carry out such analysis. Use the 1GB data set and the 2-node cluster. (10 points)

5. Query data on s3 vs local data: re-run the queries from (1) on the *10GB* data set on s3. Data is located in `s3://uwdb/tpch/athena/` with a folder for each of the following tables: `customer`, `supplier`, `orders`, `region`, `nation`, `part`, `partsupp` & `lineitem`. The data is in textfile format and the delimiter is the pipe character (|). For example, the s3 location of the data for the lineitem relation is `s3://uwdb/tpch/athena/linitem/` (15 points)

Here are sample dataload times for the 10 GB dataset to a 2 node cluster using the `d2.large` node type: - customers: 33.86s - orders: 2m 9s - lineitems: 7m 28s (single file load) 3m 41s(multi file load) - nation: 2.31s - part: 24.73s - partssup: 1m 23s - region: 2.18s - supplier: 8.27s

Alternative assignment using RDS

In this alternative assignment you will deploy an instance of AWS RDS, ingest data, and run some queries on this data. This alternative is provided for students who do not wish to run a Redshift cluster. Amazon RDS deployment can use credits provided in AWS Educate.

Directions for getting started with RDS are available at Getting started with RDS.

For this assignment you need to setup RDS with a single instance of PostgreSQL (choose `m3.large` instance type), ingest data and run some queries. Alternatively you can try MySQL or Aurora. Information on how to import data in a delimited file in s3 into a RDS instance is available at: Importing data to RDS from S3 with data pipeline.

Report and Submission for Alternative Assignment

The due date and instructions for what to turn in and how to turn in the assignment remain the same for this alternative. Additionally, please specify what type of RDS (MySQL, PostgreSQL, Aurora) database you used. We tested the assignment with PostgreSQL but you are welcome to try one of the other systems.

Remember the data files in s3 use `|` as the delimiting character. The s3 buckets are located in `us-west-2` (Oregon).

Deploy & Run RDS Queries

- Deploy an instance of Amazon RDS. You can choose Postgres, Aurora or MySQL.
- Ingest the 1GB TPC-H dataset. This data-set is in s3 at `s3://uwdb/tpch/uniform/1GB/` with a seperate file for each table (e.g., `customer.tbl`, `supplier.tbl`). You will need to create the same eight tables specified here.
- Complete question 1 from the Redshift queries