

```
#Import Pandas
```

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

```
# Load dataset
```

```
data = pd.read_excel('Data Analyst Assignment (2).xlsx')
```

```
# Display the first few rows of the dataframe
```

```
print(data.head())
```

	AWB		Courier Zone	Charged wt	Courier Slab
Multiplier \					
0	96403	Delhivery Surface 10 Kgs	A	10.5000	10.0
1					
1	96404	Shipsy	A	0.5475	0.5
1					
2	96405	Blue Dart	D	0.0006	0.5
0					
3	96406	Delhivery Surface 10 Kgs	A	10.5000	10.0
1					
4	96407	Shipsy	A	0.3540	0.5
0					

	Direction
0	Forward
1	Forward
2	Forward
3	Forward
4	Forward

```
# Display the last few rows of the dataframe
```

```
print(data.tail())
```

	AWB		Courier Zone	Charged wt	Courier Slab
Multiplier \					
60	96463	Shipsy	A	0.4680	0.5
0					
61	96464	Delhivery Surface 5 Kgs	D	6.3000	5.0
2					
62	96465	Blue Dart	D	2.2875	0.5
4					
63	96466	Shipsy	A	5.1750	0.5
10					
64	96467	Delhivery Surface	A	0.7500	0.5
1					

	Direction
60	Forward
61	Forward

```

62 Forward
63 Forward
64 Forward

# Calculate shipping cost
data['shipping_cost'] = data['Courier Slab'] * data['Charged wt']

print(data.columns)

Index(['AWB', 'Courier', 'Zone', 'Charged wt', 'Courier Slab',
      'Multiplier',
      'Direction', 'shipping_cost'],
      dtype='object')

```

## Zonewise Average Shipping Cost

```

# Zone-wise average shipping cost
zonewise_avg_cost = data.groupby('Zone')
['shipping_cost'].mean().reset_index()

# Print the result
print(zonewise_avg_cost)

```

	Zone	shipping_cost
0	A	16.399511
1	B	33.381562
2	C	64.210000
3	D	42.239735
4	E	4.100000

## Create a Summary of the Data and Identify the Most Optimum Service Provider

```

# Summary for most optimum service provider
courier_zone_summary = data.groupby(['Courier', 'Zone'])
['shipping_cost'].sum().reset_index()
Most_optimum_couriers =
courier_zone_summary.loc[courier_zone_summary.groupby('Zone')
['shipping_cost'].idxmin()]

# Print the summary for reference
print(courier_zone_summary)

```

	Courier	Zone	shipping_cost
0	Blue Dart	A	1.268625
1	Blue Dart	B	0.300000

2	Blue Dart	C	1.260000
3	Blue Dart	D	4.616550
4	Delhivery Surface	A	1.072500
5	Delhivery Surface	B	0.052500
6	Delhivery Surface 10 Kgs	A	330.000000
7	Delhivery Surface 10 Kgs	B	213.000000
8	Delhivery Surface 10 Kgs	C	225.000000
9	Delhivery Surface 10 Kgs	D	300.000000
10	Delhivery Surface 2 Kgs	C	9.000000
11	Delhivery Surface 2 Kgs	D	12.000000
12	Delhivery Surface 5 Kgs	A	174.750000
13	Delhivery Surface 5 Kgs	B	31.500000
14	Delhivery Surface 5 Kgs	C	150.000000
15	Delhivery Surface 5 Kgs	D	232.500000
16	Shipsy	A	58.791750
17	XPRESSBEES	E	0.060000
18	Xpressbees 2kg	A	8.100000
19	Xpressbees 2kg	B	22.200000
20	Xpressbees 2kg	E	12.240000

```
# Print the results for the most optimum couriers
for index, row in Most_optimum_couriers.iterrows():
    print(f"The most optimum courier for Zone {row['Zone']} is:
{row['Courier']} with a total shipping cost of
{row['shipping_cost']:.2f}.")
```

The most optimum courier for Zone A is: Delhivery Surface with a total shipping cost of 1.07.

The most optimum courier for Zone B is: Delhivery Surface with a total shipping cost of 0.05.

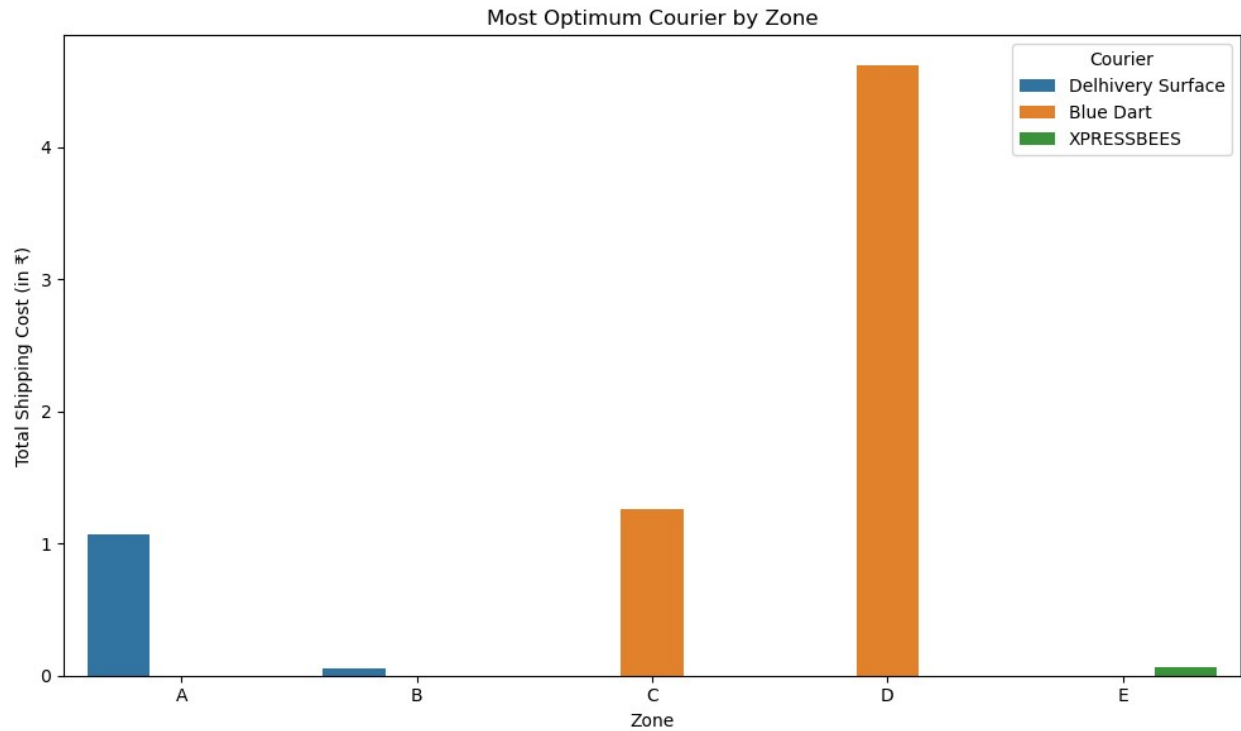
The most optimum courier for Zone C is: Blue Dart with a total shipping cost of 1.26.

The most optimum courier for Zone D is: Blue Dart with a total shipping cost of 4.62.

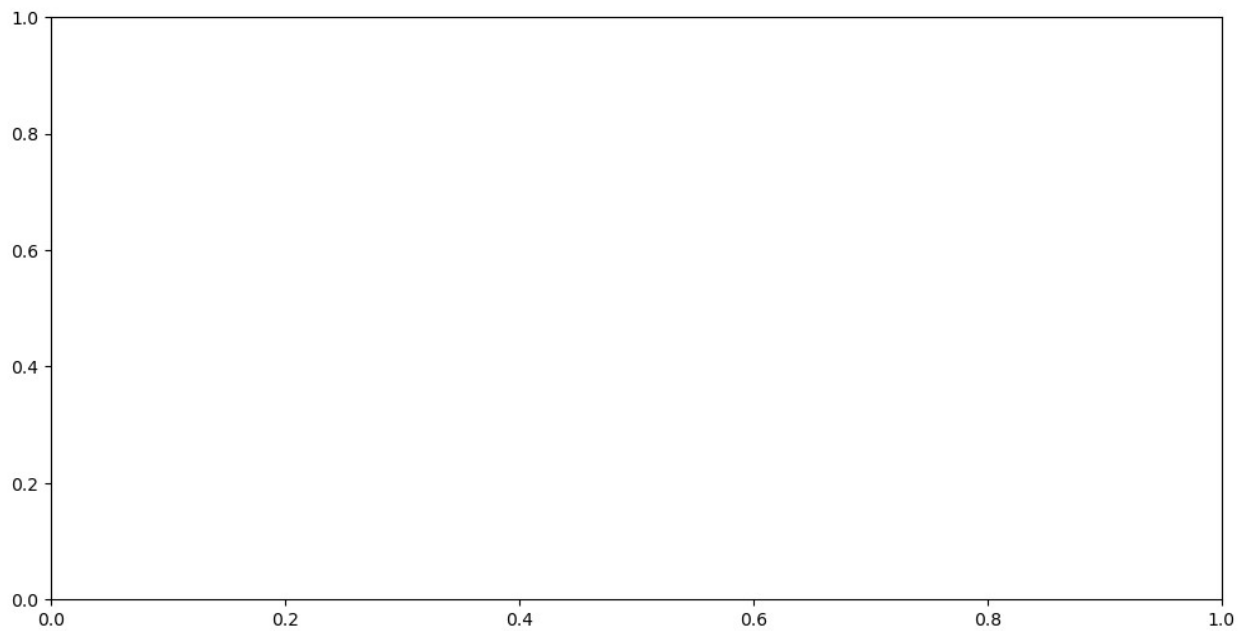
The most optimum courier for Zone E is: XPRESSBEES with a total shipping cost of 0.06.

```
# Plot data
```

```
plt.figure(figsize=(10, 6))
sns.barplot(data=optimum_couriers, x='Zone', y='shipping_cost',
hue='Courier')
plt.title('Most Optimum Courier by Zone')
plt.xlabel('Zone')
plt.ylabel('Total Shipping Cost (in ₹)')
plt.legend(title='Courier')
plt.tight_layout()
plt.show()
```



```
import matplotlib.pyplot as plt
# Create a figure with two subplots
fig, ax1 = plt.subplots(figsize=(12, 6))
```



```
# Bar plot for shipping costs per courier-zone combination
ax1.bar(courier_zone_summary['Courier'] + ' - ' +
```

```

courier_zone_summary['Zone'],
    courier_zone_summary['shipping_cost'],
    color='skyblue',
    label='Total Shipping Cost')

<BarContainer object of 21 artists>

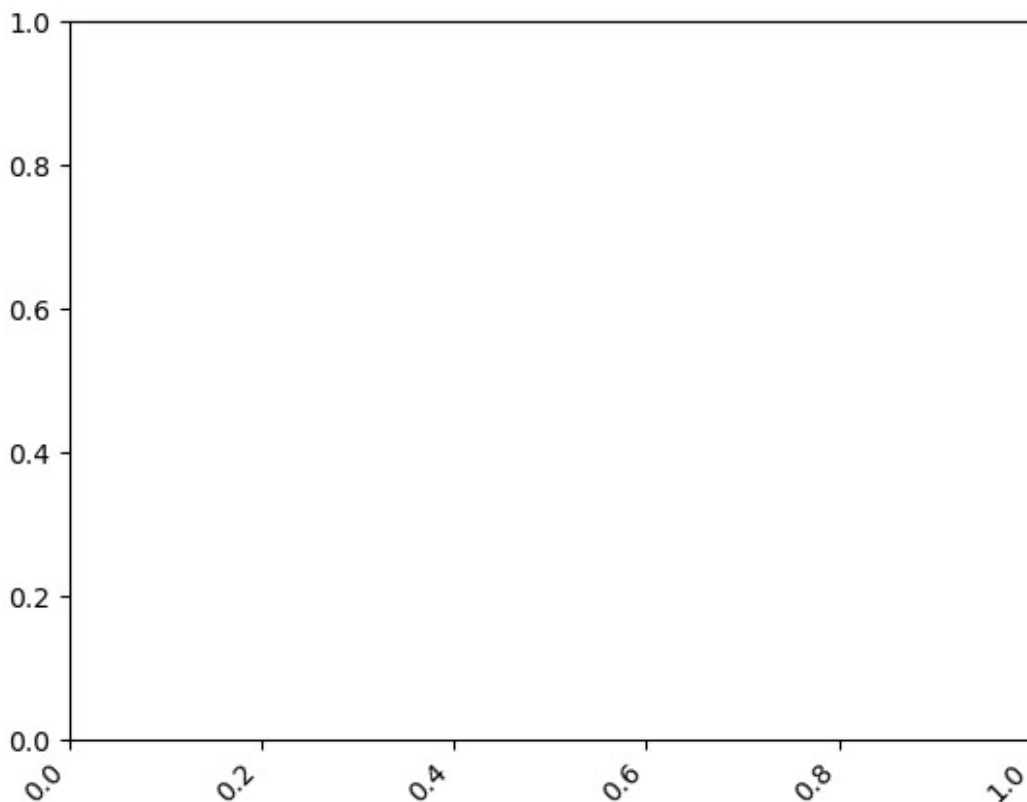
# Set axis labels and title
ax1.set_ylabel('Total Shipping Cost', color='skyblue')
ax1.tick_params(axis='y', labelcolor='skyblue')
ax1.set_title('Total Shipping Cost by Courier and Zone')

Text(0.5, 1.0, 'Total Shipping Cost by Courier and Zone')

# Rotate x-axis labels for better readability
plt.xticks(rotation=45, ha='right')

(array([0. , 0.2, 0.4, 0.6, 0.8, 1. ]),
 [Text(0.0, 0, '0.0'),
  Text(0.2, 0, '0.2'),
  Text(0.4, 0, '0.4'),
  Text(0.6000000000000001, 0, '0.6'),
  Text(0.8, 0, '0.8'),
  Text(1.0, 0, '1.0')])

```



```
# Add grid for better readability
ax1.grid(axis='y')

import warnings

# Suppress the specific UserWarning
warnings.filterwarnings("ignore", category=UserWarning,
module="matplotlib")

# Create your plot
import matplotlib.pyplot as plt

# Create a bar plot for total shipping cost by courier and zone
fig, ax1 = plt.subplots(figsize=(12, 6))

# Bar plot for shipping costs per courier-zone combination
ax1.bar(courier_zone_summary['Courier'] + ' - ' +
courier_zone_summary['Zone'],
        courier_zone_summary['shipping_cost'],
        color='skyblue',
        label='Total Shipping Cost')

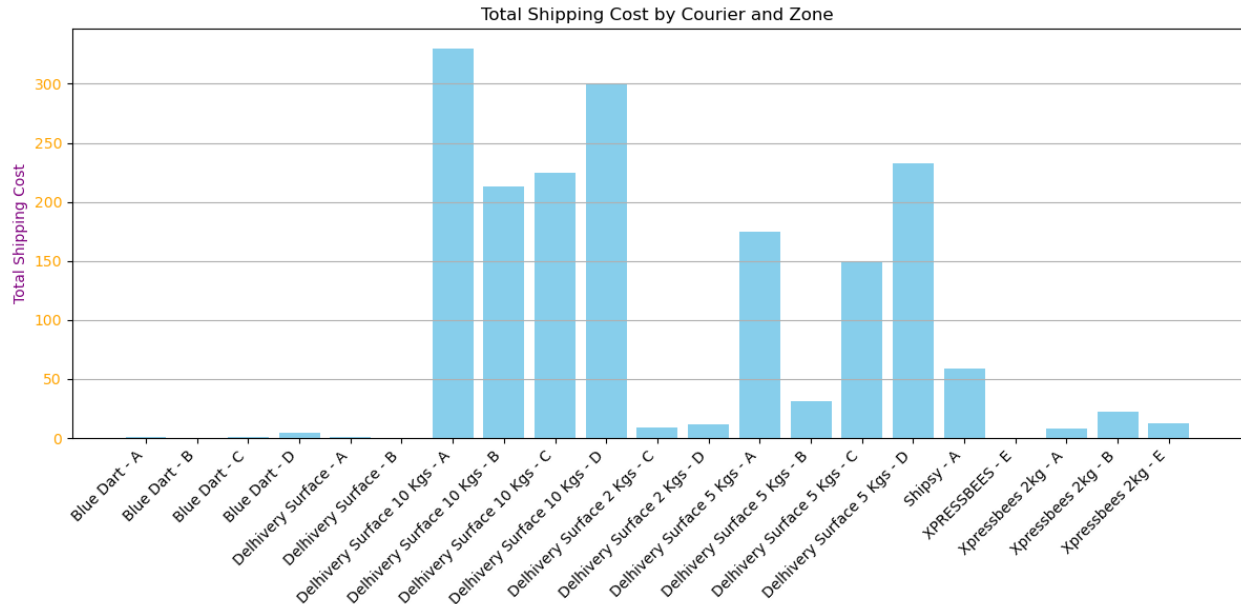
# Set axis labels and title
ax1.set_ylabel('Total Shipping Cost', color='purple')
ax1.tick_params(axis='y', labelcolor='orange')
ax1.set_title('Total Shipping Cost by Courier and Zone')

# Rotate x-axis labels for better readability
plt.xticks(rotation=45, ha='right')

# Add grid for better readability
ax1.grid(axis='y')

# Ensure layout is tight and plot is displayed
fig.tight_layout()

# Show the plot
plt.show()
```



```
# Show the plot
plt.show()

import pandas as pd

# Assuming 'Most_optimum_couriers' and 'courier_zone_summary' contain
the data
output_file_path = 'optimum_service_providers.xlsx'

# Save both data summaries to an Excel file
with pd.ExcelWriter(output_file_path) as writer:
    courier_zone_summary.to_excel(writer, sheet_name='Zonewise Costs',
index=False)
    Most_optimum_couriers.to_excel(writer, sheet_name='Optimum
Providers', index=False)

print(f"Excel file saved at {output_file_path}")

Excel file saved at optimum_service_providers.xlsx

import os
print(os.getcwd())

C:\Users\Minnat Alam

import pandas as pd

# Load the Excel file
df = pd.read_excel('optimum_service_providers.xlsx')
print(df)
```

	Courier	Zone	shipping_cost
0	Blue Dart	A	1.268625
1	Blue Dart	B	0.300000
2	Blue Dart	C	1.260000
3	Blue Dart	D	4.616550
4	Delhivery Surface	A	1.072500
5	Delhivery Surface	B	0.052500
6	Delhivery Surface 10 Kgs	A	330.000000
7	Delhivery Surface 10 Kgs	B	213.000000
8	Delhivery Surface 10 Kgs	C	225.000000
9	Delhivery Surface 10 Kgs	D	300.000000
10	Delhivery Surface 2 Kgs	C	9.000000
11	Delhivery Surface 2 Kgs	D	12.000000
12	Delhivery Surface 5 Kgs	A	174.750000
13	Delhivery Surface 5 Kgs	B	31.500000
14	Delhivery Surface 5 Kgs	C	150.000000
15	Delhivery Surface 5 Kgs	D	232.500000
16	Shipsy	A	58.791750
17	XPRESSBEES	E	0.060000
18	Xpressbees 2kg	A	8.100000
19	Xpressbees 2kg	B	22.200000
20	Xpressbees 2kg	E	12.240000