Data Lake

Project Overview

A music streaming startup, Sparkify, has grown their user base and song database even more and want to move their data warehouse to a data lake. Their data resides in S3, in a directory of JSON logs on user activity on the app, as well as a directory with JSON metadata on the songs in their app.

This project mainly is to build an ETL pipeline for a data lake hosted on S3. To complete the project, you will need to load data from S3, process the data into analytics tables using Spark, and load them back into S3. You'll deploy this Spark process on a cluster using AWS.

File Introduction

- dl.cfg - AWS Account information
- etl.py - Major ETL File that do the ETL job

Major Chanllenges and Solutions

- Filepath of Json Files - need to add a after s3.
- Pyspark has no embedded method to extract day of week. We need to create function ourselves. Additionally, on offcial developer guidebook, it has the function of add one day to day of week, which it's interesting.
- Add Sequential numbers in Pyspark.
    1. First method is to use monotonically_increasing_id() function. But the disadvantage of this method is that it doesn't necessarily start from zero and have the same intevals.
    2. Second method is to add the each row's row number. Good thing is it looks like what we need. But on the other side, we still need to do calculation if we want it start from 0.

- Create datetime and timestamp from unix time.
  1. UDF Function using Python datetime module as requested by this project;
    2. There is another easy way without using UDF.
       df.withColumn('epoch', f.date_format((df.ts/1000).cast(dataType=t.TimestampType()), "yyyy-MM-dd"))

Pitfalls I Run into

- Start to run Pyspark code before Spark session starts. Except import function, all other codes should be done after session is created