Piruz Alemi **Big Data Report:**
Subject: Amazon Web Services (AWS) + Apache Zeppeline (ZEPL) Data Science Analytic Platform: ETL, SQL, RDS
Date: April 3rd, 2020

Before I Began

1. Created a new repository for this project called `big-data-challenge`.
2. Cloned the new repository to my computer.
3. Inside my local git repository, created a directory for the level of challenge Challenge I chose. I Used folder names corresponding to the challenges: **level-1** or **level-2**.
4. Added my converted ZEPL notebook to this folder. This became the main script to run for my analysis. I also added any SQL queries. I used a `.sql` file and added it to my repo.
5. Pushed the above changes to GitHub or GitLab.

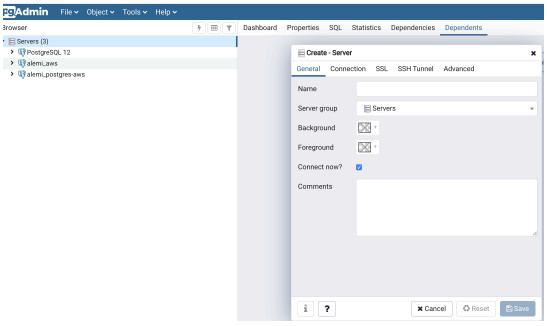
Note:

I used ETL processes in Big Data using PySpark and using AWS's Relational Databases.

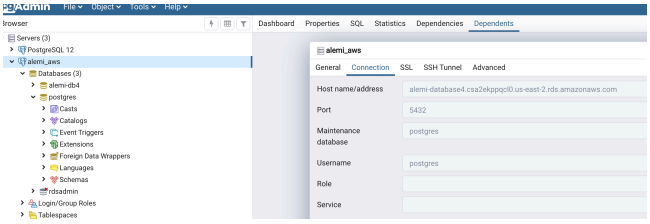
I linked the AWS RDS to PG_Admin:

1. Created in AWS RDS, an end-point for the server pointing to the alemi-db4

2. On my local PG_Admin, created a new server – alemi-db4,



3. Copied and pasted the end-point from AWS – RDS to my newly created server on my local Pg-Admin



4. This connected my PG_Admin to AWS_RDS on the amazon shared public platform

5. In Google_Drive Collaborate Python Program, under SPARKS, established the cpnnection and wrote a new table: Alemi_ETL_Info:

▾ Write DataFrame to RDS

```
# Configuration for RDS instance
mode="append"
jdbc_url = "jdbc:postgresql://alemi-database4.csa2ekppqcl0.us-east-2.rds.amazonaws.com:5432/alemi-db4"
config = {"user":"postgres",
          "password": "*******",
          "driver":"org.postgresql.Driver"}
```

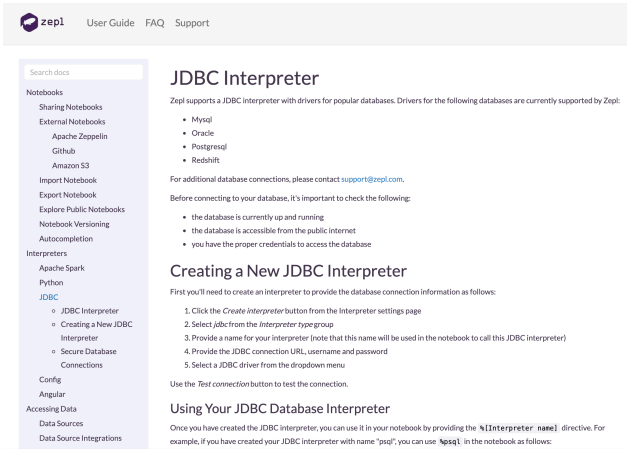
```
[ ] # Write DataFrame to table
    employee_personal_info.write.jdbc(url=jdbc_url, table='Alemi_ETL_info', mode=mode, properties=config)
```

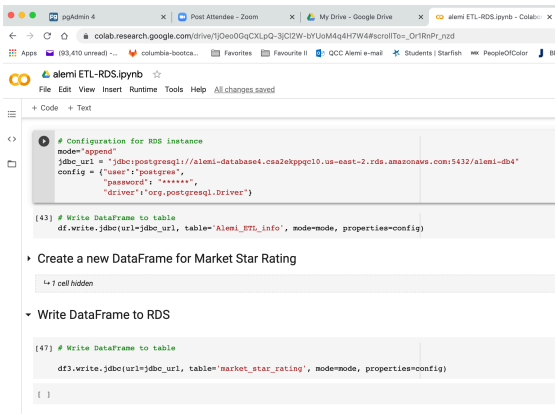
6.

Background

In this assignment you will put my ETL skills to the test. Many of Amazon's shoppers depend on product reviews to make a purchase. Amazon makes these datasets publicly available. However, they are quite large and can exceed the capacity of local machines to handle. One dataset alone contains over 1.5 million rows; with over 40 datasets, this can be quite taxing on the average local computer.

1. My first goal for this assignment was to perform the ETL process completely in the cloud and upload a DataFrame to an RDS instance.





2. My second goal was to use PySpark or SQL to perform a statistical analysis of selected data.

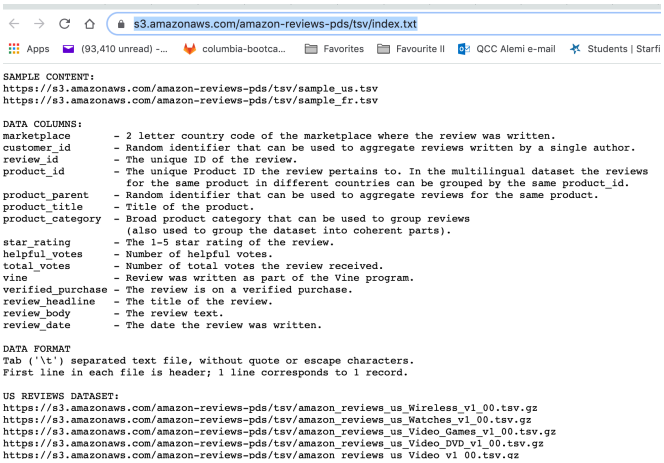
There were two levels to this homework assignment. I completed both levels.

1. Create DataFrames to match production-ready tables from two big Amazon customer review datasets.
2. Analyzed whether reviews from Amazon's Vine program are trustworthy.

Steps

Level 1

- Used the furnished schemata to create tables in my RDS database.
- Created two separate ZEPL notebooks and **extracted** any two datasets from the list at [review dataset], one into each notebook.



Note: It is possible to ETL both data sources in a single notebook, but due to the large data sizes, it was easier to work with these S3 data sources in two separate ZEPL notebooks.

Made sure to handle the header correctly. I noted that the column headers are included in the table rows, which made loading into PG_Admin straightforward. I just needed to create the 1. Server, establish the connection based on the end-point created in AWS

– RDS and 2. Create the DB. 3. Loading the table, with TSV file was a straightforward write from data frame command as in:
`df.write.jdbc(url=jdbc_url, table='Alemi_ETL_info', mode=mode, properties=config)`

- For each notebook (one dataset per notebook), complete the following:

    - **Counted** the number of records (rows) in the dataset.
    - **Transformed** the dataset to fit the tables in the schema file.
        - Made sure the DataFrames match in data type and in column name.
    - **Loaded** the DataFrames that correspond to tables into an RDS instance.
      Note: This process took up to 10 minutes for each! I made sure that everything was correct before uploading.

Level 2 (Optional)

In Amazon's Vine program, reviewers received free products in exchange for reviews.

Amazon has several policies to reduce the bias of its Vine reviews: https://www.amazon.com/gp/vine/help?ie=UTF8.

But are Vine reviews truly trustworthy? My task was to investigate whether Vine reviews are free of bias.

I Used both PySpark or—for an extra challenge—SQL to analyze the data.

- Before I used SQL, first I used Spark on ZEPL to extract and transform the data and load it into a SQL table on my RDS account. Performed my analysis with SQL queries on RDS.
- While there were no hard requirements for the analysis, I considered steps to reduce noisy data, e.g., filtering for reviews that meet a certain number of helpful votes, total votes, or both.
- Submitted a summary of my findings and analysis.

Resources I used were:

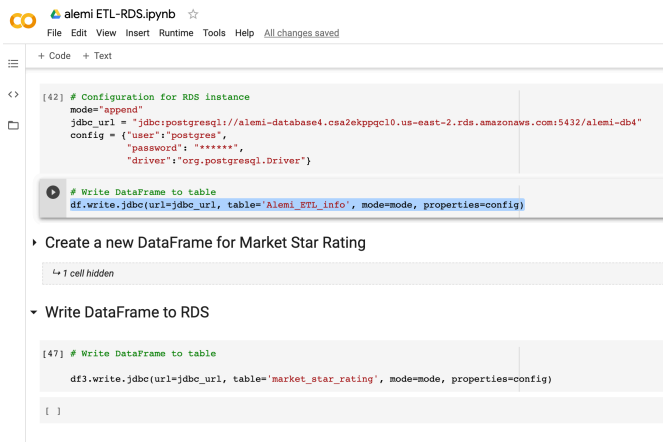
customer review datasets

Hints and Considerations

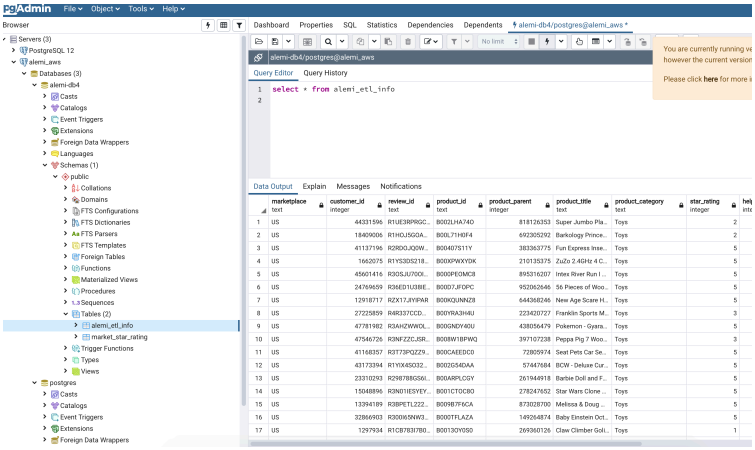
- Consulted the troubleshooting guide for handling issues with ZEPL.
- Made sure that every cell begins with `%pyspark` in ZEPL. This specifies the interpreter, which I had to have one for each cell.
- If you import a Jupyter notebook in ZEPL, made sure to delete `%python`, which is automatically added to each cell.

Submission

- Copied my ZEPL notebooks into Jupyter Notebooks and upload those to GitHub.
- Copied my SQL queries into `.sql` files and uploaded to GitHub.
- **Important:** I Did not upload notebooks that contain my RDS password and endpoint. I made sure to delete them before making your notebook public!



The loaded data in PG_Admin is noted below:



Thank you, all my supporters, for an outstanding teamwork!

Respectfully,

Piruz Alemi
April 5th, 2020