Big Data Engineer Code Test

Overview

Create a data ingestion pipeline with a sample dataset using **Scala Spark** or **PySpark**. The code can be organised however you wish, but there should be a clear logical division between the staging and transformation stages of the pipeline.

Data

The download link for the smaller MovieLens dataset and accompanying documentation can be found here.

Requirements

Staging

Your staging layer should do the following:

1. Load the ratings data into either a DataFrame or a Dataset and write the results to a delta lake table:

- Using *userId* and *movieId* as a primary key, update a row if it exists, else the row should be inserted.
- Find an appropriate partitioning strategy that will scale when more data is added.

2. Load the movies and tags data separately into either a DataFrame or a Dataset and write the results of each to separate delta lake tables.

3. Cast columns to the correct data types as appropriate.

The jobs should be able to process the original MovieLens dataset as well as any newer CSVs that might arrive as part of a future load.

Transformation

Your transformation layer should do the following:

4. Implement a method for splitting the movie genres so that there is a single genre per row. For example:

> 'Comedy|Romance'
>
> becomes:
>
> 'Comedy
> Romance'

Save the results however you wish.

5. Implement a method that finds the top 10 films by average rating. Each of the top 10 films should have at least 5 ratings to qualify as a top 10 movie. Order by the highest rated film first and write the results out to a single CSV file.

Your solution

The problem itself is quite simple but please consider this as the first iteration of a production level software. We are not interested in you showing the best deliverable possible but rather to show us how, given a limited amount of time, you will prioritise your effort to guarantee a quality deliverable in the given time.

All your unit tests **must** pass – a submission with failing tests will not pass the review stage.

You can present your solution by hosting it in a private GitHub repository (or a similar source control solution) and sending a link to the solution, or by zipping the up code and sending it directly. Please include instructions on how to build the solution if necessary.

NOTE: Before zipping up files, be sure to remove files/executables that might get blocked by email filters.