Project Report – Flight Delay Dataset Processing

Week 1: Project Initialization and Dataset Setup

Objectives

- Define project goals and key performance indicators (KPIs).
- Load airline flight delay data into Python for analysis.
- Explore schema, data types, size, and missing values.
- Perform sampling to inspect representative subsets.
- Optimize memory usage for faster processing.

Work Completed

1. **Data Loading**
    - Used pandas to read the Flight_delay.csv dataset:
    - import pandas as pd
    - import numpy as np
    - df = pd.read_csv("/Volumes/workspace/default/airlines/Flight_delay.csv")
    - Confirmed successful loading of dataset into a DataFrame (df).
2. **Schema & Data Types**
    - Checked column data types using df.dtypes.
    - Identified columns as object (strings), int, or float.
3. **Missing Values**
    - Used df.isnull().sum() to count null entries per column.
    - This step highlighted columns that required cleaning later.
4. **Dataset Size**
    - Checked shape using df.shape, giving number of rows and columns.
5. **Data Sampling**
    - Viewed first few records using df.head().
    - Generated random samples with df.sample(5).
    - Extracted fractional sample (10% of data) using frac=0.1.
6. **Memory Optimization**

- Implemented a function to optimize memory usage:
  - Converted string/object columns with limited unique values into category.
  - Downcasted integer and float columns to smaller types (int32, float32).
- Compared memory usage **before and after optimization**.
- Observed significant reduction in memory footprint, making the dataset easier to handle.

Week 2: Preprocessing and Feature Engineering

Objectives

- Handle missing/null values in delay and cancellation columns.
- Standardize and format date/time columns.
- Generate useful derived features for analysis.
- Save the cleaned dataset for reuse.

Work Completed

1. **Null Handling**
   - Targeted delay-related columns (ArrDelay, DepDelay, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay, LateAircraftDelay, Cancelled).
   - Replaced NaN values with 0, since missing delays imply no delay recorded.
2. **Date Formatting**
   - Converted the Date column to proper datetime format using pd.to_datetime.
   - Handled parsing errors with errors="coerce".
3. **Feature Engineering**
   - Extracted **Month** (df['Date'].dt.month) and **DayOfWeek** (df['Date'].dt.dayofweek).
     - DayOfWeek: Monday=0, ..., Sunday=6.
   - Processed **DepTime** (departure time stored as HHMM):

        - Converted to numeric and extracted **Hour** using integer division (DepTime // 100).
    - Created **Route** feature: Origin-Dest
4. **Verification**
    - Printed transformed columns step-by-step to validate:
        - Cleaned delay values.
        - Converted Date column.
        - Extracted Month, DayOfWeek, Hour.
        - Verified Route creation.
5. **Saving Preprocessed Data**
    - Final dataset was saved in a clean format for future analysis:
    - df.to_csv("/dbfs/FileStore/flights_preprocessed.csv", index=False)
    - This ensures reusability without repeating preprocessing steps.