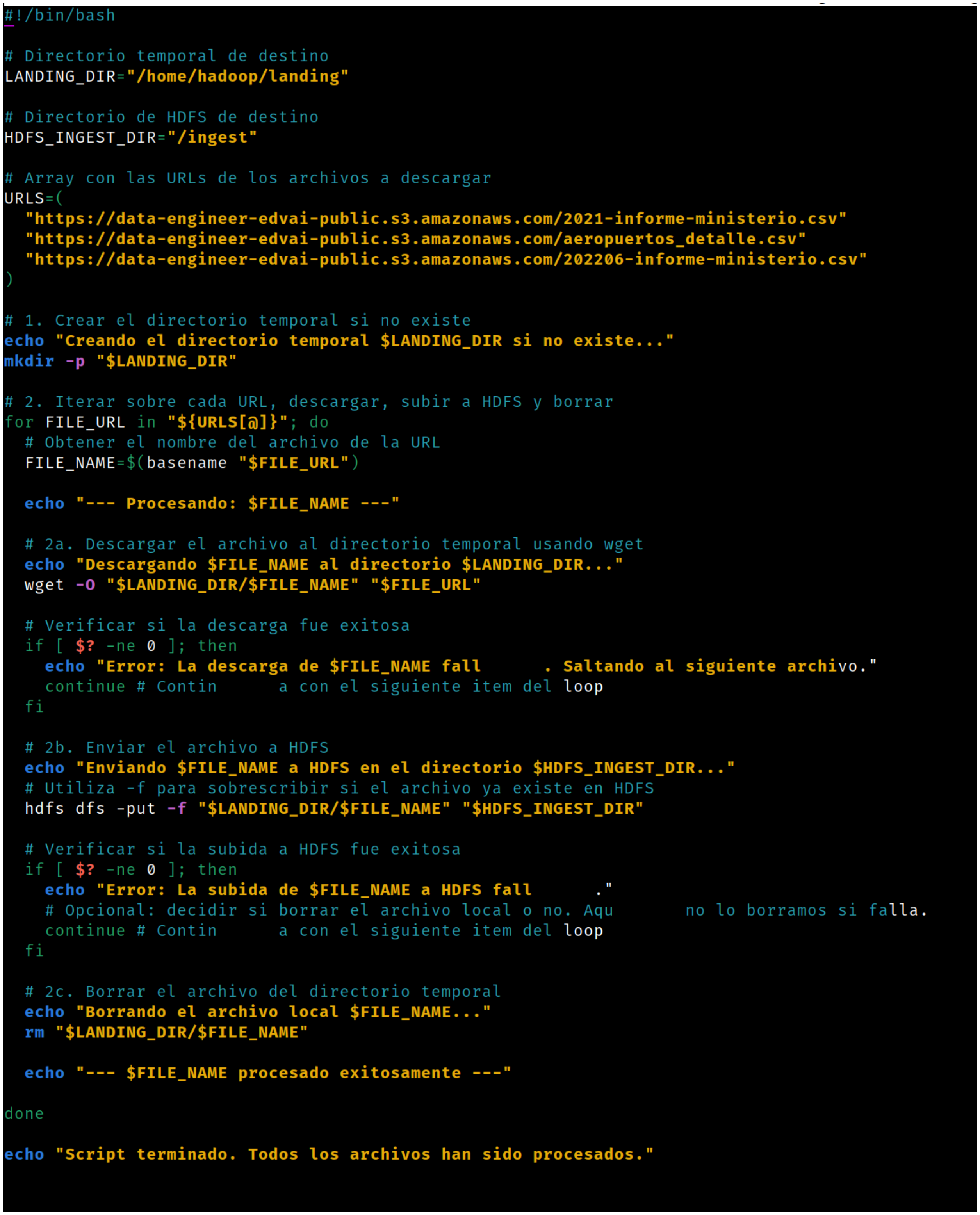
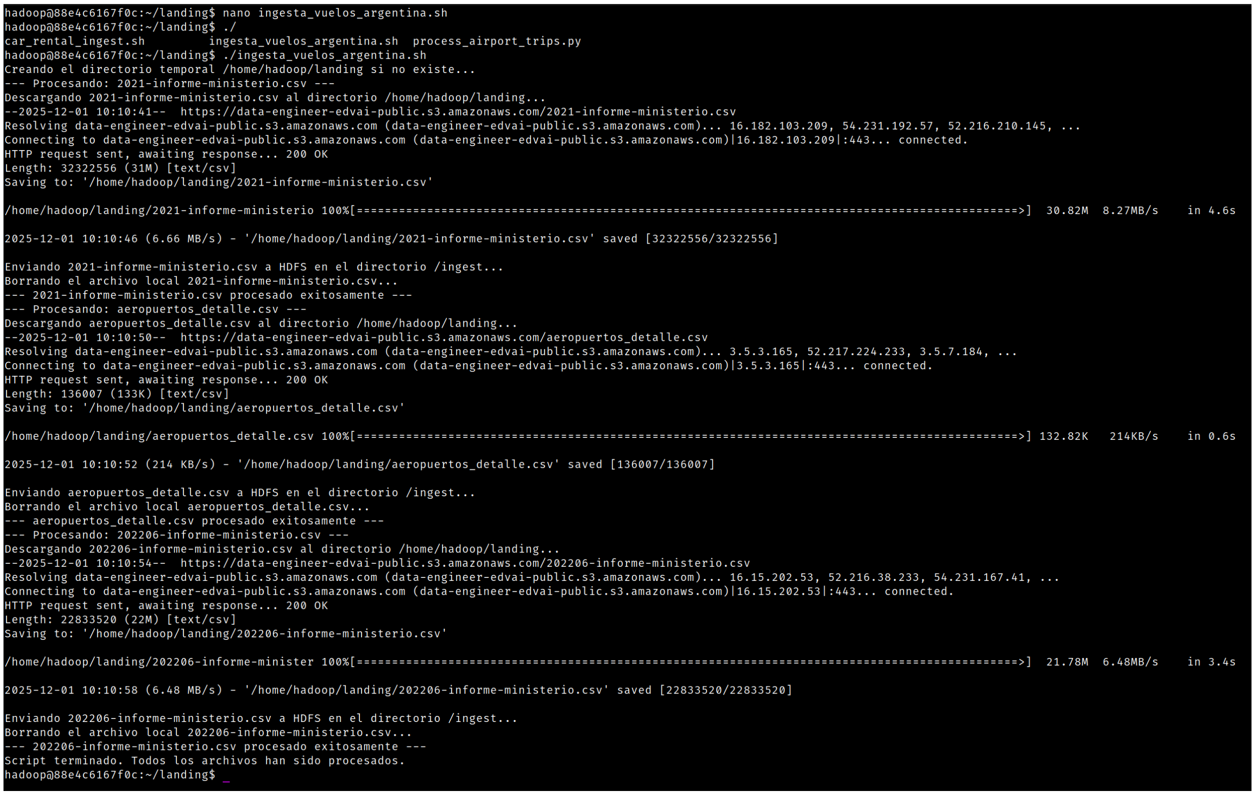
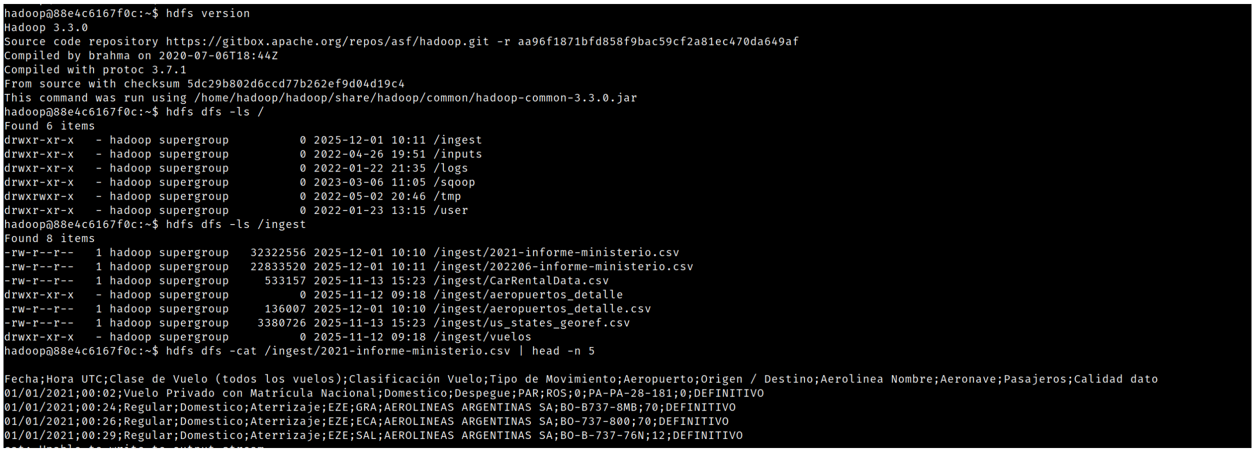
ETL Vuelos Ejercicio Final



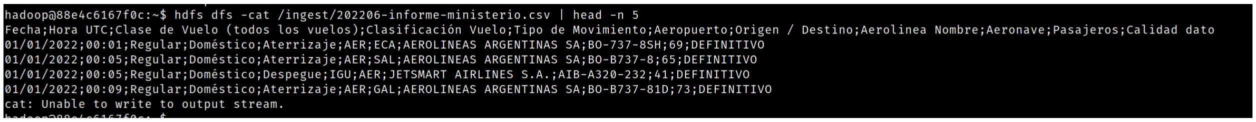
```
#!/bin/bash

# Directorio temporal de destino
LANDING_DIR="/home/hadoop/landing"

# Directorio de HDFS de destino
HDFS_INGEST_DIR="/ingest"

# Array con las URLs de los archivos a descargar
URLS=(
  "https://data-engineer-edvai-public.s3.amazonaws.com/2021-informe-ministerio.csv"
  "https://data-engineer-edvai-public.s3.amazonaws.com/aeropuertos_detalle.csv"
  "https://data-engineer-edvai-public.s3.amazonaws.com/202206-informe-ministerio.csv"
)

# 1. Crear el directorio temporal si no existe
echo "Creando el directorio temporal $LANDING_DIR si no existe..."
mkdir -p "$LANDING_DIR"

# 2. Iterar sobre cada URL, descargar, subir a HDFS y borrar
for FILE_URL in "${URLS[@]}"; do
  # Obtener el nombre del archivo de la URL
  FILE_NAME=$(basename "$FILE_URL")

  echo "--- Procesando: $FILE_NAME ---"

  # 2a. Descargar el archivo al directorio temporal usando wget
  echo "Descargando $FILE_NAME al directorio $LANDING_DIR..."
  wget -O "$LANDING_DIR/$FILE_NAME" "$FILE_URL"

  # Verificar si la descarga fue exitosa
  if [ $? -ne 0 ]; then
    echo "Error: La descarga de $FILE_NAME fall      . Saltando al siguiente archivo."
    continue # Contin      a con el siguiente item del loop
  fi

  # 2b. Enviar el archivo a HDFS
  echo "Enviando $FILE_NAME a HDFS en el directorio $HDFS_INGEST_DIR..."
  # Utiliza -f para sobrescribir si el archivo ya existe en HDFS
  hdfs dfs -put -f "$LANDING_DIR/$FILE_NAME" "$HDFS_INGEST_DIR"

  # Verificar si la subida a HDFS fue exitosa
  if [ $? -ne 0 ]; then
    echo "Error: La subida de $FILE_NAME a HDFS fall      ."
    # Opcional: decidir si borrar el archivo local o no. Aqu       no lo borramos si falla.
    continue # Contin      a con el siguiente item del loop
  fi

  # 2c. Borrar el archivo del directorio temporal
  echo "Borrando el archivo local $FILE_NAME..."
  rm "$LANDING_DIR/$FILE_NAME"

  echo "--- $FILE_NAME procesado exitosamente ---"

done

echo "Script terminado. Todos los archivos han sido procesados."
```



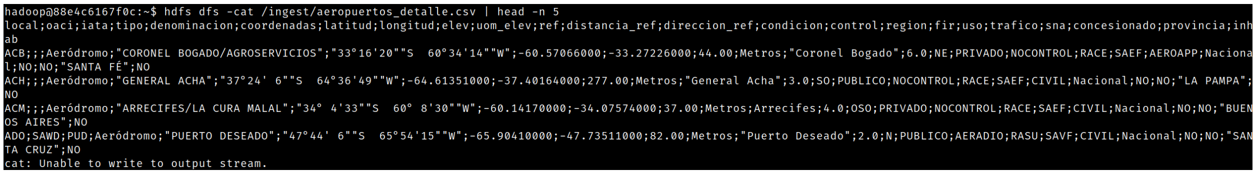
```
hadoop@88e4c6167f0c:~/landing$ nano ingesta_vuelos_argentina.sh
hadoop@88e4c6167f0c:~/landing$ ./
car_rental_ingest.sh         ingesta_vuelos_argentina.sh  process_airport_trips.py
hadoop@88e4c6167f0c:~/landing$ ./ingesta_vuelos_argentina.sh
Creando el directorio temporal /home/hadoop/landing si no existe...
--- Procesando: 2021-informe-ministerio.csv ---
Descargando 2021-informe-ministerio.csv al directorio /home/hadoop/landing...
--2025-12-01 10:10:41--  https://data-engineer-edvai-public.s3.amazonaws.com/2021-informe-ministerio.csv
Resolving data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)... 16.182.103.209, 54.231.192.57, 52.216.210.145, ...
Connecting to data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)|16.182.103.209|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 32322556 (31M) [text/csv]
Saving to: '/home/hadoop/landing/2021-informe-ministerio.csv'

/home/hadoop/landing/2021-informe-ministerio 100%[=============================================================================================>]  30.82M  8.27MB/s    in 4.6s

2025-12-01 10:10:46 (6.66 MB/s) - '/home/hadoop/landing/2021-informe-ministerio.csv' saved [32322556/32322556]

Enviando 2021-informe-ministerio.csv a HDFS en el directorio /ingest...
Borrando el archivo local 2021-informe-ministerio.csv...
--- 2021-informe-ministerio.csv procesado exitosamente ---
--- Procesando: aeropuertos_detalle.csv ---
Descargando aeropuertos_detalle.csv al directorio /home/hadoop/landing...
--2025-12-01 10:10:50--  https://data-engineer-edvai-public.s3.amazonaws.com/aeropuertos_detalle.csv
Resolving data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)... 3.5.3.165, 52.217.224.233, 3.5.7.184, ...
Connecting to data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)|3.5.3.165|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 136007 (133K) [text/csv]
Saving to: '/home/hadoop/landing/aeropuertos_detalle.csv'

/home/hadoop/landing/aeropuertos_detalle.csv 100%[=============================================================================================>] 132.82K   214KB/s    in 0.6s

2025-12-01 10:10:52 (214 KB/s) - '/home/hadoop/landing/aeropuertos_detalle.csv' saved [136007/136007]

Enviando aeropuertos_detalle.csv a HDFS en el directorio /ingest...
Borrando el archivo local aeropuertos_detalle.csv...
--- aeropuertos_detalle.csv procesado exitosamente ---
--- Procesando: 202206-informe-ministerio.csv ---
Descargando 202206-informe-ministerio.csv al directorio /home/hadoop/landing...
--2025-12-01 10:10:54--  https://data-engineer-edvai-public.s3.amazonaws.com/202206-informe-ministerio.csv
Resolving data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)... 16.15.202.53, 52.216.38.233, 54.231.167.41, ...
Connecting to data-engineer-edvai-public.s3.amazonaws.com (data-engineer-edvai-public.s3.amazonaws.com)|16.15.202.53|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 22833520 (22M) [text/csv]
Saving to: '/home/hadoop/landing/202206-informe-ministerio.csv'

/home/hadoop/landing/202206-informe-minister 100%[=============================================================================================>]  21.78M  6.48MB/s    in 3.4s

2025-12-01 10:10:58 (6.48 MB/s) - '/home/hadoop/landing/202206-informe-ministerio.csv' saved [22833520/22833520]

Enviando 202206-informe-ministerio.csv a HDFS en el directorio /ingest...
Borrando el archivo local 202206-informe-ministerio.csv...
--- 202206-informe-ministerio.csv procesado exitosamente ---
Script terminado. Todos los archivos han sido procesados.
hadoop@88e4c6167f0c:~/landing$ _
```



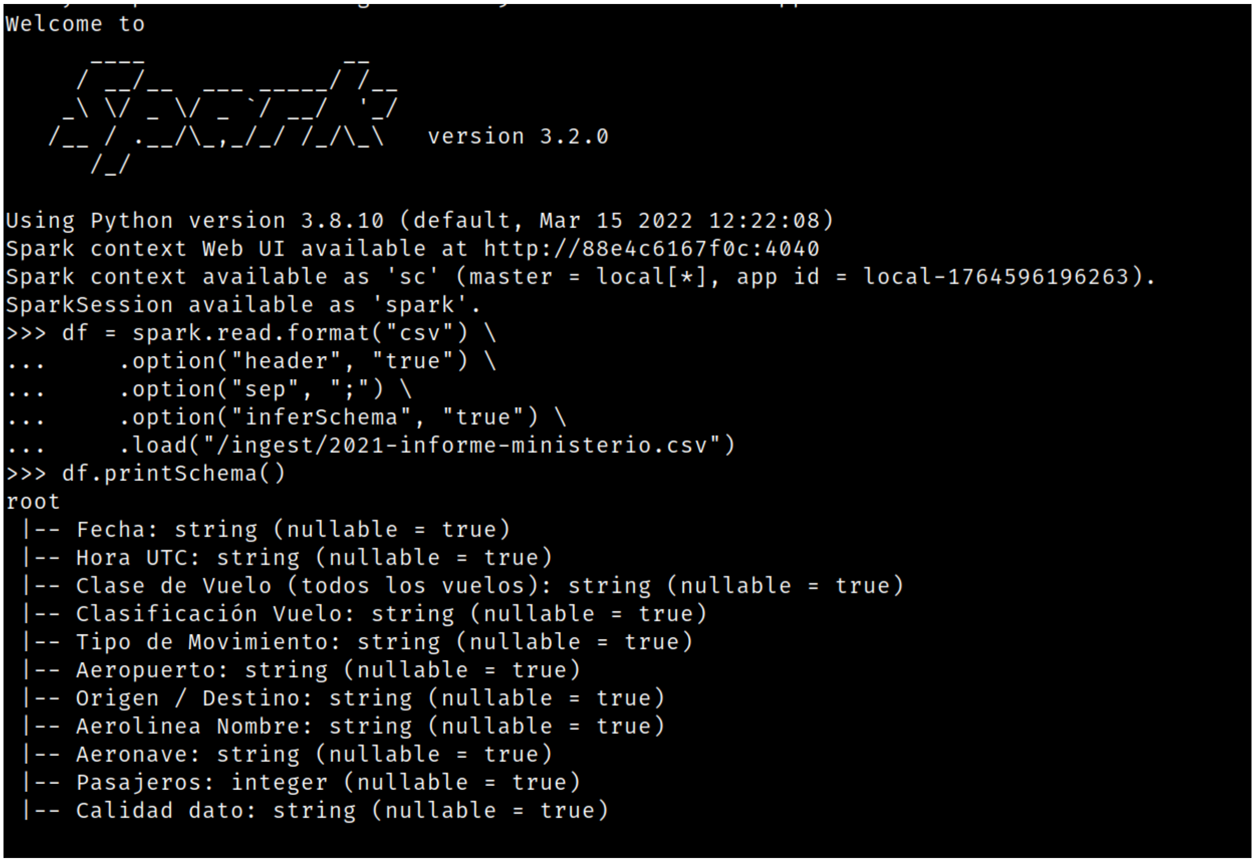
```
hadoop@88e4c6167f0c:~$ hdfs version
Hadoop 3.3.0
Source code repository https://gitbox.apache.org/repos/asf/hadoop.git -r aa96f1871bfd858f9bac59cf2a81ec470da649af
Compiled by brahma on 2020-07-06T18:44Z
Compiled with protoc 3.7.1
From source with checksum 5dc29b802d6ccd77b262ef9d04d19c4
This command was run using /home/hadoop/hadoop/share/hadoop/common/hadoop-common-3.3.0.jar
hadoop@88e4c6167f0c:~$ hdfs dfs -ls /
Found 6 items
drwxr-xr-x   - hadoop supergroup          0 2025-12-01 10:11 /ingest
drwxr-xr-x   - hadoop supergroup          0 2022-04-26 19:51 /inputs
drwxr-xr-x   - hadoop supergroup          0 2022-01-22 21:35 /logs
drwxr-xr-x   - hadoop supergroup          0 2023-03-06 11:05 /sqoop
drwxrwxr-x   - hadoop supergroup          0 2022-05-02 20:46 /tmp
drwxr-xr-x   - hadoop supergroup          0 2022-01-23 13:15 /user
hadoop@88e4c6167f0c:~$ hdfs dfs -ls /ingest
Found 8 items
-rw-r--r--   1 hadoop supergroup   32322556 2025-12-01 10:10 /ingest/2021-informe-ministerio.csv
-rw-r--r--   1 hadoop supergroup   22833520 2025-12-01 10:11 /ingest/202206-informe-ministerio.csv
-rw-r--r--   1 hadoop supergroup     533157 2025-11-13 15:23 /ingest/CarRentalData.csv
drwxr-xr-x   - hadoop supergroup          0 2025-11-12 09:18 /ingest/aeropuertos_detalle
-rw-r--r--   1 hadoop supergroup     136007 2025-12-01 10:10 /ingest/aeropuertos_detalle.csv
-rw-r--r--   1 hadoop supergroup    3380726 2025-11-13 15:23 /ingest/us_states_georef.csv
drwxr-xr-x   - hadoop supergroup          0 2025-11-12 09:18 /ingest/vuelos
hadoop@88e4c6167f0c:~$ hdfs dfs -cat /ingest/2021-informe-ministerio.csv | head -n 5

Fecha;Hora UTC;Clase de Vuelo (todos los vuelos);Clasificación Vuelo;Tipo de Movimiento;Aeropuerto;Origen / Destino;Aerolinea Nombre;Aeronave;Pasajeros;Calidad dato
01/01/2021;00:02;Vuelo Privado con Matrícula Nacional;Domestico;Despegue;PAR;ROS;0;PA-PA-28-181;0;DEFINITIVO
01/01/2021;00:24;Regular;Domestico;Aterrizaje;EZE;GRA;AEROLINEAS ARGENTINAS SA;BO-B737-8MB;70;DEFINITIVO
01/01/2021;00:26;Regular;Domestico;Aterrizaje;EZE;ECA;AEROLINEAS ARGENTINAS SA;BO-737-800;70;DEFINITIVO
01/01/2021;00:29;Regular;Domestico;Aterrizaje;EZE;SAL;AEROLINEAS ARGENTINAS SA;BO-B-737-76N;12;DEFINITIVO
cat: Unable to write to output stream.
```



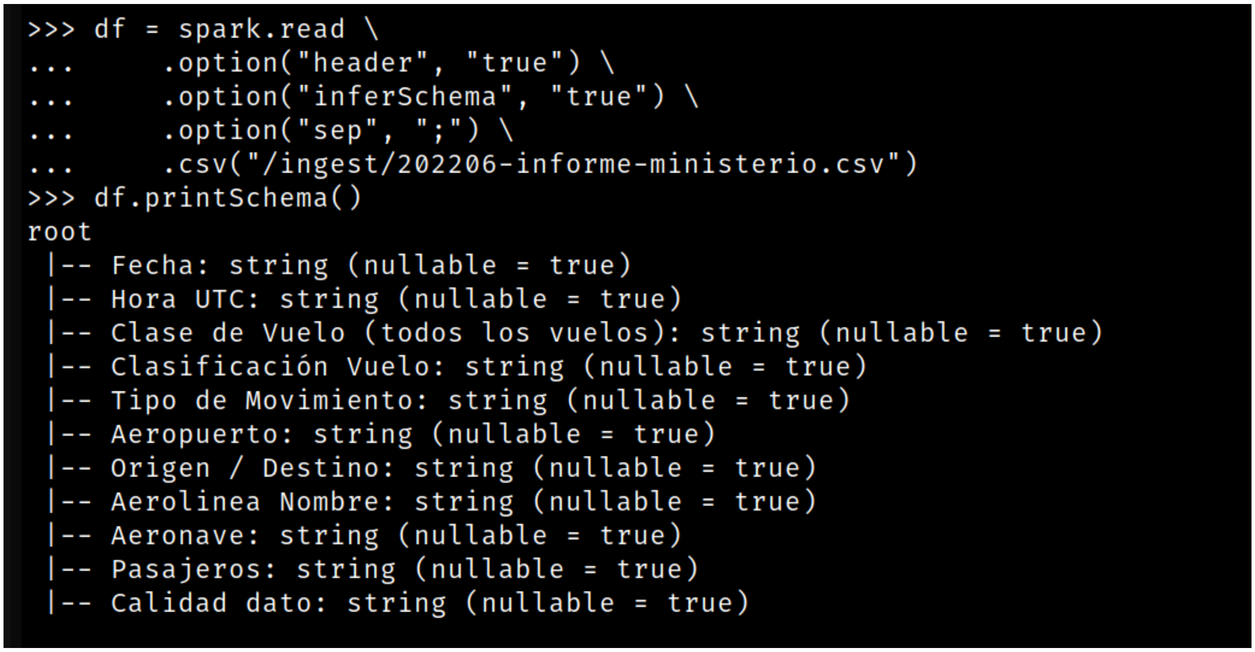
```
hadoop@88e4c6167f0c:~$ hdfs dfs -cat /ingest/202206-informe-ministerio.csv | head -n 5
Fecha;Hora UTC;Clase de Vuelo (todos los vuelos);Clasificación Vuelo;Tipo de Movimiento;Aeropuerto;Origen / Destino;Aerolinea Nombre;Aeronave;Pasajeros;Calidad dato
01/01/2022;00:01;Regular;Doméstico;Aterrizaje;AER;ECA;AEROLINEAS ARGENTINAS SA;BO-737-8SH;69;DEFINITIVO
01/01/2022;00:05;Regular;Doméstico;Aterrizaje;AER;SAL;AEROLINEAS ARGENTINAS SA;BO-B737-8;65;DEFINITIVO
01/01/2022;00:05;Regular;Doméstico;Despegue;IGU;AER;JETSMART AIRLINES S.A.;AIB-A320-232;41;DEFINITIVO
01/01/2022;00:09;Regular;Doméstico;Aterrizaje;AER;GAL;AEROLINEAS ARGENTINAS SA;BO-B737-81D;73;DEFINITIVO
cat: Unable to write to output stream.
hadoop@88e4c6167f0c:~$
```



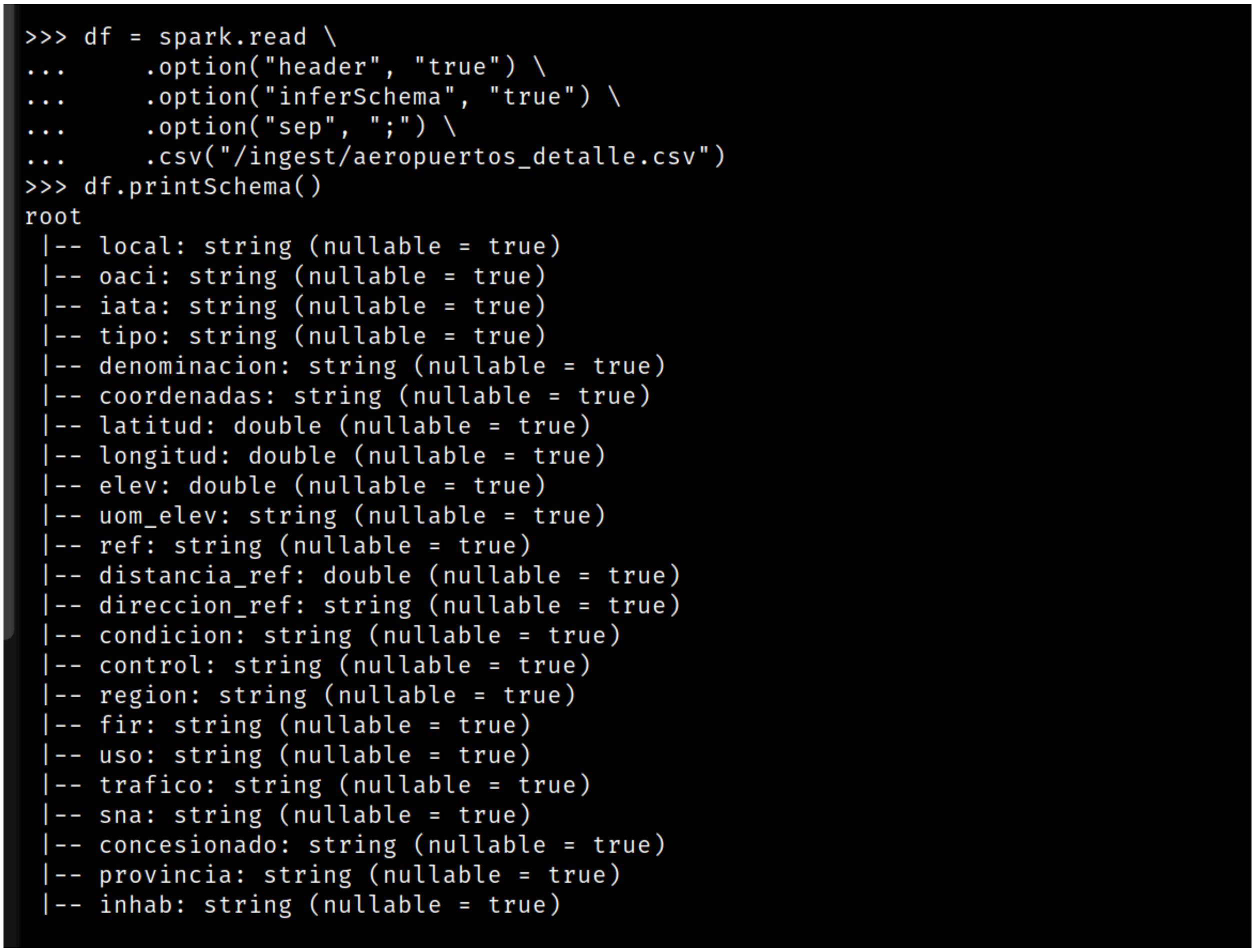
```
hadoop@88e4c6167f0c:~$ hdfs dfs -cat /ingest/aeropuertos_detalle.csv | head -n 5
local;oaci;iata;tipo;denominacion;coordenadas;latitud;longitud;elev;uom_elev;ref;distancia_ref;direccion_ref;condicion;control;region;fir;uso;trafico;sna;concesionado;provincia;inh
ab
ACB;;;Aeródromo;"CORONEL BOGADO/AGROSERVICIOS";"33°16'20""S  60°34'14""W";-60.57066000;-33.27226000;44.00;Metros;"Coronel Bogado";6.0;NE;PRIVADO;NOCONTROL;RACE;SAEF;AEROAPP;Naciona
l;NO;NO;"SANTA FÉ";NO
ACH;;;Aeródromo;"GENERAL ACHA";"37°24' 6""S  64°36'49""W";-64.61351000;-37.40164000;277.00;Metros;"General Acha";3.0;SO;PUBLICO;NOCONTROL;RACE;SAEF;CIVIL;Nacional;NO;NO;"LA PAMPA";
NO
ACM;;;Aeródromo;"ARRECIFES/LA CURA MALAL";"34° 4'33""S  60° 8'30""W";-60.14170000;-34.07574000;37.00;Metros;Arrecifes;4.0;OSO;PRIVADO;NOCONTROL;RACE;SAEF;CIVIL;Nacional;NO;NO;"BUEN
OS AIRES";NO
ADO;SAWD;PUD;Aeródromo;"PUERTO DESEADO";"47°44' 6""S  65°54'15""W";-65.90410000;-47.73511000;82.00;Metros;"Puerto Deseado";2.0;N;PUBLICO;AERADIO;RASU;SAVF;CIVIL;Nacional;NO;NO;"SAN
TA CRUZ";NO
cat: Unable to write to output stream.
```



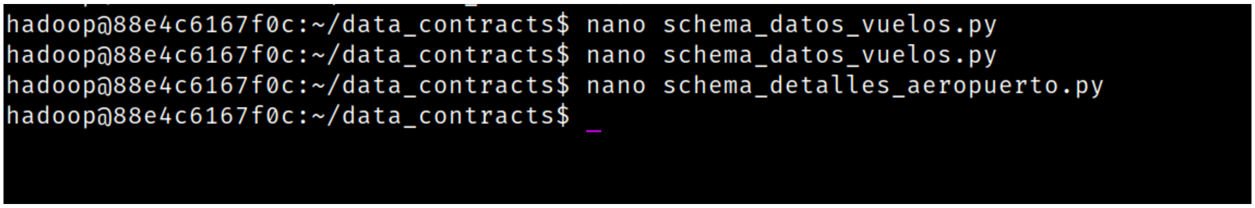
```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.2.0
      /_/

Using Python version 3.8.10 (default, Mar 15 2022 12:22:08)
Spark context Web UI available at http://88e4c6167f0c:4040
Spark context available as 'sc' (master = local[*], app id = local-1764596196263).
SparkSession available as 'spark'.
>>> df = spark.read.format("csv") \
...     .option("header", "true") \
...     .option("sep", ";") \
...     .option("inferSchema", "true") \
...     .load("/ingest/2021-informe-ministerio.csv")
>>> df.printSchema()
root
 |-- Fecha: string (nullable = true)
 |-- Hora UTC: string (nullable = true)
 |-- Clase de Vuelo (todos los vuelos): string (nullable = true)
 |-- Clasificación Vuelo: string (nullable = true)
 |-- Tipo de Movimiento: string (nullable = true)
 |-- Aeropuerto: string (nullable = true)
 |-- Origen / Destino: string (nullable = true)
 |-- Aerolinea Nombre: string (nullable = true)
 |-- Aeronave: string (nullable = true)
 |-- Pasajeros: integer (nullable = true)
 |-- Calidad dato: string (nullable = true)
```



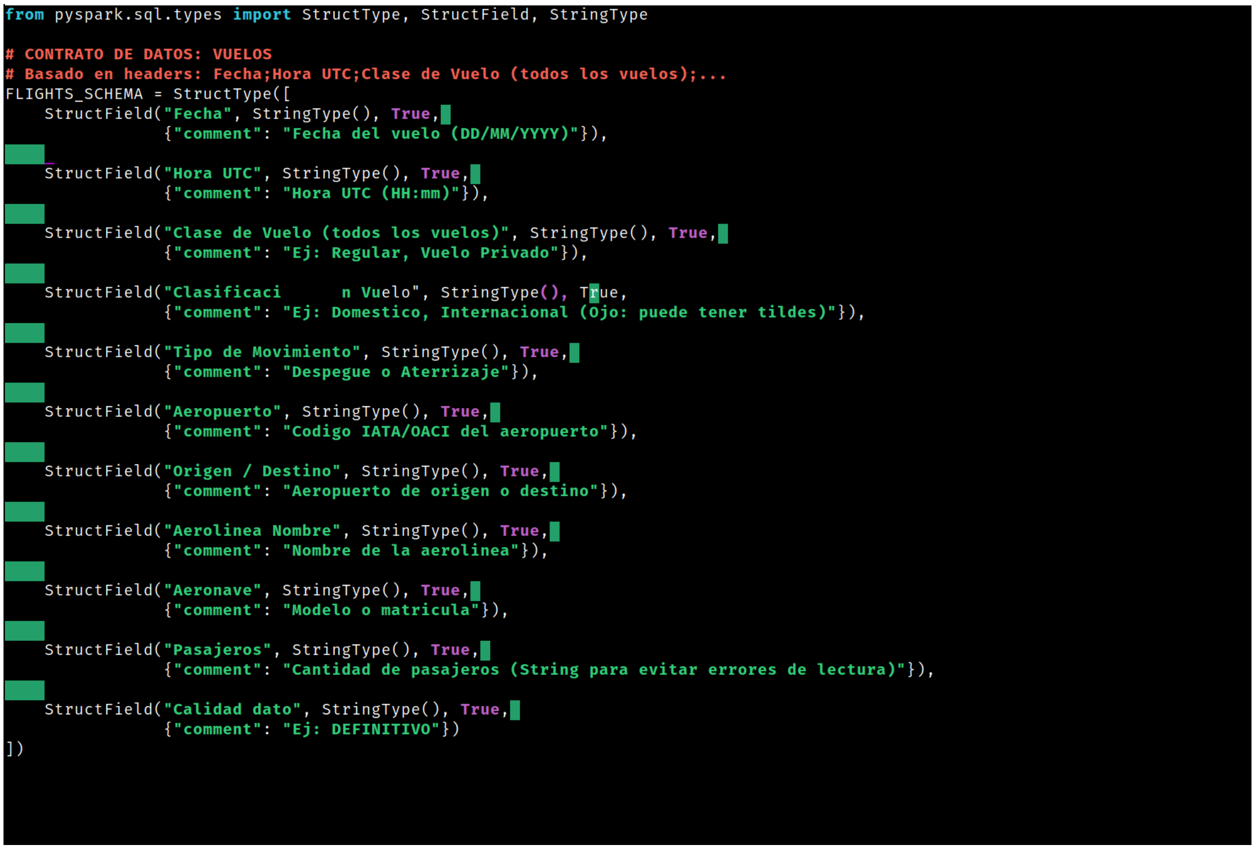
```
>>> df = spark.read \
...     .option("header", "true") \
...     .option("inferSchema", "true") \
...     .option("sep", ";") \
...     .csv("/ingest/202206-informe-ministerio.csv")
>>> df.printSchema()
root
 |-- Fecha: string (nullable = true)
 |-- Hora UTC: string (nullable = true)
 |-- Clase de Vuelo (todos los vuelos): string (nullable = true)
 |-- Clasificación Vuelo: string (nullable = true)
 |-- Tipo de Movimiento: string (nullable = true)
 |-- Aeropuerto: string (nullable = true)
 |-- Origen / Destino: string (nullable = true)
 |-- Aerolinea Nombre: string (nullable = true)
 |-- Aeronave: string (nullable = true)
 |-- Pasajeros: string (nullable = true)
 |-- Calidad dato: string (nullable = true)
```



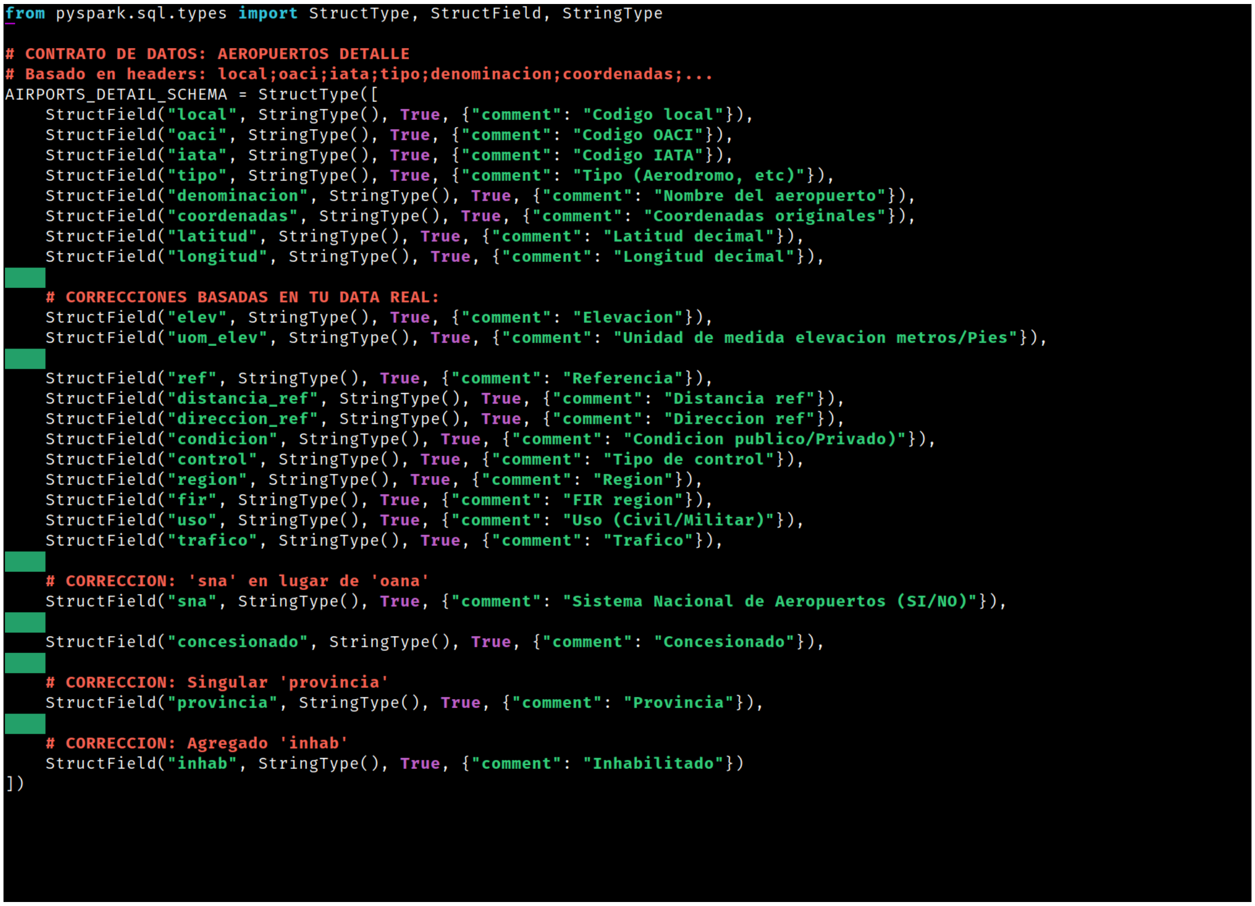
```
>>> df = spark.read \
...     .option("header", "true") \
...     .option("inferSchema", "true") \
...     .option("sep", ";") \
...     .csv("/ingest/aeropuertos_detalle.csv")
>>> df.printSchema()
root
 |-- local: string (nullable = true)
 |-- oaci: string (nullable = true)
 |-- iata: string (nullable = true)
 |-- tipo: string (nullable = true)
 |-- denominacion: string (nullable = true)
 |-- coordenadas: string (nullable = true)
 |-- latitud: double (nullable = true)
 |-- longitud: double (nullable = true)
 |-- elev: double (nullable = true)
 |-- uom_elev: string (nullable = true)
 |-- ref: string (nullable = true)
 |-- distancia_ref: double (nullable = true)
 |-- direccion_ref: string (nullable = true)
 |-- condicion: string (nullable = true)
 |-- control: string (nullable = true)
 |-- region: string (nullable = true)
 |-- fir: string (nullable = true)
 |-- uso: string (nullable = true)
 |-- trafico: string (nullable = true)
 |-- sna: string (nullable = true)
 |-- concesionado: string (nullable = true)
 |-- provincia: string (nullable = true)
 |-- inhab: string (nullable = true)
```



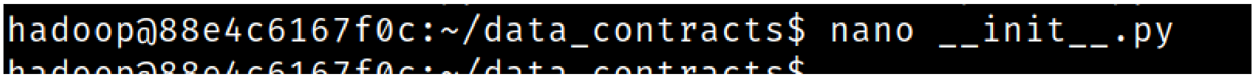
```
hadoop@88e4c6167f0c:~/data_contracts$ nano schema_datos_vuelos.py
hadoop@88e4c6167f0c:~/data_contracts$ nano schema_datos_vuelos.py
hadoop@88e4c6167f0c:~/data_contracts$ nano schema_detalles_aeropuerto.py
hadoop@88e4c6167f0c:~/data_contracts$ _
```



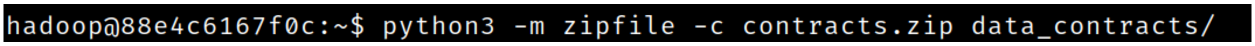
```
from pyspark.sql.types import StructType, StructField, StringType

# CONTRATO DE DATOS: VUELOS
# Basado en headers: Fecha;Hora UTC;Clase de Vuelo (todos los vuelos);...
FLIGHTS_SCHEMA = StructType([
    StructField("Fecha", StringType(), True,
                {"comment": "Fecha del vuelo (DD/MM/YYYY)"}),

    StructField("Hora UTC", StringType(), True,
                {"comment": "Hora UTC (HH:mm)"}),

    StructField("Clase de Vuelo (todos los vuelos)", StringType(), True,
                {"comment": "Ej: Regular, Vuelo Privado"}),

    StructField("Clasificaci      n Vuelo", StringType(), True,
                {"comment": "Ej: Domestico, Internacional (Ojo: puede tener tildes)"}),

    StructField("Tipo de Movimiento", StringType(), True,
                {"comment": "Despegue o Aterrizaje"}),

    StructField("Aeropuerto", StringType(), True,
                {"comment": "Codigo IATA/OACI del aeropuerto"}),

    StructField("Origen / Destino", StringType(), True,
                {"comment": "Aeropuerto de origen o destino"}),

    StructField("Aerolinea Nombre", StringType(), True,
                {"comment": "Nombre de la aerolinea"}),

    StructField("Aeronave", StringType(), True,
                {"comment": "Modelo o matricula"}),

    StructField("Pasajeros", StringType(), True,
                {"comment": "Cantidad de pasajeros (String para evitar errores de lectura)"}),

    StructField("Calidad dato", StringType(), True,
                {"comment": "Ej: DEFINITIVO"})
])
```



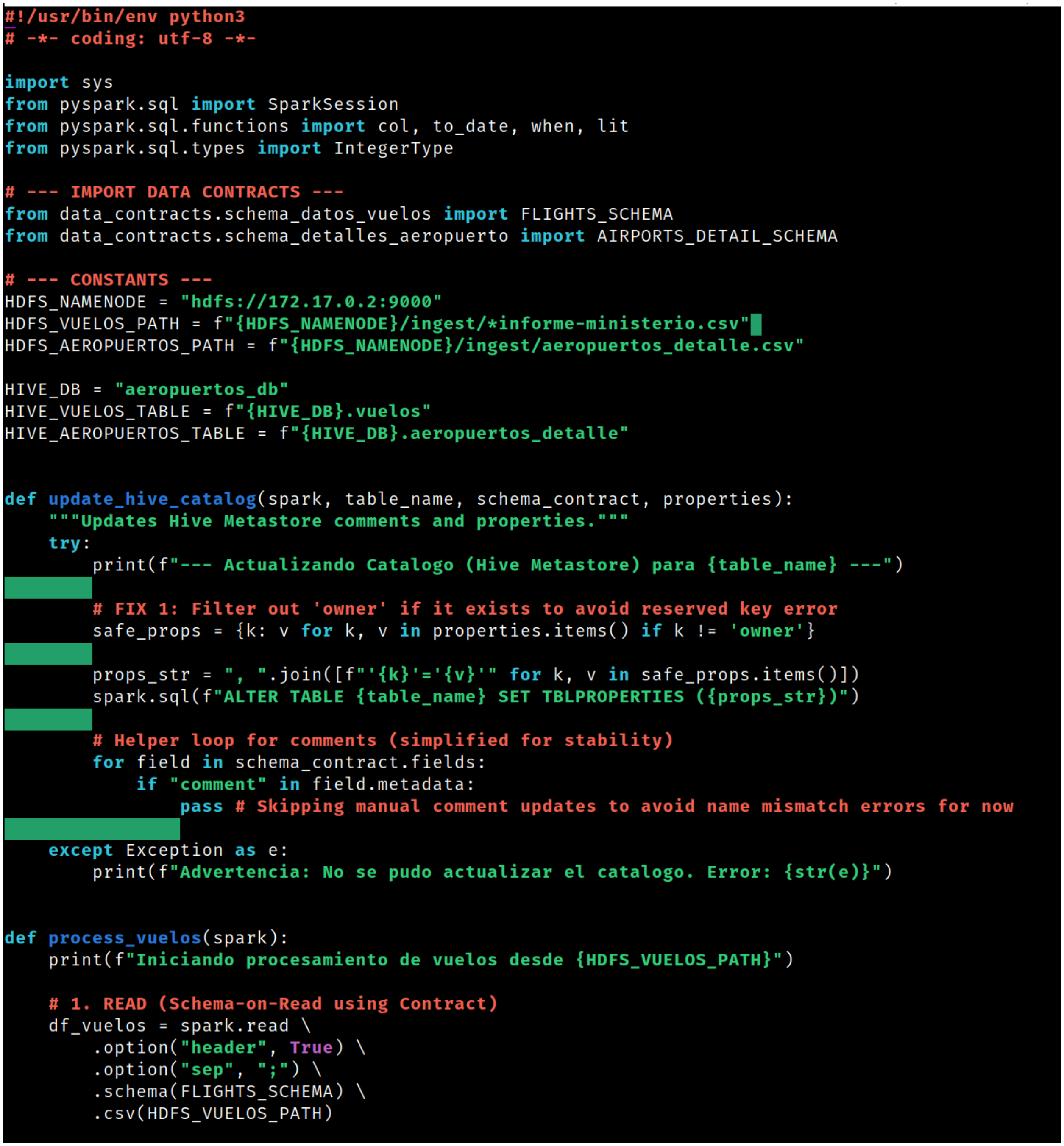
```
from pyspark.sql.types import StructType, StructField, StringType

# CONTRATO DE DATOS: AEROPUERTOS DETALLE
# Basado en headers: local;oaci;iata;tipo;denominacion;coordenadas;...
AIRPORTS_DETAIL_SCHEMA = StructType([
    StructField("local", StringType(), True, {"comment": "Codigo local"}),
    StructField("oaci", StringType(), True, {"comment": "Codigo OACI"}),
    StructField("iata", StringType(), True, {"comment": "Codigo IATA"}),
    StructField("tipo", StringType(), True, {"comment": "Tipo (Aerodromo, etc)"}),
    StructField("denominacion", StringType(), True, {"comment": "Nombre del aeropuerto"}),
    StructField("coordenadas", StringType(), True, {"comment": "Coordenadas originales"}),
    StructField("latitud", StringType(), True, {"comment": "Latitud decimal"}),
    StructField("longitud", StringType(), True, {"comment": "Longitud decimal"}),

    # CORRECCIONES BASADAS EN TU DATA REAL:
    StructField("elev", StringType(), True, {"comment": "Elevacion"}),
    StructField("uom_elev", StringType(), True, {"comment": "Unidad de medida elevacion metros/Pies"}),

    StructField("ref", StringType(), True, {"comment": "Referencia"}),
    StructField("distancia_ref", StringType(), True, {"comment": "Distancia ref"}),
    StructField("direccion_ref", StringType(), True, {"comment": "Direccion ref"}),
    StructField("condicion", StringType(), True, {"comment": "Condicion publico/Privado)"}),
    StructField("control", StringType(), True, {"comment": "Tipo de control"}),
    StructField("region", StringType(), True, {"comment": "Region"}),
    StructField("fir", StringType(), True, {"comment": "FIR region"}),
    StructField("uso", StringType(), True, {"comment": "Uso (Civil/Militar)"}),
    StructField("trafico", StringType(), True, {"comment": "Trafico"}),

    # CORRECCION: 'sna' en lugar de 'oana'
    StructField("sna", StringType(), True, {"comment": "Sistema Nacional de Aeropuertos (SI/NO)"}),

    StructField("concesionado", StringType(), True, {"comment": "Concesionado"}),

    # CORRECCION: Singular 'provincia'
    StructField("provincia", StringType(), True, {"comment": "Provincia"}),

    # CORRECCION: Agregado 'inhab'
    StructField("inhab", StringType(), True, {"comment": "Inhabilitado"})
])
```



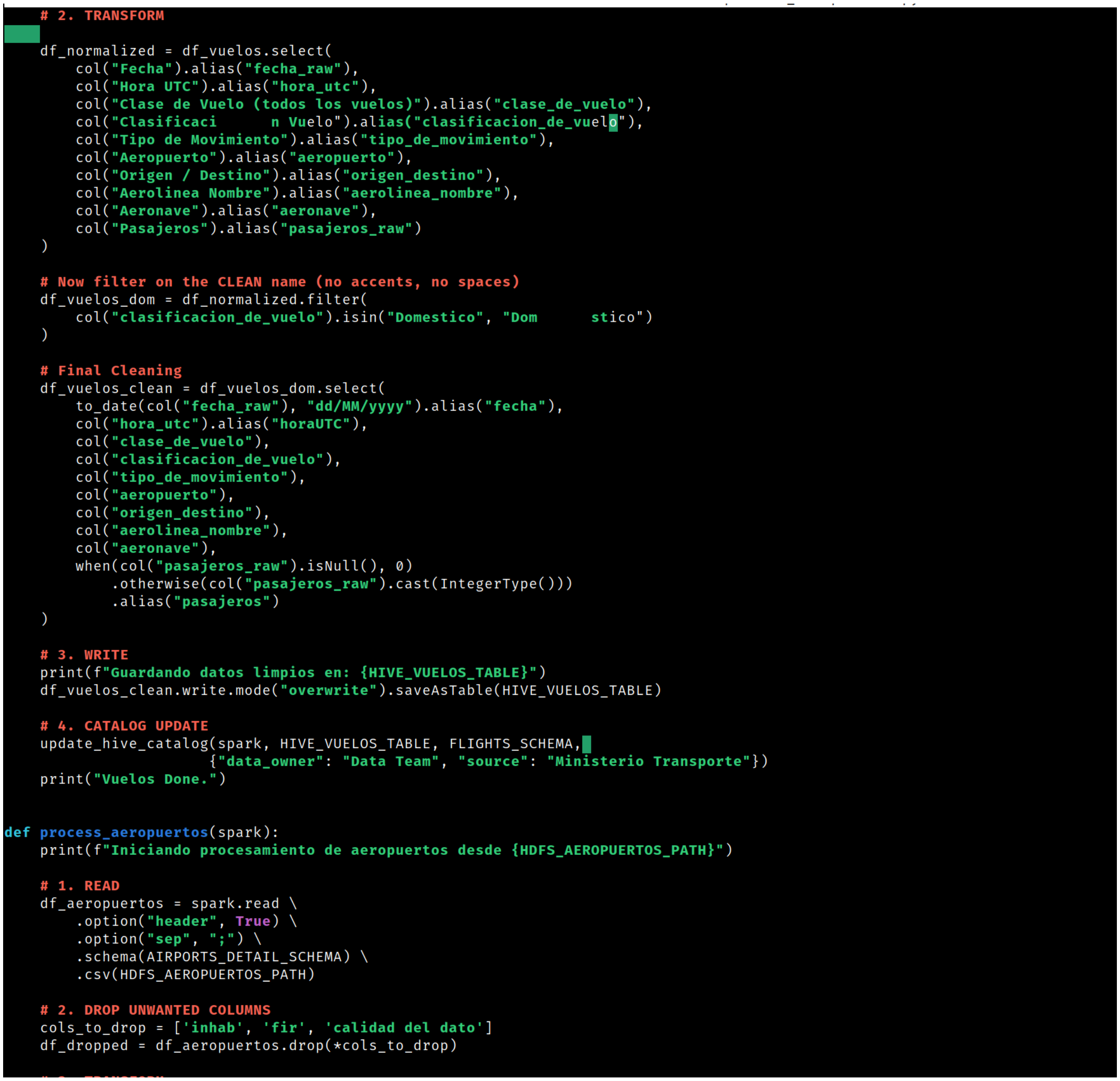
```
hadoop@88e4c6167f0c:~/data_contracts$ nano __init__.py
hadoop@88e4c6167f0c:~/data_contracts$
```



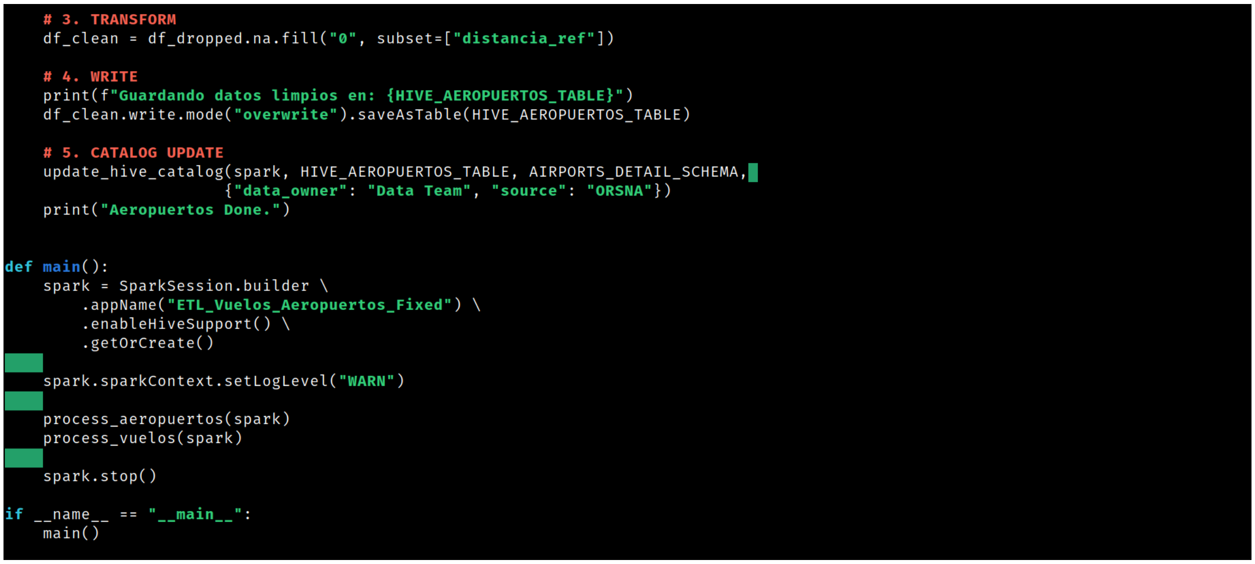
```
hadoop@88e4c6167f0c:~$ python3 -m zipfile -c contracts.zip data_contracts/
```



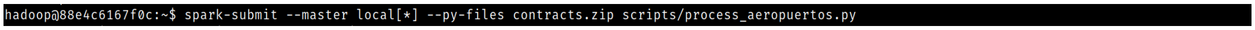
```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, to_date, when, lit
from pyspark.sql.types import IntegerType

# --- IMPORT DATA CONTRACTS ---
from data_contracts.schema_datos_vuelos import FLIGHTS_SCHEMA
from data_contracts.schema_detalles_aeropuerto import AIRPORTS_DETAIL_SCHEMA

# --- CONSTANTS ---
HDFS_NAMENODE = "hdfs://172.17.0.2:9000"
HDFS_VUELOS_PATH = f"{HDFS_NAMENODE}/ingest/*informe-ministerio.csv"
HDFS_AEROPUERTOS_PATH = f"{HDFS_NAMENODE}/ingest/aeropuertos_detalle.csv"

HIVE_DB = "aeropuertos_db"
HIVE_VUELOS_TABLE = f"{HIVE_DB}.vuelos"
HIVE_AEROPUERTOS_TABLE = f"{HIVE_DB}.aeropuertos_detalle"


def update_hive_catalog(spark, table_name, schema_contract, properties):
    """Updates Hive Metastore comments and properties."""
    try:
        print(f"--- Actualizando Catalogo (Hive Metastore) para {table_name} ---")

        # FIX 1: Filter out 'owner' if it exists to avoid reserved key error
        safe_props = {k: v for k, v in properties.items() if k != 'owner'}

        props_str = ", ".join([f"'{k}'='{v}'" for k, v in safe_props.items()])
        spark.sql(f"ALTER TABLE {table_name} SET TBLPROPERTIES ({props_str})")

        # Helper loop for comments (simplified for stability)
        for field in schema_contract.fields:
            if "comment" in field.metadata:
                pass # Skipping manual comment updates to avoid name mismatch errors for now

    except Exception as e:
        print(f"Advertencia: No se pudo actualizar el catalogo. Error: {str(e)}")


def process_vuelos(spark):
    print(f"Iniciando procesamiento de vuelos desde {HDFS_VUELOS_PATH}")

    # 1. READ (Schema-on-Read using Contract)
    df_vuelos = spark.read \
        .option("header", True) \
        .option("sep", ";") \
        .schema(FLIGHTS_SCHEMA) \
        .csv(HDFS_VUELOS_PATH)
```



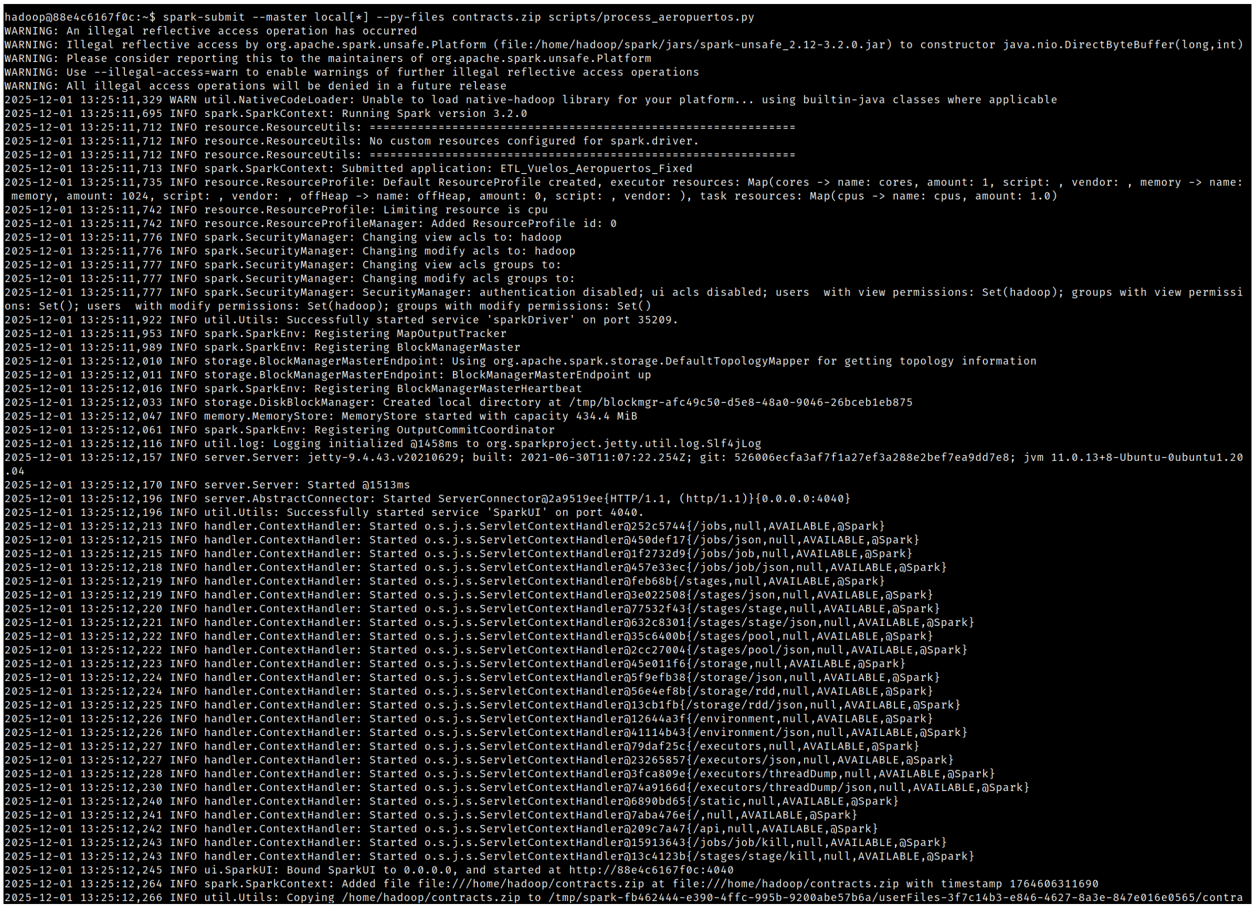
```
    # 2. TRANSFORM

    df_normalized = df_vuelos.select(
        col("Fecha").alias("fecha_raw"),
        col("Hora UTC").alias("hora_utc"),
        col("Clase de Vuelo (todos los vuelos)").alias("clase_de_vuelo"),
        col("Clasificaci      n Vuelo").alias("clasificacion_de_vuelo"),
        col("Tipo de Movimiento").alias("tipo_de_movimiento"),
        col("Aeropuerto").alias("aeropuerto"),
        col("Origen / Destino").alias("origen_destino"),
        col("Aerolinea Nombre").alias("aerolinea_nombre"),
        col("Aeronave").alias("aeronave"),
        col("Pasajeros").alias("pasajeros_raw")
    )

    # Now filter on the CLEAN name (no accents, no spaces)
    df_vuelos_dom = df_normalized.filter(
        col("clasificacion_de_vuelo").isin("Domestico", "Dom      stico")
    )

    # Final Cleaning
    df_vuelos_clean = df_vuelos_dom.select(
        to_date(col("fecha_raw"), "dd/MM/yyyy").alias("fecha"),
        col("hora_utc").alias("horaUTC"),
        col("clase_de_vuelo"),
        col("clasificacion_de_vuelo"),
        col("tipo_de_movimiento"),
        col("aeropuerto"),
        col("origen_destino"),
        col("aerolinea_nombre"),
        col("aeronave"),
        when(col("pasajeros_raw").isNull(), 0)
            .otherwise(col("pasajeros_raw").cast(IntegerType()))
            .alias("pasajeros")
    )

    # 3. WRITE
    print(f"Guardando datos limpios en: {HIVE_VUELOS_TABLE}")
    df_vuelos_clean.write.mode("overwrite").saveAsTable(HIVE_VUELOS_TABLE)

    # 4. CATALOG UPDATE
    update_hive_catalog(spark, HIVE_VUELOS_TABLE, FLIGHTS_SCHEMA,
                       {"data_owner": "Data Team", "source": "Ministerio Transporte"})
    print("Vuelos Done.")


def process_aeropuertos(spark):
    print(f"Iniciando procesamiento de aeropuertos desde {HDFS_AEROPUERTOS_PATH}")

    # 1. READ
    df_aeropuertos = spark.read \
        .option("header", True) \
        .option("sep", ";") \
        .schema(AIRPORTS_DETAIL_SCHEMA) \
        .csv(HDFS_AEROPUERTOS_PATH)

    # 2. DROP UNWANTED COLUMNS
    cols_to_drop = ['inhab', 'fir', 'calidad del dato']
    df_dropped = df_aeropuertos.drop(*cols_to_drop)

    # 3. TRANSFORM
```



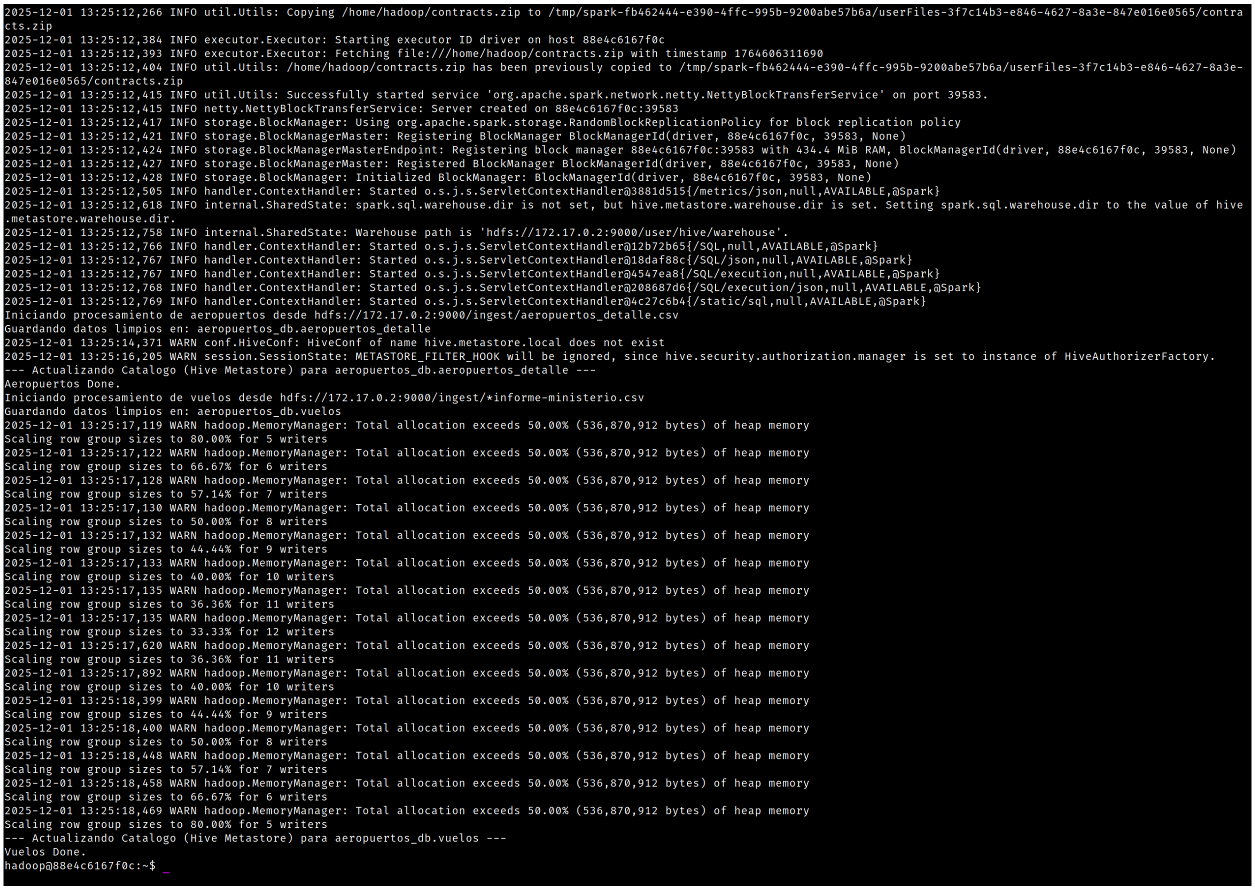
```
    # 3. TRANSFORM
    df_clean = df_dropped.na.fill("0", subset=["distancia_ref"])

    # 4. WRITE
    print(f"Guardando datos limpios en: {HIVE_AEROPUERTOS_TABLE}")
    df_clean.write.mode("overwrite").saveAsTable(HIVE_AEROPUERTOS_TABLE)

    # 5. CATALOG UPDATE
    update_hive_catalog(spark, HIVE_AEROPUERTOS_TABLE, AIRPORTS_DETAIL_SCHEMA,
                       {"data_owner": "Data Team", "source": "ORSNA"})
    print("Aeropuertos Done.")


def main():
    spark = SparkSession.builder \
        .appName("ETL_Vuelos_Aeropuertos_Fixed") \
        .enableHiveSupport() \
        .getOrCreate()

    spark.sparkContext.setLogLevel("WARN")

    process_aeropuertos(spark)
    process_vuelos(spark)

    spark.stop()

if __name__ == "__main__":
    main()
```



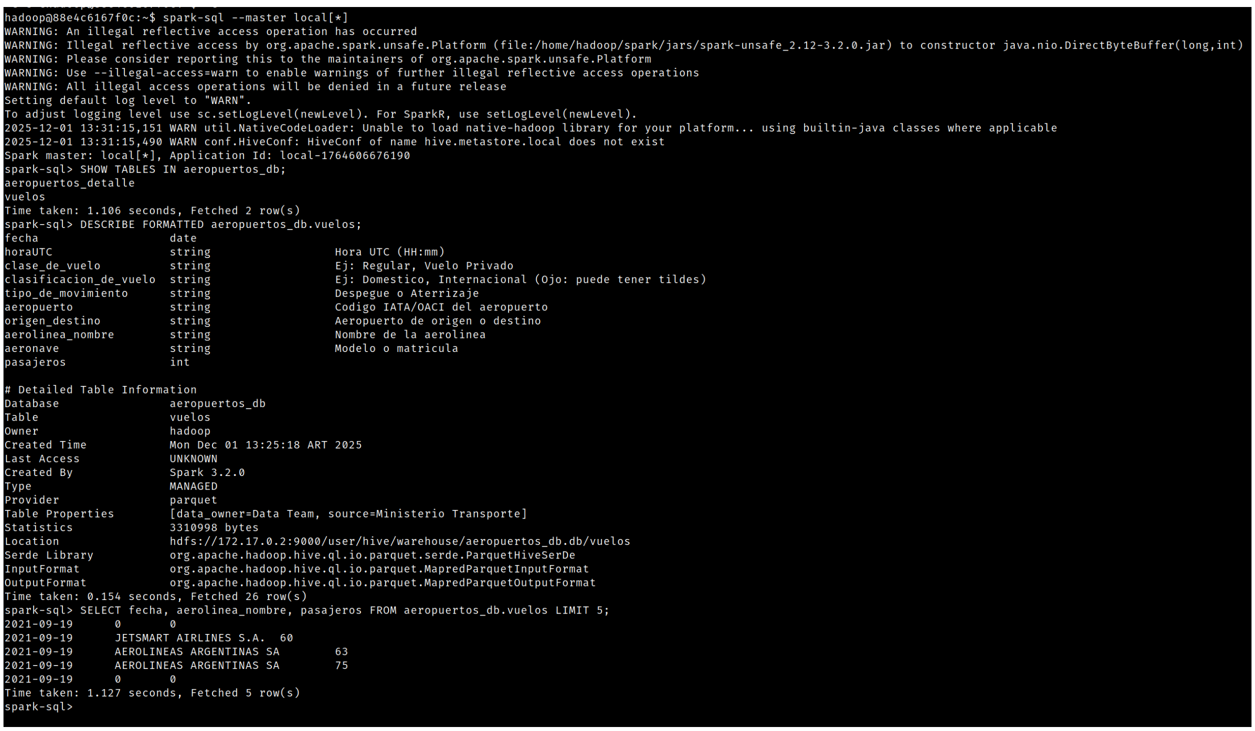
```
hadoop@88e4c6167f0c:~$ spark-submit --master local[*] --py-files contracts.zip scripts/process_aeropuertos.py
```



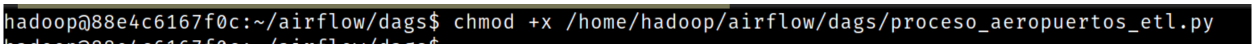
```
hadoop@88e4c6167f0c:~$ spark-submit --master local[*] --py-files contracts.zip scripts/process_aeropuertos.py
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/home/hadoop/spark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
2025-12-01 13:25:11,329 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2025-12-01 13:25:11,695 INFO spark.SparkContext: Running Spark version 3.2.0
2025-12-01 13:25:11,712 INFO resource.ResourceUtils: ==============================================================
2025-12-01 13:25:11,712 INFO resource.ResourceUtils: No custom resources configured for spark.driver.
2025-12-01 13:25:11,712 INFO resource.ResourceUtils: ==============================================================
2025-12-01 13:25:11,713 INFO spark.SparkContext: Submitted application: ETL_Vuelos_Aeropuertos_Fixed
2025-12-01 13:25:11,735 INFO resource.ResourceProfile: Default ResourceProfile created, executor resources: Map(cores -> name: cores, amount: 1, script: , vendor: , memory -> name:
 memory, amount: 1024, script: , vendor: , offHeap -> name: offHeap, amount: 0, script: , vendor: ), task resources: Map(cpus -> name: cpus, amount: 1.0)
2025-12-01 13:25:11,742 INFO resource.ResourceProfile: Limiting resource is cpu
2025-12-01 13:25:11,742 INFO resource.ResourceProfileManager: Added ResourceProfile id: 0
2025-12-01 13:25:11,776 INFO spark.SecurityManager: Changing view acls to: hadoop
2025-12-01 13:25:11,776 INFO spark.SecurityManager: Changing modify acls to: hadoop
2025-12-01 13:25:11,777 INFO spark.SecurityManager: Changing view acls groups to:
2025-12-01 13:25:11,777 INFO spark.SecurityManager: Changing modify acls groups to:
2025-12-01 13:25:11,777 INFO spark.SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  with view permissions: Set(hadoop); groups with view permissi
ons: Set(); users  with modify permissions: Set(hadoop); groups with modify permissions: Set()
2025-12-01 13:25:11,922 INFO util.Utils: Successfully started service 'sparkDriver' on port 35209.
2025-12-01 13:25:11,953 INFO spark.SparkEnv: Registering MapOutputTracker
2025-12-01 13:25:11,989 INFO spark.SparkEnv: Registering BlockManagerMaster
2025-12-01 13:25:12,010 INFO storage.BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology information
2025-12-01 13:25:12,011 INFO storage.BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
2025-12-01 13:25:12,016 INFO spark.SparkEnv: Registering BlockManagerMasterHeartbeat
2025-12-01 13:25:12,033 INFO storage.DiskBlockManager: Created local directory at /tmp/blockmgr-afc49c50-d5e8-48a0-9046-26bceb1eb875
2025-12-01 13:25:12,047 INFO memory.MemoryStore: MemoryStore started with capacity 434.4 MiB
2025-12-01 13:25:12,061 INFO spark.SparkEnv: Registering OutputCommitCoordinator
2025-12-01 13:25:12,116 INFO util.log: Logging initialized @1458ms to org.sparkproject.jetty.util.log.Slf4jLog
2025-12-01 13:25:12,157 INFO server.Server: jetty-9.4.43.v20210629; built: 2021-06-30T11:07:22.254Z; git: 526006ecfa3af7f1a27ef3a288e2bef7ea9dd7e8; jvm 11.0.13+8-Ubuntu-0ubuntu1.20
.04
2025-12-01 13:25:12,170 INFO server.Server: Started @1513ms
2025-12-01 13:25:12,196 INFO server.AbstractConnector: Started ServerConnector@2a9519ee{HTTP/1.1, (http/1.1)}{0.0.0.0:4040}
2025-12-01 13:25:12,196 INFO util.Utils: Successfully started service 'SparkUI' on port 4040.
2025-12-01 13:25:12,213 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@252c5744{/jobs,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,215 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@450def17{/jobs/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,215 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@1f2732d9{/jobs/job,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,218 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@457e33ec{/jobs/job/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,219 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@feb68b{/stages,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,219 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@3e022508{/stages/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,220 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@77532f43{/stages/stage,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,221 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@632c8301{/stages/stage/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,222 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@35c6400b{/stages/pool,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,222 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@2cc27004{/stages/pool/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,223 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@45e011f6{/storage,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,224 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@5f9efb38{/storage/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,224 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@56e4ef8b{/storage/rdd,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,225 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@13cb1fb{/storage/rdd/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,226 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@12644a3f{/environment,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,226 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@41114b43{/environment/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,227 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@79daf25c{/executors,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,227 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@23265857{/executors/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,228 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@3fca809e{/executors/threadDump,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,230 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@74a9166d{/executors/threadDump/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,240 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@6890bd65{/static,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,241 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@7aba476e{/,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,242 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@209c7a47{/api,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,243 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@15913643{/jobs/job/kill,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,243 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@13c4123b{/stages/stage/kill,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,245 INFO ui.SparkUI: Bound SparkUI to 0.0.0.0, and started at http://88e4c6167f0c:4040
2025-12-01 13:25:12,264 INFO spark.SparkContext: Added file file:///home/hadoop/contracts.zip at file:///home/hadoop/contracts.zip with timestamp 1764606311690
2025-12-01 13:25:12,266 INFO util.Utils: Copying /home/hadoop/contracts.zip to /tmp/spark-fb462444-e390-4ffc-995b-9200abe57b6a/userFiles-3f7c14b3-e846-4627-8a3e-847e016e0565/contra
```



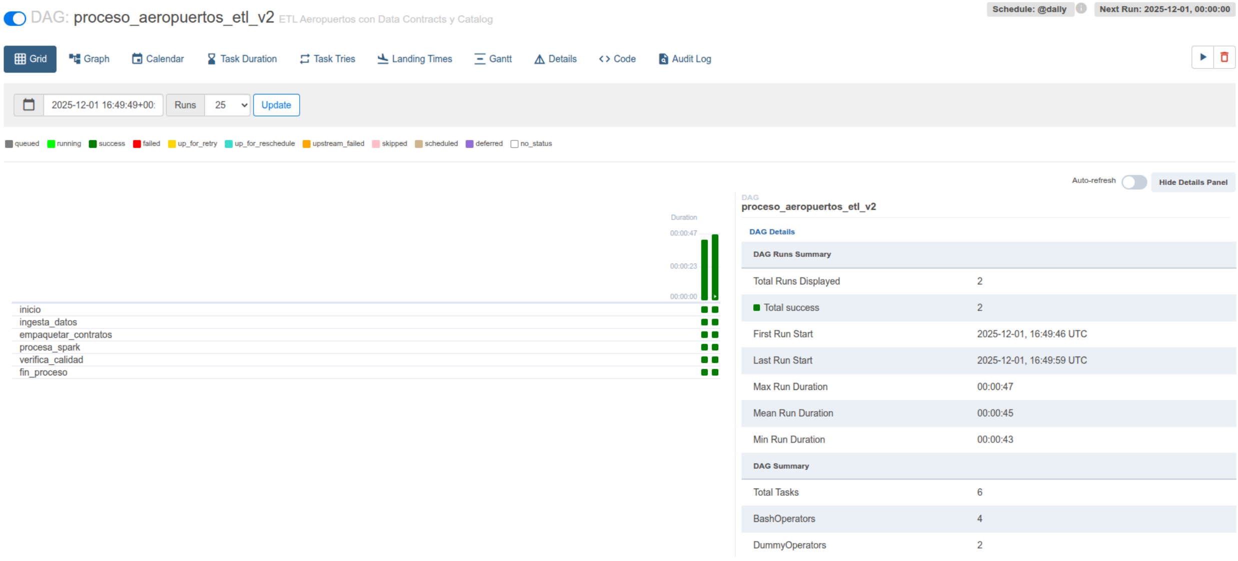
```
2025-12-01 13:25:12,266 INFO util.Utils: Copying /home/hadoop/contracts.zip to /tmp/spark-fb462444-e390-4ffc-995b-9200abe57b6a/userFiles-3f7c14b3-e846-4627-8a3e-847e016e0565/contra
cts.zip
2025-12-01 13:25:12,384 INFO executor.Executor: Starting executor ID driver on host 88e4c6167f0c
2025-12-01 13:25:12,393 INFO executor.Executor: Fetching file:///home/hadoop/contracts.zip with timestamp 1764606311690
2025-12-01 13:25:12,404 INFO util.Utils: /home/hadoop/contracts.zip has been previously copied to /tmp/spark-fb462444-e390-4ffc-995b-9200abe57b6a/userFiles-3f7c14b3-e846-4627-8a3e-
847e016e0565/contracts.zip
2025-12-01 13:25:12,415 INFO util.Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 39583.
2025-12-01 13:25:12,415 INFO netty.NettyBlockTransferService: Server created on 88e4c6167f0c:39583
2025-12-01 13:25:12,417 INFO storage.BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block replication policy
2025-12-01 13:25:12,421 INFO storage.BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 88e4c6167f0c, 39583, None)
2025-12-01 13:25:12,424 INFO storage.BlockManagerMasterEndpoint: Registering block manager 88e4c6167f0c:39583 with 434.4 MiB RAM, BlockManagerId(driver, 88e4c6167f0c, 39583, None)
2025-12-01 13:25:12,427 INFO storage.BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 88e4c6167f0c, 39583, None)
2025-12-01 13:25:12,428 INFO storage.BlockManager: Initialized BlockManager: BlockManagerId(driver, 88e4c6167f0c, 39583, None)
2025-12-01 13:25:12,505 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@3881d515{/metrics/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,618 INFO internal.SharedState: spark.sql.warehouse.dir is not set, but hive.metastore.warehouse.dir is set. Setting spark.sql.warehouse.dir to the value of hive
.metastore.warehouse.dir.
2025-12-01 13:25:12,758 INFO internal.SharedState: Warehouse path is 'hdfs://172.17.0.2:9000/user/hive/warehouse'.
2025-12-01 13:25:12,766 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@12b72b65{/SQL,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,767 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@18daf88c{/SQL/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,767 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@4547ea8{/SQL/execution,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,768 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@208687d6{/SQL/execution/json,null,AVAILABLE,@Spark}
2025-12-01 13:25:12,769 INFO handler.ContextHandler: Started o.s.j.s.ServletContextHandler@4c27c6b4{/static/sql,null,AVAILABLE,@Spark}
Iniciando procesamiento de aeropuertos desde hdfs://172.17.0.2:9000/ingest/aeropuertos_detalle.csv
Guardando datos limpios en: aeropuertos_db.aeropuertos_detalle
2025-12-01 13:25:14,371 WARN conf.HiveConf: HiveConf of name hive.metastore.local does not exist
2025-12-01 13:25:16,205 WARN session.SessionState: METASTORE_FILTER_HOOK will be ignored, since hive.security.authorization.manager is set to instance of HiveAuthorizerFactory.
--- Actualizando Catalogo (Hive Metastore) para aeropuertos_db.aeropuertos_detalle ---
Aeropuertos Done.
Iniciando procesamiento de vuelos desde hdfs://172.17.0.2:9000/ingest/*informe-ministerio.csv
Guardando datos limpios en: aeropuertos_db.vuelos
2025-12-01 13:25:17,119 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 80.00% for 5 writers
2025-12-01 13:25:17,122 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 66.67% for 6 writers
2025-12-01 13:25:17,128 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 57.14% for 7 writers
2025-12-01 13:25:17,130 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 50.00% for 8 writers
2025-12-01 13:25:17,132 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 44.44% for 9 writers
2025-12-01 13:25:17,133 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 40.00% for 10 writers
2025-12-01 13:25:17,135 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 36.36% for 11 writers
2025-12-01 13:25:17,135 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 33.33% for 12 writers
2025-12-01 13:25:17,620 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 36.36% for 11 writers
2025-12-01 13:25:17,892 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 40.00% for 10 writers
2025-12-01 13:25:18,399 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 44.44% for 9 writers
2025-12-01 13:25:18,400 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 50.00% for 8 writers
2025-12-01 13:25:18,448 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 57.14% for 7 writers
2025-12-01 13:25:18,458 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 66.67% for 6 writers
2025-12-01 13:25:18,469 WARN hadoop.MemoryManager: Total allocation exceeds 50.00% (536,870,912 bytes) of heap memory
Scaling row group sizes to 80.00% for 5 writers
--- Actualizando Catalogo (Hive Metastore) para aeropuertos_db.vuelos ---
Vuelos Done.
hadoop@88e4c6167f0c:~$ _
```



```
hadoop@88e4c6167f0c:~$ spark-sql --master local[*]
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.spark.unsafe.Platform (file:/home/hadoop/spark/jars/spark-unsafe_2.12-3.2.0.jar) to constructor java.nio.DirectByteBuffer(long,int)
WARNING: Please consider reporting this to the maintainers of org.apache.spark.unsafe.Platform
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
2025-12-01 13:31:15,151 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2025-12-01 13:31:15,490 WARN conf.HiveConf: HiveConf of name hive.metastore.local does not exist
Spark master: local[*], Application Id: local-1764606676190
spark-sql> SHOW TABLES IN aeropuertos_db;
aeropuertos_detalle
vuelos
Time taken: 1.106 seconds, Fetched 2 row(s)
spark-sql> DESCRIBE FORMATTED aeropuertos_db.vuelos;
fecha                   date
horaUTC                 string                  Hora UTC (HH:mm)
clase_de_vuelo          string                  Ej: Regular, Vuelo Privado
clasificacion_de_vuelo  string                  Ej: Domestico, Internacional (Ojo: puede tener tildes)
tipo_de_movimiento      string                  Despegue o Aterrizaje
aeropuerto              string                  Codigo IATA/OACI del aeropuerto
origen_destino          string                  Aeropuerto de origen o destino
aerolinea_nombre        string                  Nombre de la aerolinea
aeronave                string                  Modelo o matricula
pasajeros               int

# Detailed Table Information
Database                aeropuertos_db
Table                   vuelos
Owner                   hadoop
Created Time            Mon Dec 01 13:25:18 ART 2025
Last Access             UNKNOWN
Created By              Spark 3.2.0
Type                    MANAGED
Provider                parquet
Table Properties        [data_owner=Data Team, source=Ministerio Transporte]
Statistics              3310998 bytes
Location                hdfs://172.17.0.2:9000/user/hive/warehouse/aeropuertos_db.db/vuelos
Serde Library           org.apache.hadoop.hive.ql.io.parquet.serde.ParquetHiveSerDe
InputFormat             org.apache.hadoop.hive.ql.io.parquet.MapredParquetInputFormat
OutputFormat            org.apache.hadoop.hive.ql.io.parquet.MapredParquetOutputFormat
Time taken: 0.154 seconds, Fetched 26 row(s)
spark-sql> SELECT fecha, aerolinea_nombre, pasajeros FROM aeropuertos_db.vuelos LIMIT 5;
2021-09-19      0       0
2021-09-19      JETSMART AIRLINES S.A.  60
2021-09-19      AEROLINEAS ARGENTINAS SA        63
2021-09-19      AEROLINEAS ARGENTINAS SA        75
2021-09-19      0       0
Time taken: 1.127 seconds, Fetched 5 row(s)
spark-sql>
```



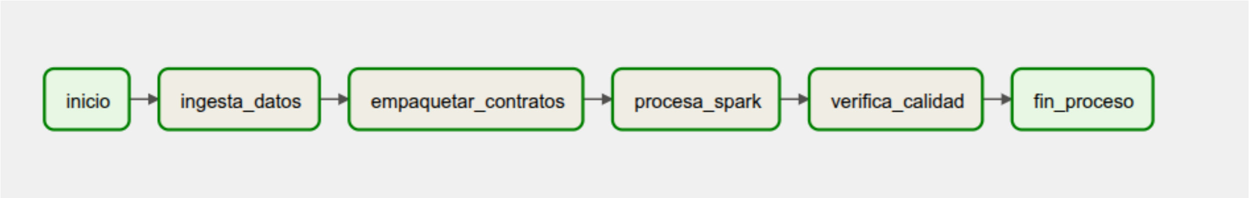
```
hadoop@88e4c6167f0c:~/airflow/dags$ chmod +x /home/hadoop/airflow/dags/proceso_aeropuertos_etl.py
hadoop@88e4c6167f0c:~/airflow/dags$
```



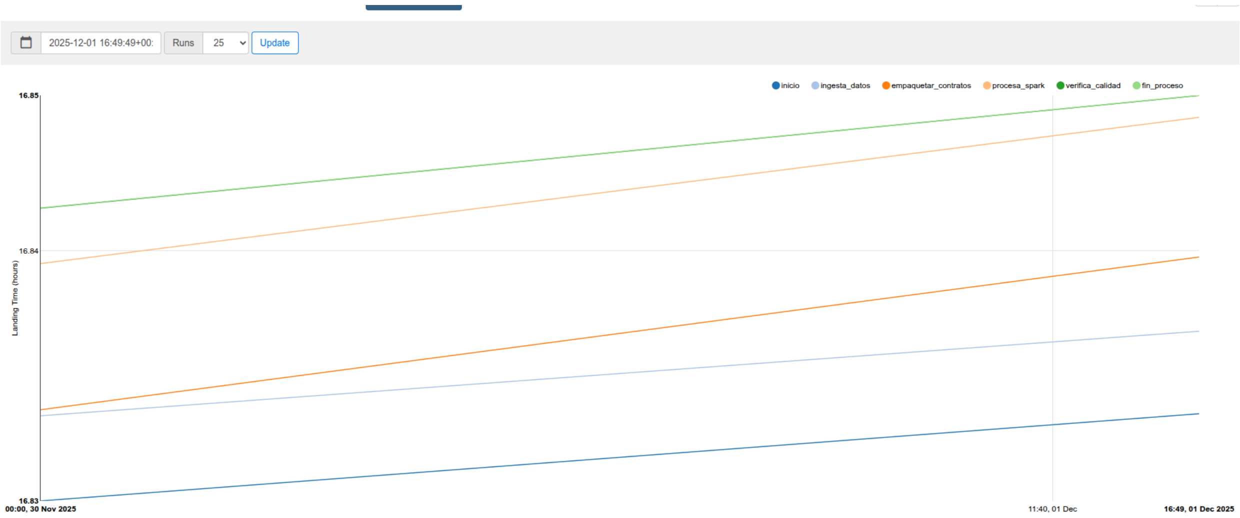
DAG
proceso_aeropuertos_etl_v2

DAG Details

DAG Runs Summary	
Total Runs Displayed	2
🟩 Total success	2
First Run Start	2025-12-01, 16:49:46 UTC
Last Run Start	2025-12-01, 16:49:59 UTC
Max Run Duration	00:00:47
Mean Run Duration	00:00:45
Min Run Duration	00:00:43
DAG Summary	
Total Tasks	6
BashOperators	4
DummyOperators	2



inicio
ingesta_datos
empaquetar_contratos
procesa_spark
verifica_calidad
fin_proceso


2025-12-01 16:49:49+00:
Runs
Update
inicio
ingesta_datos
empaquetar_contratos
procesa_spark
verifica_calidad
fin_proceso
Landing Time (hours)
00:00, 30 Nov 2025
11:40, 01 Dec
16:49, 01 Dec 2025