Flight Performance Analysis using Graphframes and Databricks

What is the Agenda of the project?
This project uses GraphFrames in the Databricks notebook to quickly and easily analyze flight performance data organized into graph structures. The graph structure makes it easy to ask many questions that are not as intuitive as the table structure, like finding structural motifs, finding the shortest route between cities, and ranking airports with PageRank. In this, we will be utilizing departure delay data to perform analysis and answer the following questions:

- Determine the number of airports and trips
- Determining the longest delay in this dataset
- Determining the number of delayed vs. on-time / early flights
- Which flights departing SFO are most likely to have significant delays
- Which destinations tend to have delays
- Which destinations tend to have significant delays departing from SEA
- Relationships through Motif Finding
- Airport Ranking using PageRank
- Most popular flights
- Top Transfer Cities

Dataset Description:
The dataset consists of 1048576 data points, including the following parameters:

- Date
- Delay
- Distance
- Origin
- Destination

Tech stack:
➔Language: Python
➔Package: Pyspark
➔Services: Databricks, Graphframes, Spark

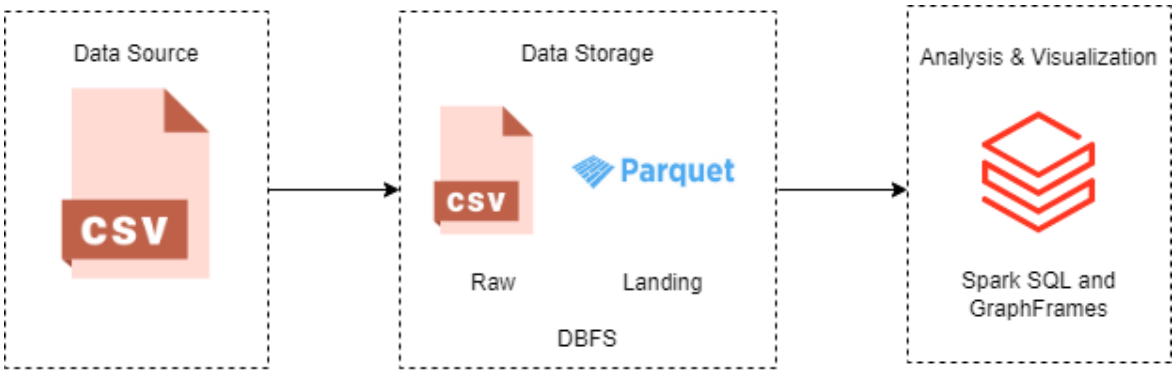
Databricks:
Databricks is a cloud platform for large-scale data engineering and collaborative data science. Databricks combines best-in-class data warehousing and data lakes into a laser house architecture. Databricks is developing a web-based Spark framework that provides automated cluster management and IPython-style notebooks. In addition to building the Databricks platform, the company co-hosted the Spark Massive Open Online Courses and a conference for the Spark community. Databricks also provides a platform for other workloads including machine learning, data warehousing, streaming analytics, and business intelligence.

Graphframes:
GraphFrames is an Apache Spark module that creates DataFrame-based Graphs. The DataFrame API, paired with a new API for motif searching, allows users to design more expressive searches. DataFrame speed changes within the Spark SQL engine also help the user. It has Scala, Java, and Python high-level APIs. It intends to give GraphX capabilities as well as additional capability using Spark DataFrames. Among the new features are motif finding, DataFrame-based serialization, and very expressive graph searches.

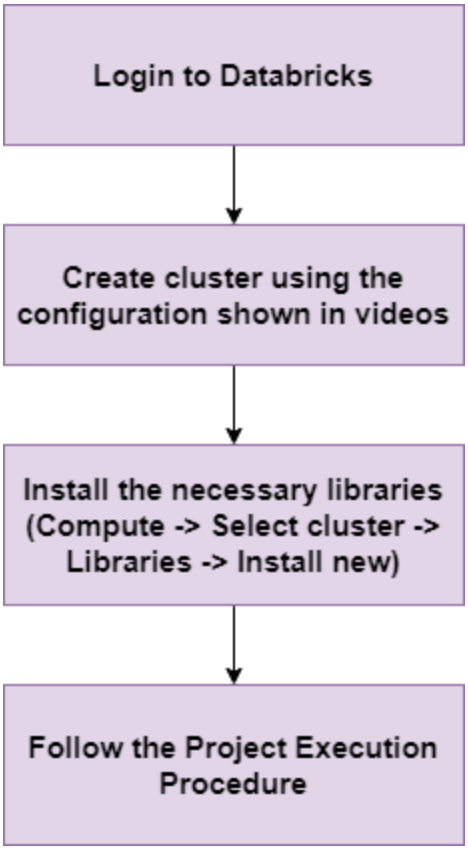
Key Takeaways:

- Understanding the project and how to use Databricks
- Creating cluster in Databricks
- Installing libraries on a cluster in Databricks
- Understanding the Spark SQL
- Understanding the Spark GraphFrames
- Importing and storing data in Databricks
- Understanding the use of Spark UI, Event logs, Driver logs and Metrics
- Create notebook on a cluster in Databricks
- Understanding the cluster architecture, workers, drivers and jobs in Databricks
- Understanding the concept of Graph analysis
- Directed vs Undirected edges
- Creating Graphframe in a notebook
- Understanding the concept of TriangleCount and PageRank algorithms
- Understanding the use of Motif finding
- Understanding the concept of Breadth-first search (BDF)

Project architecture:



Project workflow:



Folder structure:
Raw data:
/FileStore/tables/raw/airport_codes_na.txt
/FileStore/tables/raw/departuredelays.csv

```
|
-- FileStore
    |
    -- tables
        |
        -- raw
          |
            -- airport_codes_na.txt
            -- departuredelays.csv
```