ETL and Data Pipelines

Assignment overview

This assignment is in two parts, ETL using Python and Data Pipelines using Apache AirFlow.

Part 1: ETL using Python

The Data Warehouse gets information from several sources including the transactional (OLTP) database. Transactional data from the OLTP database (in this case MySQL) needs to be propagated to the Warehouse on frequent basis. This data movement can be updated using ETL processes.

In this first part of the assignment, you will setup an ETL process using Python to extract new transactional data for each day from the MySQL database, transform it and then load it into the data warehouse in DB2.

You will begin by preparing the lab environment by starting the **MySQL** server. You will then create a sales database in MySQL and import the sales.sql into the sales database. Next, you will verify your access to the cloud instance of the **IBM DB2** server. You will then be uploading the sales.csv to a table in DB2. To do so you will download the python files that have the template code to connect to MySQL and DB2 databases.

The final task requires you to automate the extraction of daily incremental data and load yesterday's data into the data warehouse. You will download the python script from this link and use it as a template to write a python script that automatically loads yesterday's data from the production database into the data warehouse. After performing each task, take a screenshot of the command you used and its output, and name the screenshot.

Part 2: Data Pipelines using Apache AirFlow

Our data platform includes a Big Data repository that is used for analytics using Machine Learning with Apache Spark. This Big Data repository gets data from several sources including the Data Warehouse and the Web Server log. As data from the web server log is logged, it needs to be added to the Big Data system on a frequent basis - making it an ideal process to automate using a data pipeline.

In this second part of the assignment, you will create and run a DAG using Apache Airflow to extract daily data from the web server log, process it, and store it in a format to prepare it for loading into the Big Data platform.

To complete this part of the assignment, you will perform a couple of exercises, but before proceeding with the assignment, you will prepare the lab environment by starting the **Apache Airflow** and then downloading the dataset from the source (link provided) to the mentioned destination. In the first exercise, you will perform a series of tasks to create a DAG that runs daily. You will create a task that extracts the IP address field from the webserver log file and then saves it into a text file. The next task creation requires you to This task should filter out all the occurrences of ipaddress "198.46.149.143" from text file and save the output to a new text file. In the final task creation, you will load the data by archiving the transformed text file into a TAR file. Before moving on to the next exercise, you will define the task pipeline as per the given details.

In the second exercise, you will get the DAG operational by saving the defined DAG into a PY file. Further, you will submit, unpause and then monitor the DAG runs for the Airflow console. After performing each task, take a screenshot of the command you used and its output, and name the screenshot.