```
import pandas as pd
import matplotlib.pyplot as plt
```

```
df = pd.read_csv("delhi_metro_updated.csv")
df
```

	TripID	Date	From_Station	To_Station	Distance_km	Fare	Cost_per_passenger	Pa
0	59771	2022-05-08	Inderlok	Kashmere Gate	12.94	77.99	18.27	
1	21363	2023-01-12	Model Town	Dilshad Garden	2.33	35.89	83.71	
2	127325	2023-07-13	Kashmere Gate	Netaji Subhash Place	5.56	64.35	43.70	
3	140510	2022-11-10	Chandni Chowk	Hauz Khas	4.02	144.13	14.98	
4	144298	2022-11-06	Rajiv Chowk	Kalkaji Mandir	9.66	104.96	83.84	
...	...	...	...	...	...	...	...	
36459	100775	2022-01-08	Hauz Khas	Kalkaji Mandir	4.91	112.86	60.10	
36460	30493	2023-04-27	Jasola Vihar	Model Town	4.26	77.78	63.43	
36461	102992	2024-04-16	Laxmi Nagar	Mandi House	10.51	193.51	85.12	
36462	30959	2024-01-08	inderlok	Janakpuri West	0.95	180.59	32.69	
36463	62603	2023-11-22	Rajouri Gar	NaN	NaN	NaN	NaN	

36464 rows × 10 columns

Next steps: Generate code with df New interactive sheet

```
df = df.copy()
```

```
df['Date'] = pd.to_datetime(df['Date'])
```

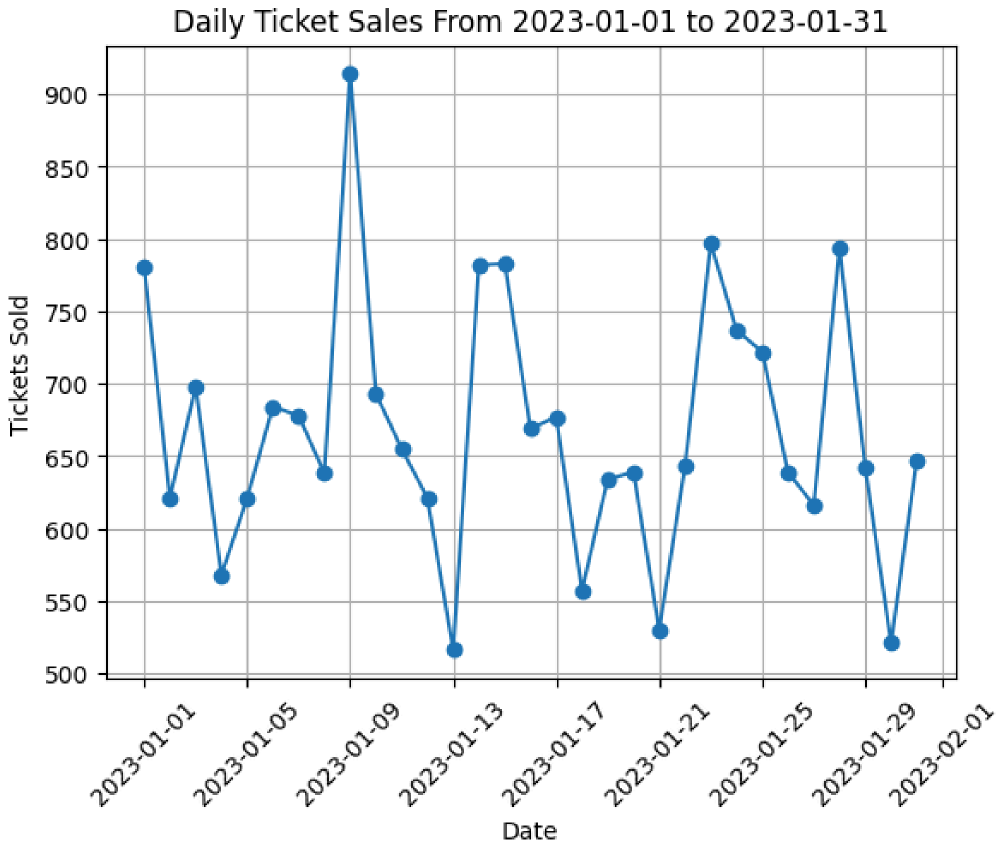
```
start_date = "2023-01-01"
end_date = "2023-01-31"

df = df[(df['Date'] >= start_date) & (df['Date'] <= end_date)]
```

```
df['Route'] = df['From_Station'] + " → " + df['To_Station']
```

```
daily_sales = df.groupby('Date')['Passengers'].sum()

plt.figure()
plt.plot(daily_sales, marker='o')
plt.xlabel("Date")
plt.ylabel("Tickets Sold")
plt.title("Daily Ticket Sales From "+ start_date+" to "+ end_date)
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```



```
ticket_stats = df['Passengers'].describe()
ticket_stats
```

	Passengers
count	1026.000000
mean	20.192008
std	4.484246
min	6.000000
25%	17.000000
50%	20.000000
75%	23.000000
max	38.000000

dtype: float64

```
route_stat_report = (
    df.groupby('Route')['Passengers']
    .agg(['sum', 'mean', 'max', 'min'])
    .reset_index()
)

route_stat_report.columns = [
    'Route',
    'Total_Tickets',
    'Average_Tickets',
    'Maximum_Tickets',
    'Minimum_Tickets'
]

route_stat_report
```

	Route	Total_Tickets	Average_Tickets	Maximum_Tickets	Minimum_Tickets
0	AIIMS → Netaji Subhash Place	22.0	22.000000	22.0	22.0
1	AIIMS → Old Delhi	12.0	12.000000	12.0	12.0
2	AIIMS → Punjabi Bagh	18.0	18.000000	18.0	18.0
3	AIIMS → Kashmere Gate	14.0	14.000000	14.0	14.0
4	AIIMS → Rajouri Garden	10.0	10.000000	10.0	10.0
...	...	...	...	...	...
541	Rajouri Garden → New Delhi	71.0	23.666667	27.0	18.0
542	Rajouri Garden → Noida City Centre	11.0	11.000000	11.0	11.0
543	Rajouri Garden → Pragati Maidan	24.0	12.000000	14.0	10.0
544	Rajouri Garden → Punjabi Bagh	58.0	19.333333	22.0	16.0
545	Rajouri Garden → Shivaji Park	34.0	17.000000	20.0	14.0

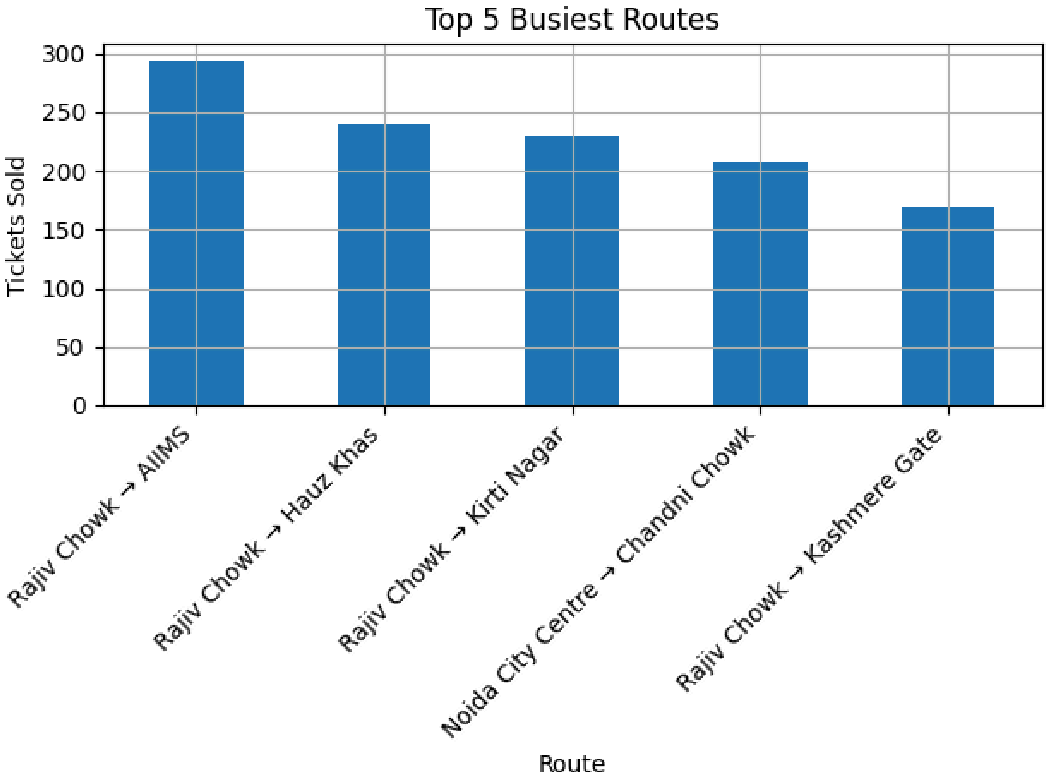
546 rows × 5 columns

Next steps: Generate code with `route_stat_report` | New interactive sheet

```
top_routes_sales = (
    df.groupby('Route')['Passengers']
    .sum()
    .sort_values(ascending=False)
    .head(5)
)

plt.figure()
top_routes_sales.plot(kind='bar')
plt.xlabel("Route")
plt.ylabel("Tickets Sold")
plt.title("Top 5 Busiest Routes")
plt.xticks(rotation=45, ha='right')
plt.grid(axis='x')
plt.grid(axis='y')
plt.tight_layout()

plt.show()
```



```
print("Total Tickets Sold:", df['Passengers'].sum())
```

```
Total Tickets Sold: 20717.0
```

```
df['Day'] = df['Date'].dt.day_name()

daywise_sales = df.groupby('Day')['Passengers'].sum()
order = ['Monday','Tuesday','Wednesday','Thursday','Friday','Saturday','Sunday']
daywise_sales = daywise_sales.reindex(order)

plt.figure()
plt.bar(daywise_sales.index, daywise_sales.values)
plt.xlabel("Day")
plt.ylabel("Tickets Sold")
plt.title("Day-wise Ticket Sales")
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.grid(axis='x')
plt.show()
```