**1. Title Page**

**Title:** *Analyzing Website Traffic Data*  
**Report Submitted By:** Siddhant Tiwari  
**Institution : KIET Group Of Instituion**  
**Course Name:** Introduction to AI  
**Date:** 10/03/2025

**2. Introduction**

In this report, we analyze the traffic data of a website to gain insights into user behavior, traffic trends, and site performance. The objective is to examine key metrics such as page views, user sessions, bounce rate, and user demographics. Analyzing website traffic helps identify areas for improvement and allows businesses or content creators to make data-driven decisions to enhance user experience, increase engagement, and optimize marketing efforts.

**3. Methodology**

1. **Data Collection:** Website traffic data was collected using tools like Google Analytics.
2. **Data Preprocessing:** The data was cleaned, missing values were handled, and time intervals were applied.
3. **Analysis:** Key metrics (e.g., page views, sessions, traffic sources) were analyzed.
4. **Visualization:** Trends were visualized using graphs (line charts, bar charts) to identify patterns.

**4 . code**

# Step 1: Import necessary libraries

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# Step 2: Load the CSV file into a DataFrame

from google.colab import files

uploaded = files.upload()

# Assuming the file name is 'website\_wata.csv'

file\_name = 'website\_wata.csv'

df = pd.read\_csv(file\_name)

# Step 3: Display the first few rows of the DataFrame

print(df.head())

# Step 4: Basic Data Analysis

# Summary statistics

print(df.describe())

# Check for missing values

print(df.isnull().sum())

# Step 5: Data Visualization

# Histogram of Page Views

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

sns.histplot(df['Page Views'], bins=30, kde=True)

plt.title('Distribution of Page Views')

plt.xlabel('Page Views')

plt.ylabel('Frequency')

plt.show()

# Boxplot of Session Duration by Traffic Source

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

sns.boxplot(x='Traffic Source', y='Session Duration', data=df)

plt.title('Session Duration by Traffic Source')

plt.xlabel('Traffic Source')

plt.ylabel('Session Duration')

plt.show()

# Scatter plot of Time on Page vs Conversion Rate

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

sns.scatterplot(x='Time on Page', y='Conversion Rate', data=df)

plt.title('Time on Page vs Conversion Rate')

plt.xlabel('Time on Page')

plt.ylabel('Conversion Rate')

plt.show()

# Correlation Heatmap

plt.figure(figsize=(10, 8))

# Select only numerical features for correlation calculation

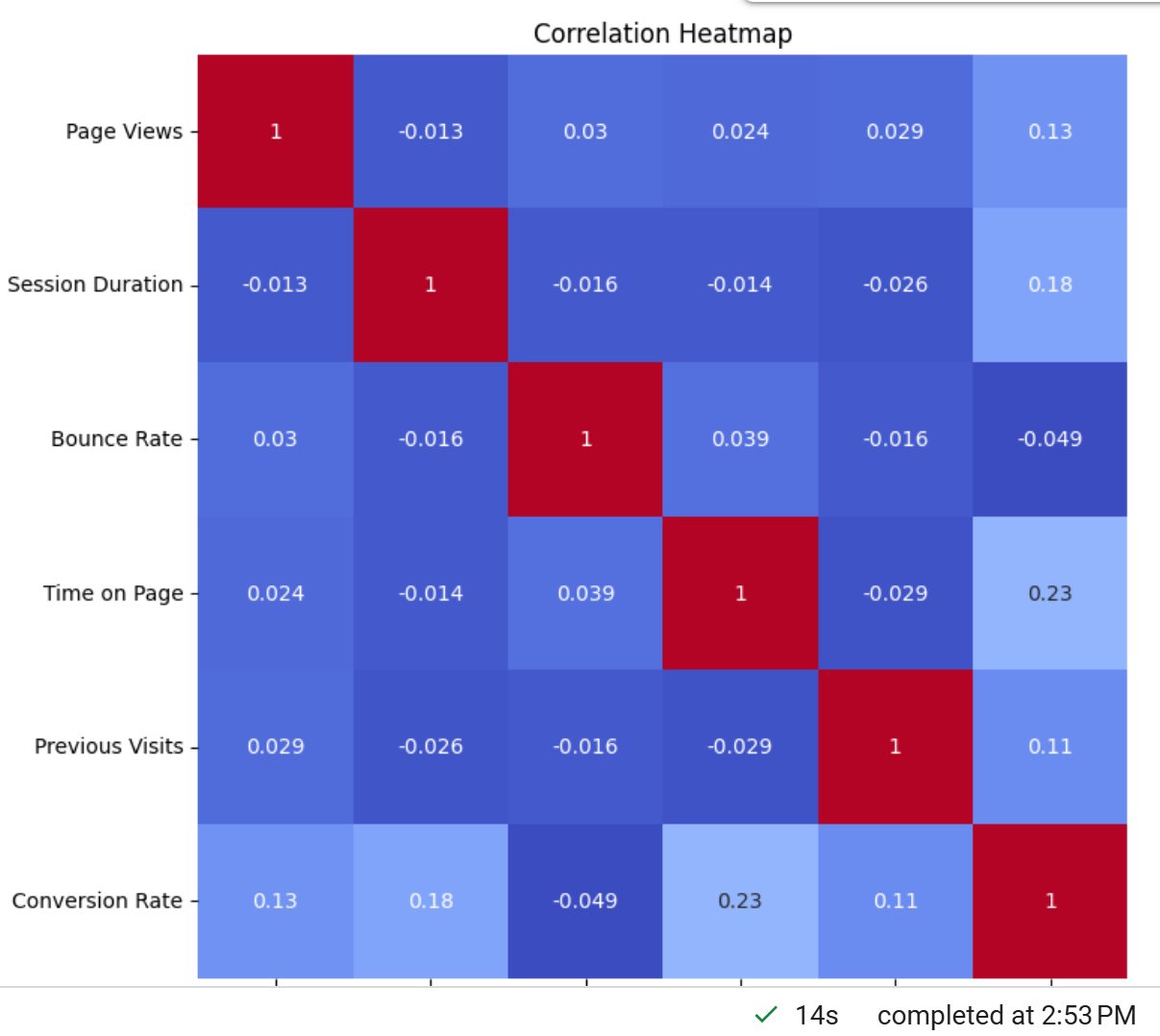
numerical\_df = df.select\_dtypes(include=['number'])

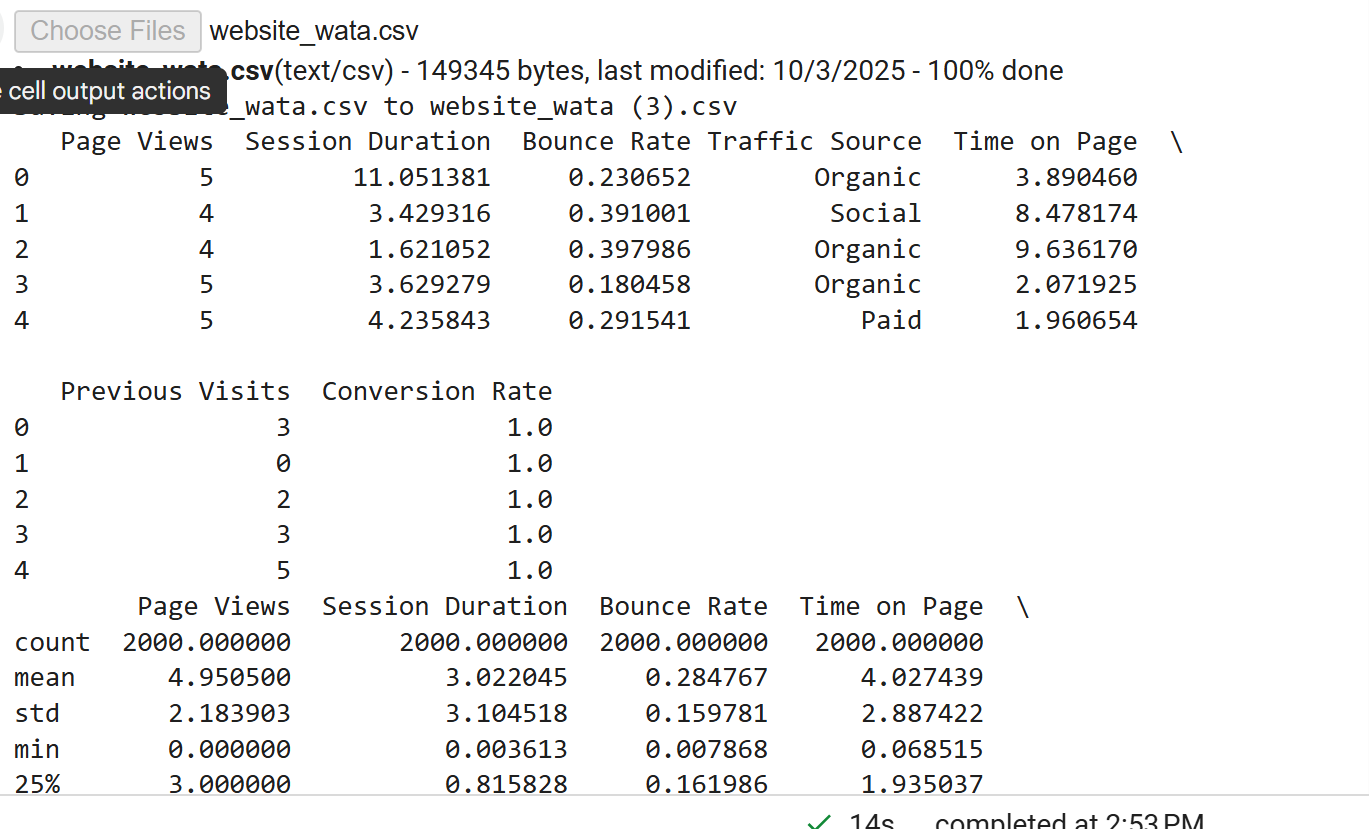
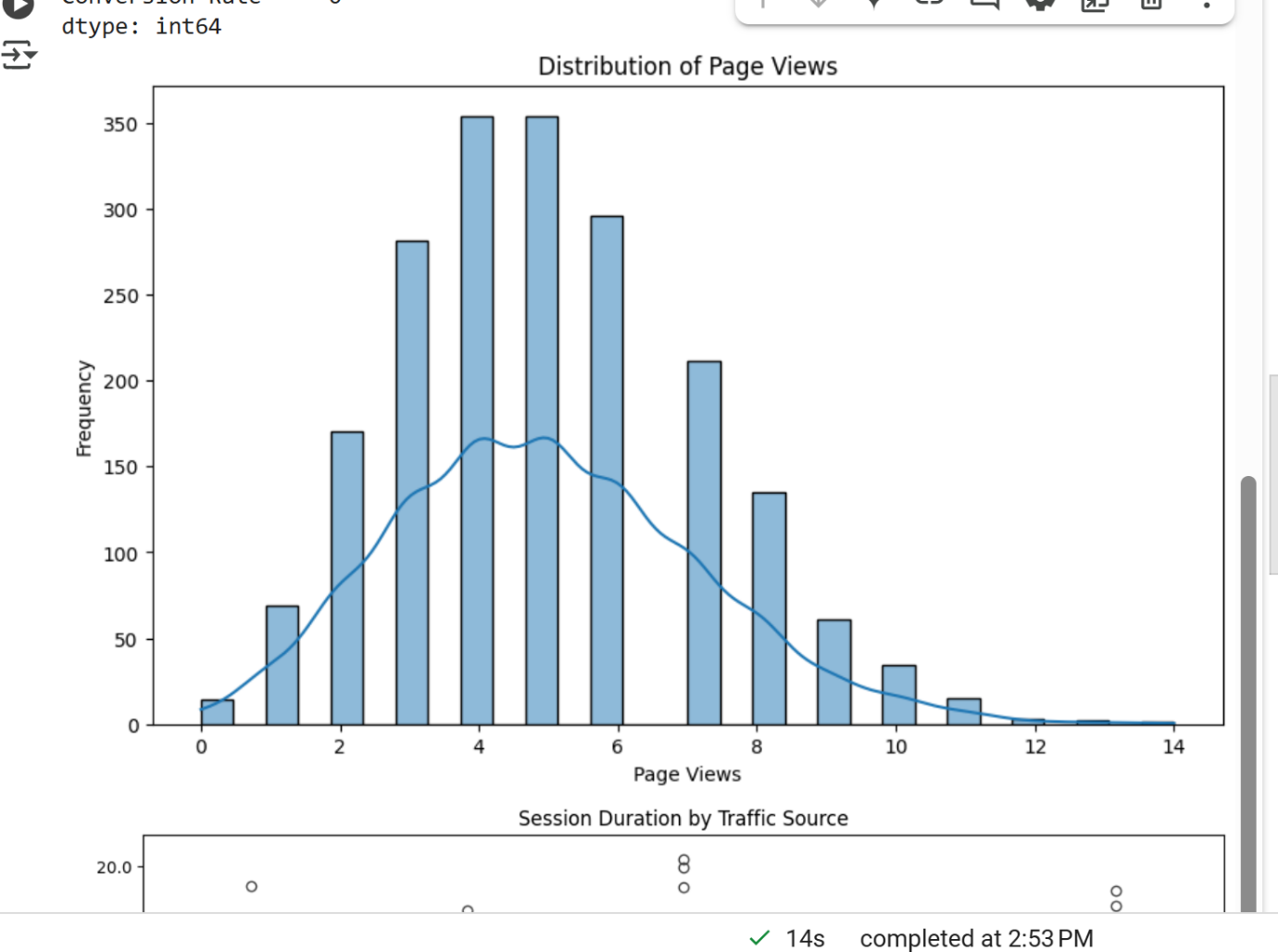
corr = numerical\_df.corr()

sns.heatmap(corr, annot=True, cmap='coolwarm')

plt.title('Correlation Heatmap')

plt.show()

1. **Screen short**

****