Desafío - Spark II

- Para realizar este desafío debes haber estudiado previamente todo el material disponibilizado correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el `.zip` en el LMS.
- Desarrollo desafío:
    - El desafío se debe desarrollar de manera Individual/Grupal.
    - Debes tener precaución con las rutas y el sistema operativo que utilices.

Sobre los datos del ejercicio

- El Buró de Estadísticas de Transportes del Departamento de Transporte de los Estados Unidos de América realiza un seguimiento en tiempo real del desempeño de los vuelos domésticos realizados por operadores de gran escala.

- Para este ejercicio se trabajará con tres tablas en formato columnar parquet que se detallan a continuación:
    - `flights.parquet`: Tabla correspondiente a todos los vuelos realizados en el 2015.
    - `airports.parquet`: Tabla correspondiente a todos los aeropuertos dentro de los Estados Unidos de América.
    - `airlines.parquet`: Tabla correspondiente a todos los operadores de gran escala en el 2015.

Ejercicio 1: Preliminares

- Genere una instancia de trabajo en AWS EMR con los componentes necesarios de `Spark` y habilite un puerto dinámico para utilizar un notebook desde JupyterHub.

- Genere un objeto con `SparkSession` y asegúrese de habilitar el soporte con Hive.

- Utilizando su objeto creado con `SparkSession`, realice el import de los objetos `parquet` que se encuentran en la siguiente dirección del bucket del curso `s3://bigdata-desafio/challenges/u4act2/`.

- Infiera el `schema` de cada objeto creado.

Ejercicio 2: Implementación de Queries

En su calidad de Científico de Datos, su jefe le genera una serie de consultas que deberá implementar utilizando sus conocimientos en SparkSQL y sus objetos DataFrame. La única limitante es que estará trabajando en un cluster habilitado sólo con el kernel PySpark3, por lo que no podrá utilizar librerías como pandas, numpy y matplotlib. Cabe destacar que usted no tendrá permisos de superusuario para instalar librerías.

- **Query 1:** Cantidad de vuelos por mes. Reporte los meses con una mayor cantidad de vuelos.

- **Query 2:** Cantidad de vuelos por día y mes. Reporte los días con una mayor cantidad de vuelos.

- **Query 3:** Cantidad de aeropuertos por Estado. Reporte los estados con una mayor cantidad de aeropuertos.

- **Query 4:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports`, identifique los aeropuertos con una mayor cantidad de vuelos.

  Tips:
  - *Para identificar los aeropuertos, utilice la columna `ORIGIN_AIRPORT` de la tabla `flights`.*
  - *Haga un join con la tabla `airports`, utilizando el `IATA_CODE` como registro identificador.*

- **Query 5:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports`, identifique los estados con una mayor cantidad de vuelos.

  Tips:
  - *Se sugiere implementar una query de SQL para resolver este problema.*
  - *No se olvide de registrar las tablas temporales.*

- **Query 6:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports`, identifique el promedio de retraso en partida (con la columna `DEPARTURE_DELAY`) y llegada (con la columna `ARRIVAL_DELAY`) para cada aeropuerto de origen (con la columna `ORIGIN_AIRPORT`). Reporte los cinco aeropuertos con un mayor retraso promedio de partida.

- **Query 7:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports`, identifique las principales razones de cancelación de vuelos.

- **Query 8:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports` y sólo considerando los cinco aeropuertos con un mayor retraso promedio de partida, identifique las principales causas de cancelación de vuelos.

  Tips:
  - *Genere una lista con las siglas de cada aeropuerto con mayor retraso.*
  - *Para filtrar registros por elementos dentro de una lista, pueden hacer uso de las funciones `where` e `isin`.*

- **Query 9:** Excluyendo los aeropuertos que no aparezcan en la tabla `airports`, identifique el tiempo promedio de retraso en partida y llegada para cada aerolínea.

  Tips:
  - *Se sugiere implementar una query de SQL para resolver este problema.*
  - *No se olvide de registrar las tablas temporales.*

Ejercicio 3: Descarga de resultados y eliminación de la instancia

- Saque un screenshot del término de la instancia AWS EMR.

- Descargue el notebook desde JupyterHub.

- Comprima los archivos en un `.zip`.