**✈️ Module 3: Route & Delay Analysis – Full Python Code**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

🔧 Step 1: Load the dataset

df = pd.read\_csv("air\_arabia\_flight\_operations\_data.csv")

🧩 Step 1.1: Add missing columns if not present

if 'HoldingTime\_Min' not in df.columns:

df['HoldingTime\_Min'] = np.random.randint(0, 21, size=len(df)) # Simulate holding time

if 'Diversion' not in df.columns:

df['Diversion'] = np.where(df['DelayMinutes'] > 30, 'Yes', 'No') # Simulate diversions

if 'DepartureAirport' not in df.columns:

airport\_list = ['SHJ', 'DXB', 'AUH', 'MCT']

df['DepartureAirport'] = np.random.choice(airport\_list, size=len(df)) # Simulate airports

✅ Preview key columns

print(df[['Route', 'ActualFuel\_kg', 'PlannedFuel\_kg', 'DelayMinutes',

'TaxiOutTime\_min', 'HoldingTime\_Min', 'Diversion', 'DepartureAirport']].head())

🔍 Step 2: Calculate Fuel Wastage

df['FuelWastage\_kg'] = df['ActualFuel\_kg'] - df['PlannedFuel\_kg']

df['FuelWastage\_kg'] = df['FuelWastage\_kg'].apply(lambda x: x if x > 0 else 0)

print("Average Fuel Wastage:", df['FuelWastage\_kg'].mean())

📊 Step 3: Analyze Fuel Wastage vs Delay/Holding/Diversion

# Delay vs Fuel Wastage

plt.figure(figsize=(8,6))

sns.scatterplot(data=df, x='DelayMinutes', y='FuelWastage\_kg', alpha=0.6)

plt.title('Delay vs Fuel Wastage')

plt.xlabel('Delay (minutes)')

plt.ylabel('Fuel Wasted (kg)')

plt.tight\_layout()

plt.show()

# Holding Time vs Fuel Wastage

plt.figure(figsize=(8,6))

sns.scatterplot(data=df, x='HoldingTime\_Min', y='FuelWastage\_kg', alpha=0.6)

plt.title('Holding Time vs Fuel Wastage')

plt.xlabel('Holding Time (min)')

plt.ylabel('Fuel Wasted (kg)')

plt.tight\_layout()

plt.show()

# Diversion impact

diversion\_fuel = df[df['Diversion'] == 'Yes']['FuelWastage\_kg']

print("Avg fuel wasted in diversions:", diversion\_fuel.mean())

🚕 Step 4: Taxi-out Time vs Fuel Burn

plt.figure(figsize=(8,6))

sns.scatterplot(data=df, x='TaxiOutTime\_min', y='ActualFuel\_kg', hue='AircraftType', alpha=0.6)

plt.title('Taxi-out Time vs Fuel Burn')

plt.xlabel('Taxi-out Time (min)')

plt.ylabel('Fuel Burn (kg)')

plt.tight\_layout()

plt.show()

📍 Step 5: Routes with Highest Fuel Wastage

route\_waste = df.groupby('Route')['FuelWastage\_kg'].mean().reset\_index().sort\_values(by='FuelWastage\_kg', ascending=False)

plt.figure(figsize=(10,6))

sns.barplot(data=route\_waste.head(10), x='FuelWastage\_kg', y='Route', palette='Reds')

plt.title('Top 10 Routes by Average Fuel Wastage')

plt.xlabel('Avg Fuel Wasted (kg)')

plt.ylabel('Route')

plt.tight\_layout()

plt.show()

🛫 Step 6: Airports Contributing to Fuel Inefficiency

airport\_waste = df.groupby('DepartureAirport')['FuelWastage\_kg'].mean().reset\_index().sort\_values(by='FuelWastage\_kg', ascending=False)

plt.figure(figsize=(10,6))

sns.barplot(data=airport\_waste.head(10), x='FuelWastage\_kg', y='DepartureAirport', palette='Oranges')

plt.title('Top Airports by Avg Fuel Wastage')

plt.xlabel('Avg Fuel Wasted (kg)')

plt.ylabel('Departure Airport')

plt.tight\_layout()

plt.show()