Homework Assignment #5

(Spark)

You have been working all quarter long on different database systems and experiencing their varying environments and approaches to handling and querying data to find useful insights. As the quarter progresses, ZotMusic's business has taken off with a huge growth of their user base. Given the explosion in the data size at ZotMusic, you have decided to explore another interesting framework, one that enables you to execute powerful parallel query tasks to expedite big data processing. You have been hearing about Apache Spark for a while now and feel that this is the perfect time to get some hands-on experience with it. You decided to talk to a colleague who has some experience using Spark to get help in loading the full ZotMusic data, and your colleague delivered - so your data now awaits on Canvas!

Your colleague has also advised you to first use the **free** Databricks Community Edition to explore Spark, and then process the full dataset using a **paid** Databricks premium version (hosted on **AWS)** so that you can catch up with what other data engineers at ZotMusic are already doing daily. Don't worry; this will cost only a few bucks.

Get Ready… (and set) for Part I of this assignment

To start, first follow the **first part** of the (separately provided) setup instructions **carefully** to create a Community Edition account with Databricks to use for your adventures here. For this part of the assignment, you will need to download the **sample dataset (zot-music-dataset-assignment5-sample)** to finish each question, so that you can focus on coming up with and refining your queries. But keep in mind that you will **NOT** submit your assignment based on results on the sample dataset.

Go...!

It's time to explore the new system! Begin your Big Data analytics adventure by opening the provided Spark notebook template file and then **describe**-ing the schema of the preloaded data frames along with their associated created views.

1. Start by running some schema description statements in order to better understand the preloaded relations.
   A. First explore the schema for the *users* relation. Write a **Python** snippet using Spark **dataframes** to show the schema of *users***.**
   B. What is the data type of the *address* field?
   C. Now let's see how to view a schema using SQL in Spark. Write a **SQL statement** to view the schema of *records*.
   D. What is the data type of the *songs* field?

2. After scratching the surface on how to use Spark, you are eager to start writing queries and analyzing the data! Answer questions [2.A - 2.G] in **TWO** ways (both) - first by providing dataframe code fragments and then by providing equivalent SQL statements - to answer each of the desired queries.
   A. To start your Spark journey, you would like to view all the details of the user on the platform whose user_id is **'user_W2xxLgbyRoqp'**.
   B. Running aggregation queries on top of data is a powerful way to analyze data. Determine and print the total number of Pop records (records whose genre is "Pop") placed by each artist and sort the results by the number of Pop records. Include their user_id and the number of Pop records that they have released in your result, and limit the number of result rows to 5.
   C. Aggregated reports are easier to read and present as compared to raw collected data. Your boss is interested in looking at last year’s streaming data on ZotMusic. You have been asked to find the top 3 most streamed songs of last November. That is, for each song streamed by listeners, determine and print the number of sessions initiated by listeners in November 2023. Sort the results in descending order of the session count. **Note**: you should print both the song's record_id, song's track number, and the session count, and “top 3” implies that they should be listed in descending order.
   D. You have heard that Spark can deal with array data fairly well and you want to explore this feature. Your boss wants to see which genre is most interesting to users. Find the genre that the largest number of users have indicated interest in (i.e., included in a user's `genre` attribute), and print the genre's name and the number of interested users.
   E. Join is one of the most interesting operations for data analysis since it helps in connecting the dots when it comes to data, so you want to understand how joins work in Spark. Find and print the review_id of reviews posted between 2024-08-03 (inclusive) and 2024-09-08 (exclusive), the **emails** of the users who posted the reviews, along with the posted_at time for each review. Sort the results in increasing order of posted time, and limit the results to only the first 5 rows.
   F. Find and print the emails of the top 3 listeners who received the most upvotes for their posted reviews, i.e., create a list of listeners with the total number of upvotes that a listener has received for all this listener's reviews. This time you should only return their emails and upvote counts. Sort your list in descending order based on the number of upvotes received, and list only the first 3 results.
   G. You now have some mastery of the Spark engine and you’re feeling like you’re ready to tackle a more complex problem. Find the **top 5** artists that have the highest ***average certified*** *record ratings* (*ACRRs*) on the platform. Here:
      i. The definition of a *certified record rating* for a record is that the review where the rating comes from should have received **at least 5 upvotes**, **AND** that the **listener who posted the review should have played at least 275 sessions** on ZotMusic.

ii. The *average certified record rating (ACRR)* for an artist is simply the **arithmetic average** of **all these certified record ratings** in reviews posted for **any record** belonging to this artist.

For each such artist, return their user_id, their ACRR, the total number of certified record ratings their records have received, and their first and last name. Print the 5 results in descending order of their ACRR.

Go….. again! (Part II of this assignment)

3. Congratulations! You have successfully finished the first part of this assignment and are halfway to becoming a Spark expert. So far you have been answering each question using a **sample** dataset so that you could focus on refining your queries instead of waiting for their executions to finish. But that has been preventing you from learning the true power of Spark. Now for the next part of this assignment, you will try your answers on a real, large dataset and submit your solutions **only** on this full dataset.

   First, follow these steps:

   1) Download the **full dataset (zot-music-dataset-assignment5-full)** from Canvas and upload it to the Databricks Community Version.
   2) Utilize the **template notebook** to load the complete dataset as the data source of your solution.
   3) Run your solution notebook on the complete dataset once **from start to finish** to produce results for each question on the large dataset. **Observe** how long it takes for each query to finish execution in the free, single-node Databricks Community Version.

Then, download your solution notebook as an .ipynb file. Carefully follow the second part of the setup instructions to create a **paid Databricks account (hosted on AWS)**, and follow these steps:

1) Upload the **complete** dataset to your premium workspace.
2) Upload your .ipynb notebook downloaded from the Community Version to the premium workspace.
3) Create a compute cluster with **at least** 2 nodes in your premium workspace, and attach the compute cluster to your uploaded notebook.
4) Run the **whole notebook** again, from start to finish, on the complete dataset using this compute cluster. Observe again the time it takes for each query to finish.

Now for the actual question that you need to answer: **What difference do you observe between using the free, single-node version of Spark and the paid, multi-node version of Spark?**

What to Turn In

When you have finished your assignment, you should use Gradescope to turn in a PDF of your notebook file that lists all queries you have run ***from start to finish, on the full dataset***. Include your answers to the questions that aren’t queries as comments or Markdown cells. Please follow the steps below in order to generate the file for submission:

1. Download the template .ipynb file from the Canvas. (You should have gotten this taken care of in the setup document, actually.)
2. After answering the queries both ways (i.e., in Dataframe and SQL form), make sure that you can run ***the entire file*** from start to finish and that you get no execution errors at all when you run it that way.
3. Click “File” -> “Export” -> “HTML”. Then, you can open the downloaded HTML file and print it as a PDF file as your final result. Once done, double check that all the code ***and the results*** are all clearly visible in your PDF file.
4. Submit your PDF file to Gradescope as usual, and remember to kindly assign relevant pages for each question to make grading easier!