**Introduction**

The objective of this project is to analyze flight booking data from Easemytrip, focusing on routes between India's top six metro cities. This analysis aims to uncover patterns in flight prices, durations, and airline frequencies, providing insights for both consumers and airlines.

**Data Overview**

* **Dataset**: The analysis utilizes a dataset containing 300,261 entries and 11 features, sourced from Clean\_Dataset.csv.
* **Features**: Key features include airline, source city, destination city, departure time, arrival time, price, and duration

## Methodology

We utilized several key libraries for this analysis:

**Pandas** for data manipulation,

import pandas as pd

**NumPy** for numerical operations, and

import numpy as np

**Matplotlib** along with **Seaborn** for data visualization.

import matplotlib.pyplot as plt

import seaborn as sns

These tools allowed us to effectively analyze and visualize flight booking data.

### Data Loading

The dataset was loaded into a Pandas DataFrame for analysis .

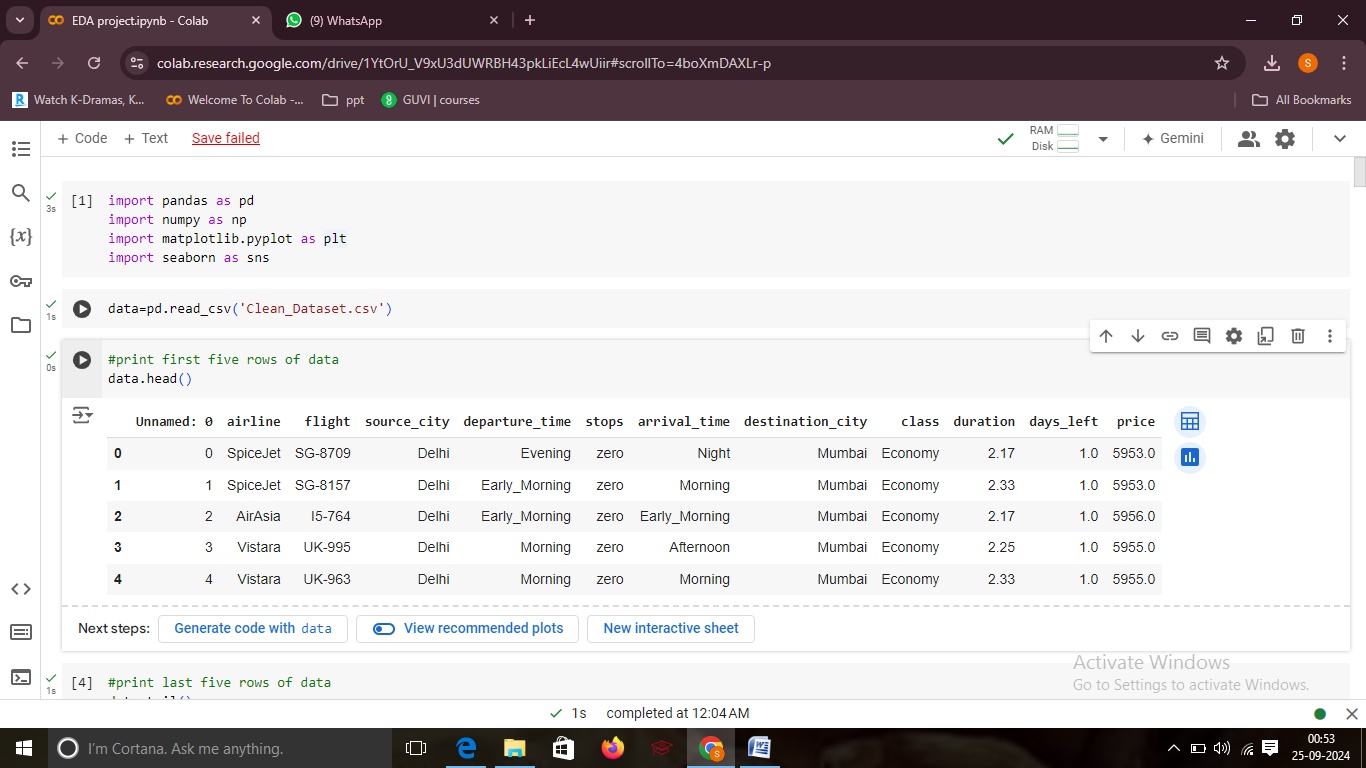
data=pd.read\_csv('Clean\_Dataset.csv')

### Initial Data Inspection

The first five rows were reviewed to understand the dataset's structure.

#print first five rows of data

data.head()



### Initial Data Inspection

The last five rows were reviewed to understand the dataset's structure.

#print last five rows of data

data.tail()

### eda photo flight2.jpg

### Data Cleaning

**Missing Values**: Checked for null values and removed any rows with missing data.

#Cleaning the data for missing values, null values

data.isnull().sum()

data.dropna(inplace=True)

**Index Column**:

Removed the 'index' column if it was present.

#Remove IN-DATA 'index' column

if 'index' in data.columns:

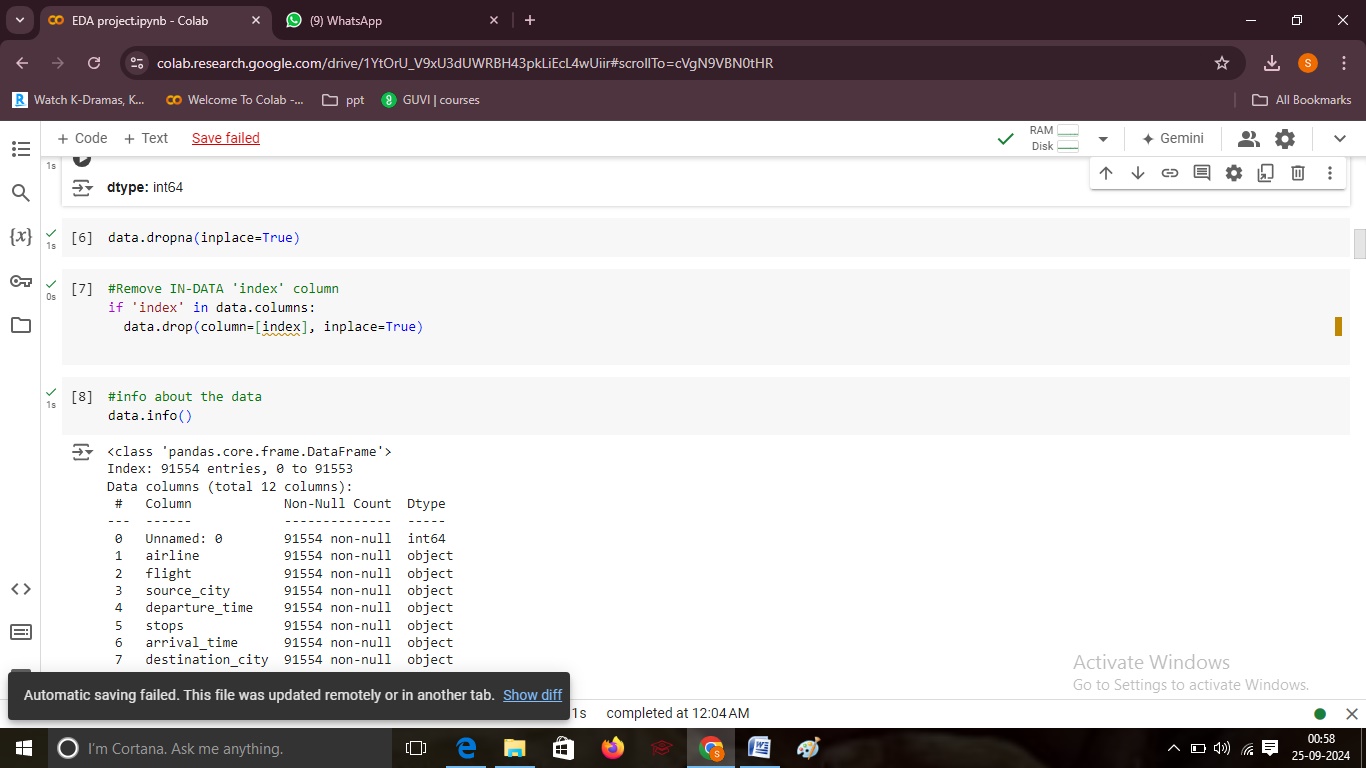
  data.drop(column=[index], inplace=True)

### Dataset Insights

Used info() and describe() to gather information about the data types and statistical properties.

#info about the data

data.info()



#descripition about the data

data.describe()

## eda photo flight5.jpg

## Visualizations

### Airline Frequency

A count plot visualized the distribution of flights across different airlines.

#VISUALIZATION

#1.Airline in the dataset

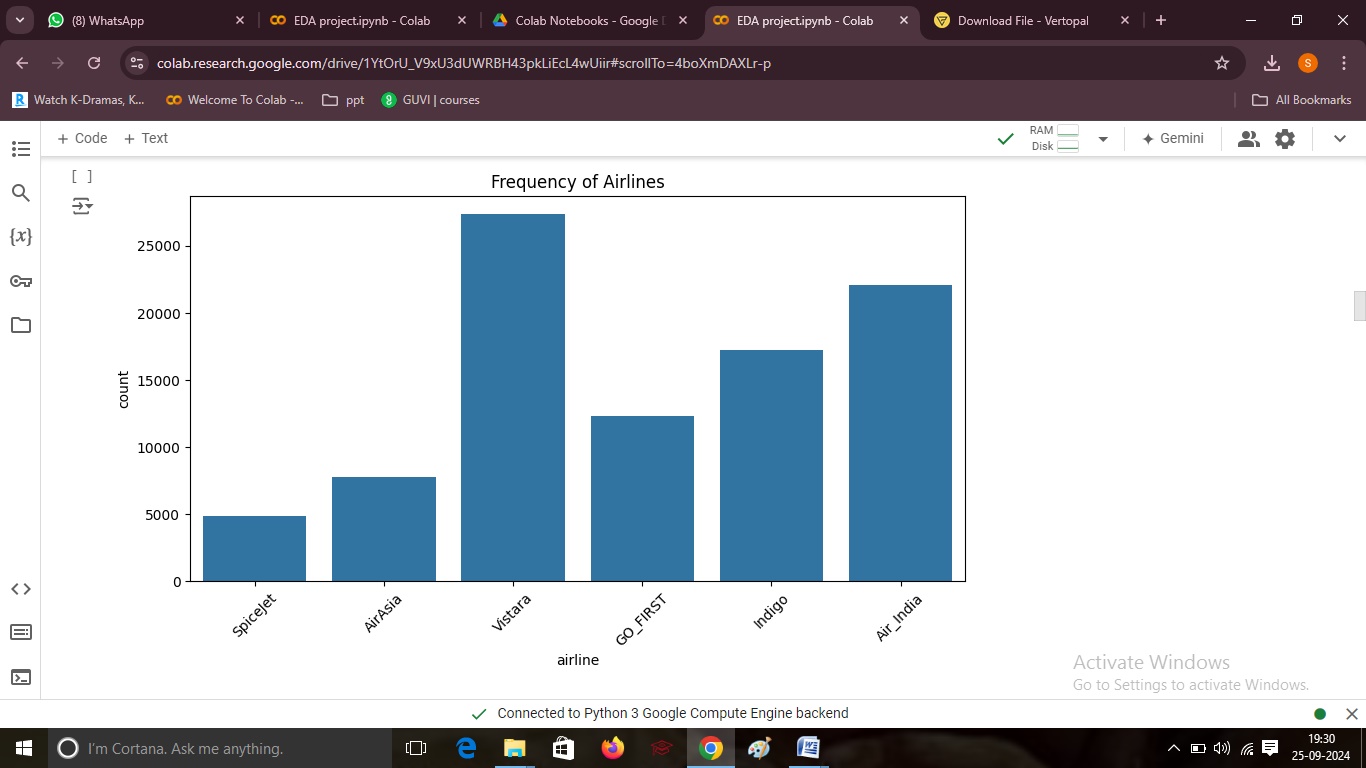
plt.figure(figsize=(10, 5))

sns.countplot(data=data, x='airline')

plt.title('Frequency of Airlines')

plt.xticks(rotation=45)

plt.show()



### Departure vs. Arrival Time

A bar plot displayed the relationship between departure and arrival times.

#2.Departure Time Against Arrival Time

plt.figure(figsize=(10, 5))

sns.barplot(data=data, x='departure\_time', y='arrival\_time',color='red')

plt.title('Departure Time vs Arrival Time')

plt.xticks(rotation=45)

plt.show()



### Source vs. Destination Cities

A count plot compared the source cities to destination cities, revealing popular routes.

#3.Source City Against Destination City

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

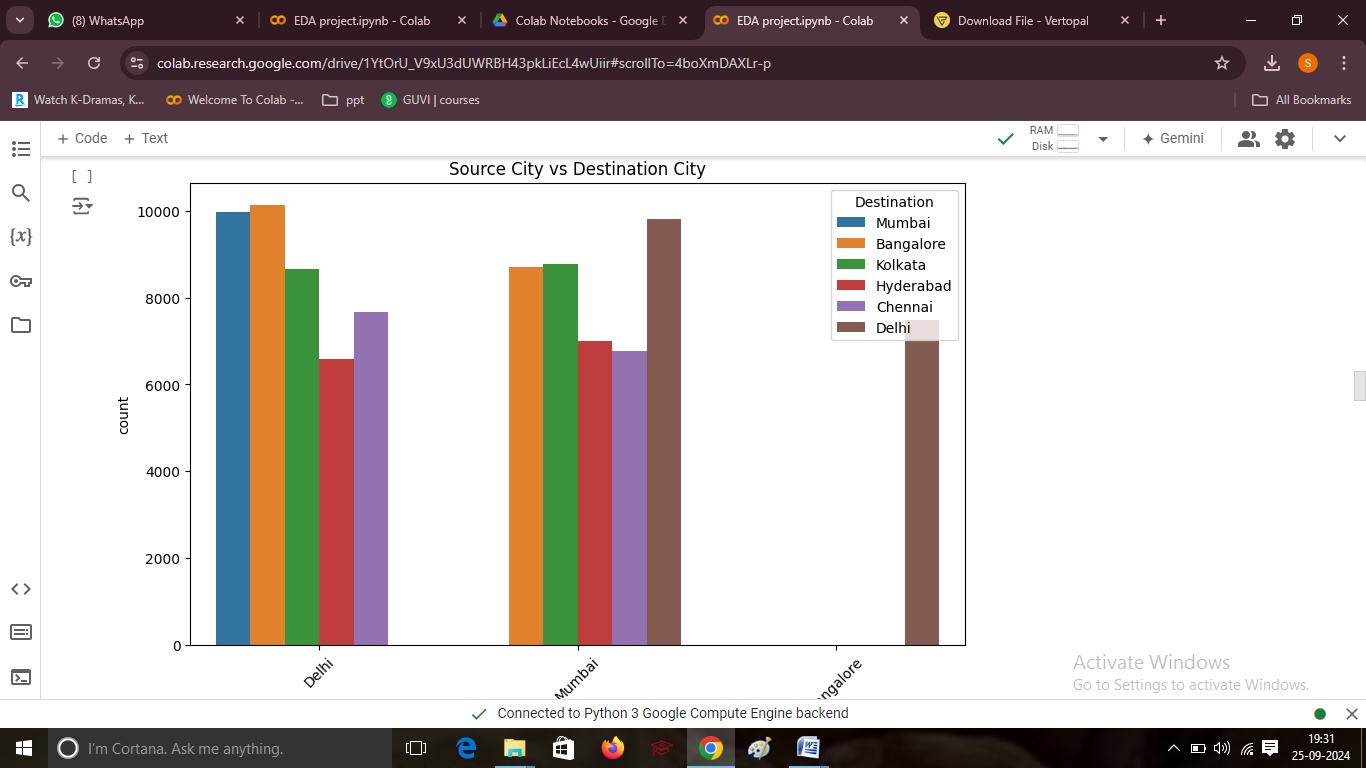
sns.countplot(data=data, x='source\_city', hue='destination\_city')

plt.title('Source City vs Destination City')

plt.xticks(rotation=45)

plt.legend(title='Destination')

plt.show()



### Price Variation by Airline

A box plot illustrated how ticket prices varied across different airlines.

#4.Price Variation with Airlines

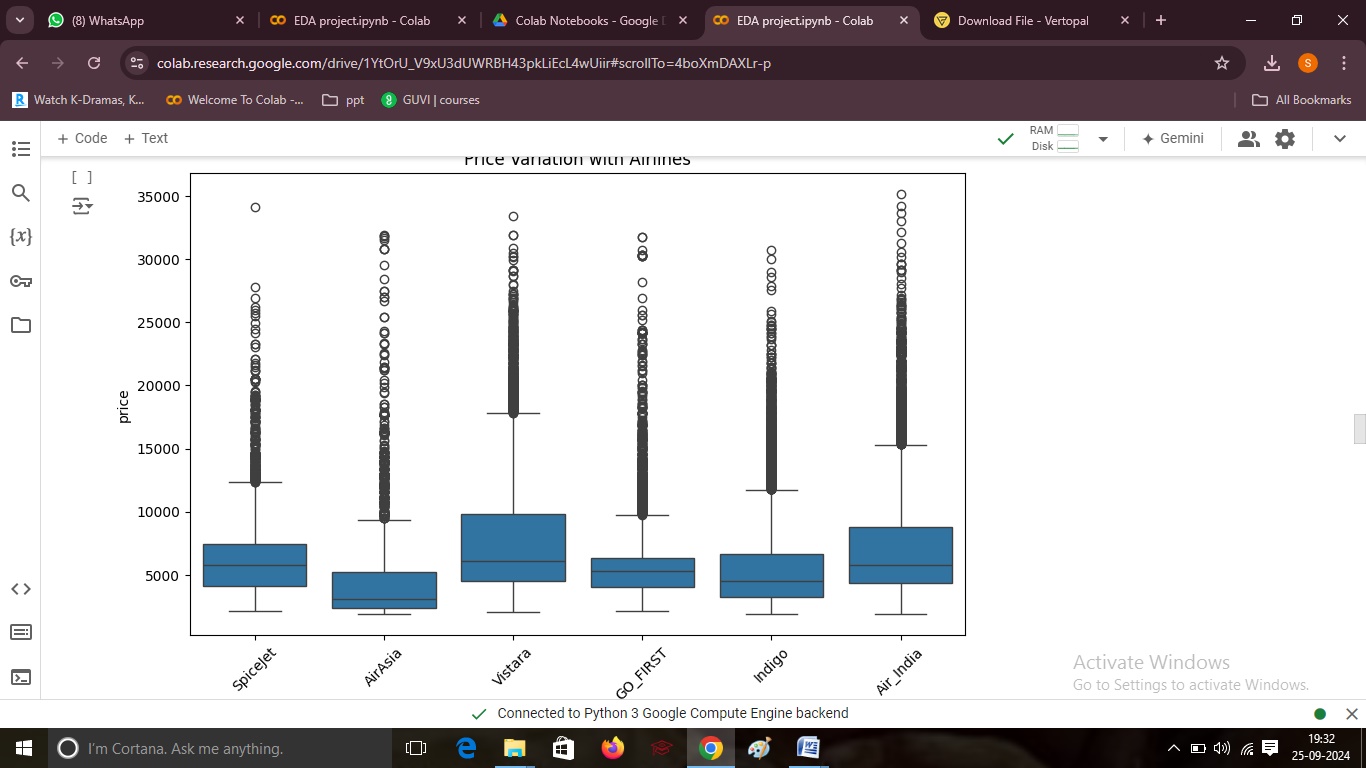
plt.figure(figsize=(10, 6))

sns.boxplot(data=data, x='airline', y='price')

plt.title('Price Variation with Airlines')

plt.xticks(rotation=45)

plt.show()



### Ticket Price Trends

Line plots explored the relationship between ticket prices, departure time, and arrival time.

#5.Ticket Price vs Departure and Arrival Time

plt.figure(figsize=(12, 6))

sns.lineplot(data=data, x='departure\_time', y='price', label='departure\_price', color='blue')

sns.lineplot(data=data, x='arrival\_time', y='price', label='arrival\_price', color='orange')

plt.title('Ticket Price vs Departure and Arrival Time')

plt.xticks(rotation=45)

plt.legend()

plt.show()



### Price Changes with Source and Destination

A box plot visualized price changes based on source and destination cities.\

#6.Price changes with Source and Destination

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

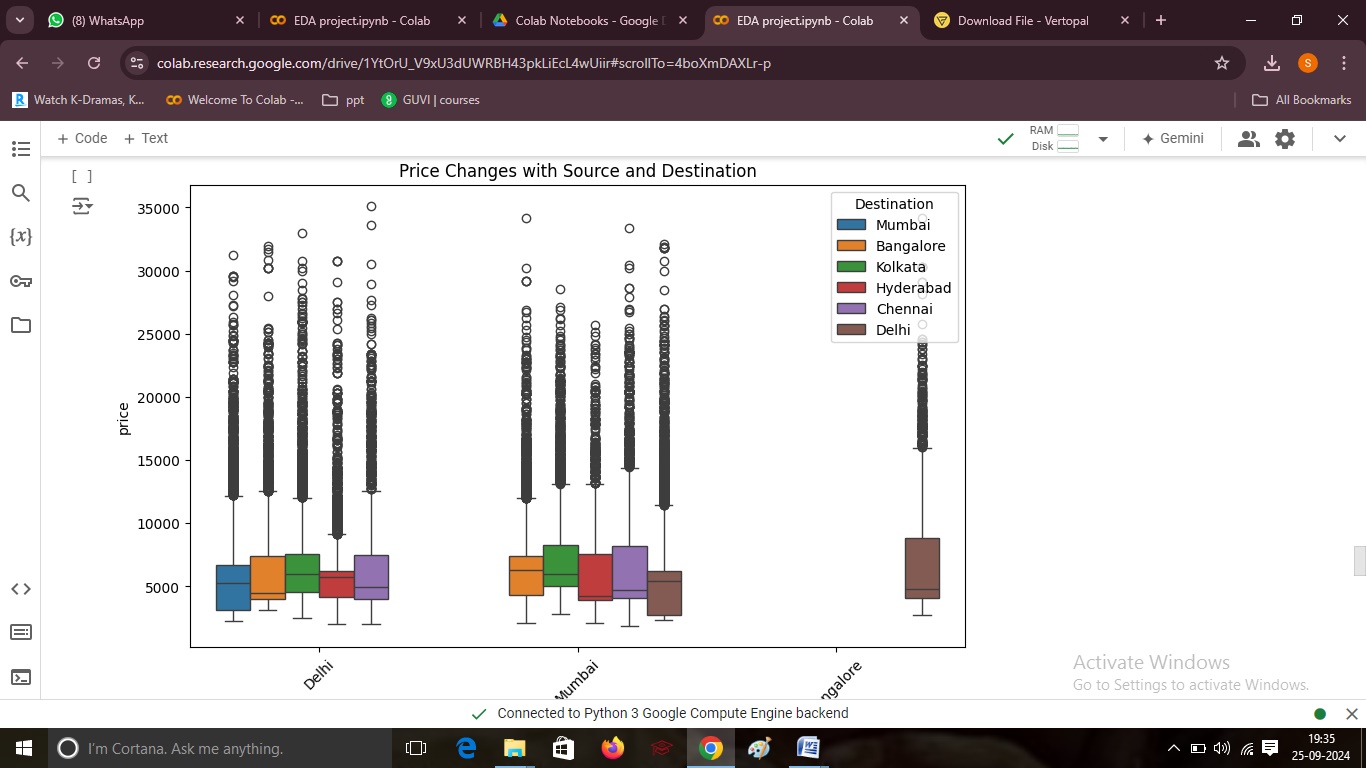
sns.boxplot(data=data, x='source\_city', y='price', hue='destination\_city')

plt.title('Price Changes with Source and Destination')

plt.xticks(rotation=45)

plt.legend(title='Destination')

plt.show()



### Duration of Travel vs. Source City

A box plot represented travel durations for different source cities.

#7.Duration of travel vs city

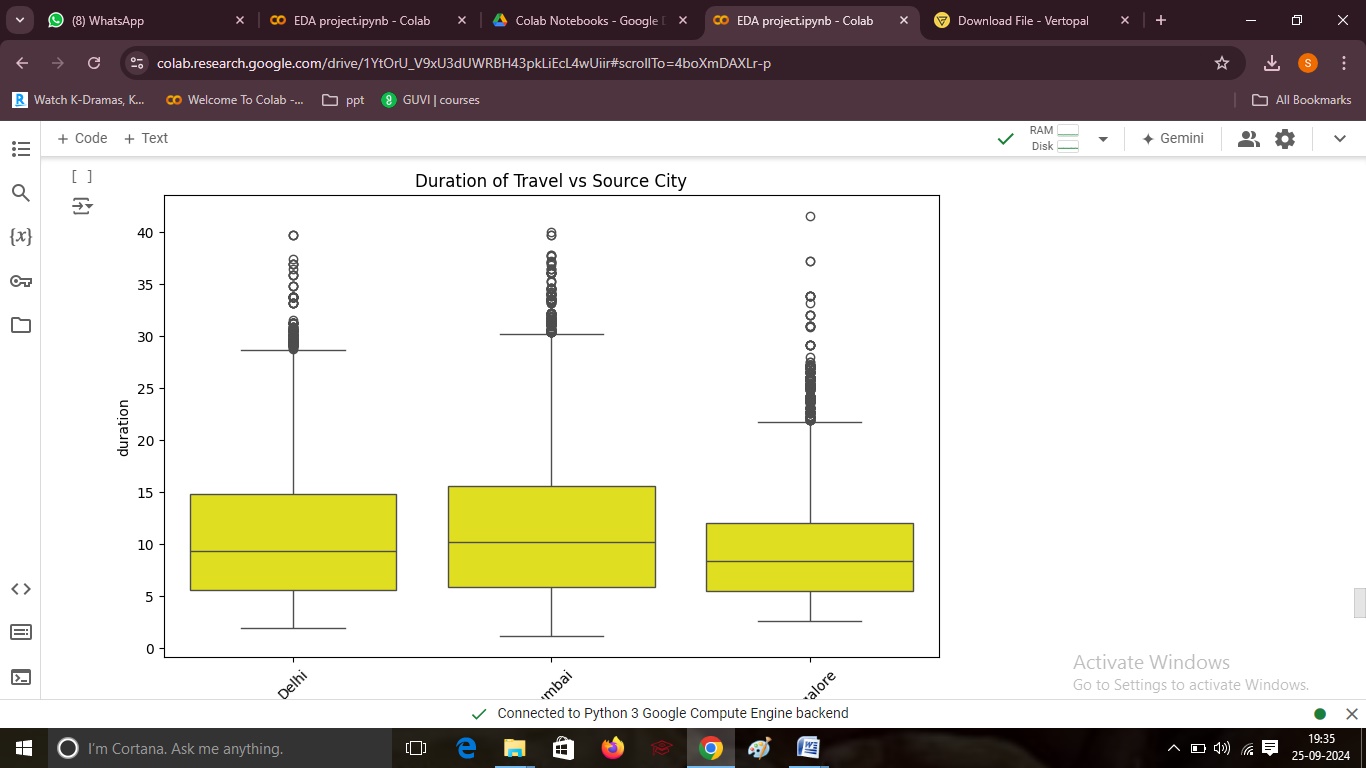
plt.figure(figsize=(10, 6))

sns.boxplot(data=data, x='source\_city', y='duration',color='yellow')

plt.title('Duration of Travel vs Source City')

plt.xticks(rotation=45)

plt.show()



**High price with class type for city.**

The heatmap illustrates average flight prices by source city and class type, highlighting pricing variations across different classes.

#8.High price with class type for city

average\_price = data.groupby(['source\_city', 'class'])['price'].mean().unstack()

plt.figure(figsize=(12, 8))

sns.heatmap(average\_price, annot=True, fmt=".2f", cmap='YlGnBu', linewidths=0.5)

plt.title('Heatmap of Average Prices by Class Type for Each City')

plt.xlabel('Class Type')

plt.ylabel('Source City')

plt.show()

