

```
In [1]: from google.colab import drive
```

```
In [2]: drive.mount('/content/gdrive')
```

Mounted at /content/gdrive

```
In [3]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [4]: d_f=pd.read_excel(r'/content/gdrive/My Drive/Company X - Order Report.xlsx')
Df=pd.read_excel(r'/content/gdrive/My Drive/Company X - Pincode Zones.xlsx')
DF=pd.read_excel(r'/content/gdrive/My Drive/Company X - SKU Master.xlsx')
dF=pd.read_excel(r'/content/gdrive/My Drive/Courier Company - Rates.xlsx')
df=pd.read_excel(r'/content/gdrive/My Drive/Courier Company - Invoice.xlsx')
```

```
In [5]: temp = DF[DF.duplicated(subset=['SKU','Weight (g)'], keep=False)]
temp
```

```
Out[5]:      SKU  Weight (g)
  61 GIFTBOX202002      500
  62 GIFTBOX202002      500
```

```
In [6]: DF = DF.drop_duplicates(subset=['SKU','Weight (g)'], keep='first')
DF
```

```
Out[6]:      SKU  Weight (g)
  0  8904223815682      210
  1  8904223815804      160
  2  8904223815859      165
  3  8904223815866      113
  4  8904223815873       65
  ...
  60 GIFTBOX202001      500
  61 GIFTBOX202002      500
  63 GIFTBOX202003      500
  64 GIFTBOX202004      500
  65  SACHETS001         10
```

65 rows × 2 columns

```
In [7]: Temp = d_f[d_f.duplicated(subset=['ExternOrderNo','SKU','Order Qty'], keep=False)]
Temp
```

Out[7]:

	ExternOrderNo	SKU	Order Qty
107	2001811363	8904223815859	1
110	2001811363	8904223815859	1
316	2001806885	8904223819499	2
317	2001806885	8904223819499	2

```
In [8]: d_f = d_f.drop_duplicates(subset=['ExternOrderNo', 'SKU', 'Order Qty'], keep='first')
d_f
```

Out[8]:

	ExternOrderNo	SKU	Order Qty
0	2001827036	8904223818706	1
1	2001827036	8904223819093	1
2	2001827036	8904223819109	1
3	2001827036	8904223818430	1
4	2001827036	8904223819277	1
...
395	2001806229	8904223818942	1
396	2001806229	8904223818850	1
397	2001806226	8904223818850	2
398	2001806210	8904223816214	1
399	2001806210	8904223818874	1

398 rows × 3 columns

```
In [9]: merge_df=pd.merge(d_f,DF, on='SKU', how='left')
merge_df
```

Out[9]:

	ExternOrderNo	SKU	Order Qty	Weight (g)
0	2001827036	8904223818706	1	127
1	2001827036	8904223819093	1	150
2	2001827036	8904223819109	1	100
3	2001827036	8904223818430	1	165
4	2001827036	8904223819277	1	350
...
393	2001806229	8904223818942	1	133
394	2001806229	8904223818850	1	240
395	2001806226	8904223818850	2	240
396	2001806210	8904223816214	1	120
397	2001806210	8904223818874	1	100

398 rows × 4 columns

In [10]:

```
merge_df['Total Weight']=merge_df['Order Qty']*merge_df['Weight (g)']
merge_df
```

Out[10]:

	ExternOrderNo	SKU	Order Qty	Weight (g)	Total Weight
0	2001827036	8904223818706	1	127	127
1	2001827036	8904223819093	1	150	150
2	2001827036	8904223819109	1	100	100
3	2001827036	8904223818430	1	165	165
4	2001827036	8904223819277	1	350	350
...
393	2001806229	8904223818942	1	133	133
394	2001806229	8904223818850	1	240	240
395	2001806226	8904223818850	2	240	480
396	2001806210	8904223816214	1	120	120
397	2001806210	8904223818874	1	100	100

398 rows × 5 columns

In [11]:

```
result = merge_df.groupby('ExternOrderNo').agg({'Order Qty': 'sum', 'Total Weight': 'sum'})
result.rename(columns={'ExternOrderNo': 'Order ID'}, inplace=True)
result
```

Out[11]:

	Order ID	Order Qty	Total Weight
0	2001806210	2	220
1	2001806226	2	480
2	2001806229	3	500
3	2001806232	8	1302
4	2001806233	2	245
...
119	2001821995	2	477
120	2001822466	10	1376
121	2001823564	6	672
122	2001825261	13	1557
123	2001827036	9	1676

124 rows × 3 columns

In [12]:

```
temp_3 = Df[Df.duplicated(subset=['Warehouse Pincode', 'Customer Pincode', 'Zone'])]
```

Out[12]:

	Warehouse Pincode	Customer Pincode	Zone
2	121003	140301	B
5	121003	140301	B
13	121003	248001	B
16	121003	248001	B
22	121003	302031	B
23	121003	335001	B
28	121003	302002	B
31	121003	313001	B
32	121003	302002	B
34	121003	302017	B
35	121003	302017	B
37	121003	313001	B
38	121003	313001	B
42	121003	313001	B
47	121003	302017	B
58	121003	335001	B
62	121003	302031	B
63	121003	302012	B
68	121003	302018	B
69	121003	302017	B
71	121003	302020	B
72	121003	302018	B
73	121003	302017	B
74	121003	302012	B
78	121003	302020	B
119	121003	173212	E
123	121003	173212	E

In [13]:

```
Df = Df.drop_duplicates(subset=['Warehouse Pincode', 'Customer Pincode', 'Zone'])
```

Out[13]:

	Warehouse Pincode	Customer Pincode	Zone
0	121003	143001	B
1	121003	208019	B
2	121003	140301	B
3	121003	284001	B
4	121003	248006	B
...
118	121003	783301	E
119	121003	173212	E
120	121003	174101	E
121	121003	173213	E
122	121003	175101	E

108 rows × 3 columns

In [14]:

df

Out[14]:

	AWB Code	Order ID	Charged Weight	Warehouse Pincode	Customer Pincode	Zone	Type of Shipment	Billing Amount (Rs.)
0	1091117223244	2001806458	1.00	121003	143001	B	Forward charges	61.3
1	1091117229776	2001806885	1.00	121003	208019	B	Forward charges	61.3
2	1091117323112	2001807058	1.15	121003	140301	B	Forward charges	89.6
3	1091117326612	2001807814	0.79	121003	284001	B	Forward charges	61.3
4	1091117327275	2001807956	1.08	121003	248006	B	Forward charges	89.6
...
119	1091117614452	2001809383	0.50	121003	303702	D	Forward and RTO charges	86.7
120	1091120922803	2001820978	0.50	121003	313301	D	Forward charges	45.4
121	1091121846136	2001811305	0.50	121003	302020	D	Forward charges	45.4
122	1091117435661	2001808295	0.20	121003	673002	E	Forward and RTO charges	107.3
123	1091120014461	2001818390	0.80	121003	783301	E	Forward and RTO charges	213.5

124 rows × 8 columns

In [15]:

```
df = pd.merge(df, Df[['Customer Pincode', 'Zone']], on='Customer Pincode', how='left')
```

Out[15]:

	AWB Code	Order ID	Charged Weight	Warehouse Pincode	Customer Pincode	Zone_x	Type of Shipment	Billing Amount (Rs.)	Zone_
0	1091117223244	2001806458	1.00	121003	143001	B	Forward charges	61.3	
1	1091117229776	2001806885	1.00	121003	208019	B	Forward charges	61.3	
2	1091117323112	2001807058	1.15	121003	140301	B	Forward charges	89.6	
3	1091117326612	2001807814	0.79	121003	284001	B	Forward charges	61.3	
4	1091117327275	2001807956	1.08	121003	248006	B	Forward charges	89.6	
...
119	1091117614452	2001809383	0.50	121003	303702	D	Forward and RTO charges	86.7	
120	1091120922803	2001820978	0.50	121003	313301	D	Forward charges	45.4	
121	1091121846136	2001811305	0.50	121003	302020	D	Forward charges	45.4	
122	1091117435661	2001808295	0.20	121003	673002	E	Forward and RTO charges	107.3	
123	1091120014461	2001818390	0.80	121003	783301	E	Forward and RTO charges	213.5	

124 rows × 9 columns

In [16]: `df.rename(columns={'Zone_y': 'Delivery Zone as per X', 'Zone_x': 'Delivery Zone charge'})`
`df`

Out[16]:

	AWB Code	Order ID	Charged Weight	Warehouse Pincode	Customer Pincode	Delivery Zone charged by Courier Company	Type of Shipment	Billing Amount (Rs.)	Del Zo
0	1091117223244	2001806458	1.00	121003	143001	B	Forward charges	61.3	
1	1091117229776	2001806885	1.00	121003	208019	B	Forward charges	61.3	
2	1091117323112	2001807058	1.15	121003	140301	B	Forward charges	89.6	
3	1091117326612	2001807814	0.79	121003	284001	B	Forward charges	61.3	
4	1091117327275	2001807956	1.08	121003	248006	B	Forward charges	89.6	
...
119	1091117614452	2001809383	0.50	121003	303702	D	Forward and RTO charges	86.7	
120	1091120922803	2001820978	0.50	121003	313301	D	Forward charges	45.4	
121	1091121846136	2001811305	0.50	121003	302020	D	Forward charges	45.4	
122	1091117435661	2001808295	0.20	121003	673002	E	Forward and RTO charges	107.3	
123	1091120014461	2001818390	0.80	121003	783301	E	Forward and RTO charges	213.5	

124 rows × 9 columns

In [17]:

```
cols = ['AWB Code', 'Order ID', 'Charged Weight', 'Warehouse Pincode', 'Customer Pincode', 'Delivery Zone as per X', 'Delivery Zone charged by Courier Company', 'Type of Shipment', 'Billing Amount (Rs.)']
df = df[cols]
```

Out[17]:

	AWB Code	Order ID	Charged Weight	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment	B Arr
0	1091117223244	2001806458	1.00	121003	143001	B	B	Forward charges	
1	1091117229776	2001806885	1.00	121003	208019	B	B	Forward charges	
2	1091117323112	2001807058	1.15	121003	140301	B	B	Forward charges	
3	1091117326612	2001807814	0.79	121003	284001	B	B	Forward charges	
4	1091117327275	2001807956	1.08	121003	248006	B	B	Forward charges	
...
119	1091117614452	2001809383	0.50	121003	303702	B	D	Forward and RTO charges	
120	1091120922803	2001820978	0.50	121003	313301	B	D	Forward charges	
121	1091121846136	2001811305	0.50	121003	302020	B	D	Forward charges	
122	1091117435661	2001808295	0.20	121003	673002	E	E	Forward and RTO charges	
123	1091120014461	2001818390	0.80	121003	783301	E	E	Forward and RTO charges	

124 rows × 9 columns



In [18]: `df = pd.merge(df, result[['Order ID', 'Total Weight']], on='Order ID', how='left')`

Out[18]:

	AWB Code	Order ID	Charged Weight	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment	B
0	1091117223244	2001806458	1.00	121003	143001	B	B	Forward charges	
1	1091117229776	2001806885	1.00	121003	208019	B	B	Forward charges	
2	1091117323112	2001807058	1.15	121003	140301	B	B	Forward charges	
3	1091117326612	2001807814	0.79	121003	284001	B	B	Forward charges	
4	1091117327275	2001807956	1.08	121003	248006	B	B	Forward charges	
...
119	1091117614452	2001809383	0.50	121003	303702	B	D	Forward and RTO charges	
120	1091120922803	2001820978	0.50	121003	313301	B	D	Forward charges	
121	1091121846136	2001811305	0.50	121003	302020	B	D	Forward charges	
122	1091117435661	2001808295	0.20	121003	673002	E	E	Forward and RTO charges	
123	1091120014461	2001818390	0.80	121003	783301	E	E	Forward and RTO charges	

124 rows × 10 columns

In [19]:

```
df['Total Weight'] = df['Total Weight'] / 1000
cols = [
    'AWB Code', 'Order ID', 'Charged Weight', 'Total Weight', 'Warehouse Pincode',
    'Customer Pincode', 'Delivery Zone as per X', 'Delivery Zone charged by Courier Company',
    'Type of Shipment', 'Billing Amount (Rs.)'
]
df = df[cols]
df
```

Out[19]:

	AWB Code	Order ID	Charged Weight	Total Weight	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type Shipment
0	1091117223244	2001806458	1.00	0.700	121003	143001	B	B	For char
1	1091117229776	2001806885	1.00	0.420	121003	208019	B	B	For char
2	1091117323112	2001807058	1.15	1.168	121003	140301	B	B	For char
3	1091117326612	2001807814	0.79	0.607	121003	284001	B	B	For char
4	1091117327275	2001807956	1.08	1.080	121003	248006	B	B	For char
...
119	1091117614452	2001809383	0.50	0.607	121003	303702	B	D	For and cha
120	1091120922803	2001820978	0.50	0.515	121003	313301	B	D	For char
121	1091121846136	2001811305	0.50	0.750	121003	302020	B	D	For char
122	1091117435661	2001808295	0.20	0.245	121003	673002	E	E	For and cha
123	1091120014461	2001818390	0.80	0.841	121003	783301	E	E	For and cha

124 rows × 10 columns

◀	▶
---	---

In [20]: dF

Out[20]:

	Zone	Weight Slabs	Forward Fixed Charge	Forward Additional Weight Slab Charge	RTO Fixed Charge	RTO Additional Weight Slab Charge
0	A	0.50	29.5	23.6	13.6	23.6
1	B	1.00	33.0	28.3	20.5	28.3
2	C	1.25	40.1	38.9	31.9	38.9
3	D	1.50	45.4	44.8	41.3	44.8
4	E	2.00	56.6	55.5	50.7	55.5

```
In [21]: df = pd.merge(df, df[['Zone', 'Weight Slabs']], left_on='Delivery Zone as per X', right_on='Delivery Zone charged by Courier Company')
df
```

Out[21]:

	AWB Code	Order ID	Charged Weight	Total Weight	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type	Shipper
0	1091117223244	2001806458	1.00	0.700	121003	143001	B	B	For cha	
1	1091117229776	2001806885	1.00	0.420	121003	208019	B	B	For cha	
2	1091117323112	2001807058	1.15	1.168	121003	140301	B	B	For cha	
3	1091117326612	2001807814	0.79	0.607	121003	284001	B	B	For cha	
4	1091117327275	2001807956	1.08	1.080	121003	248006	B	B	For cha	
...	
119	1091117614452	2001809383	0.50	0.607	121003	303702	B	D	For and cha	
120	1091120922803	2001820978	0.50	0.515	121003	313301	B	D	For cha	
121	1091121846136	2001811305	0.50	0.750	121003	302020	B	D	For cha	
122	1091117435661	2001808295	0.20	0.245	121003	673002	E	E	For and cha	
123	1091120014461	2001818390	0.80	0.841	121003	783301	E	E	For and cha	

124 rows × 12 columns

```
In [22]: df = pd.merge(df, df[['Zone', 'Weight Slabs']], left_on='Delivery Zone charged by Courier Company', right_on='Delivery Zone as per X')
df
```

Out[22]:

	AWB Code	Order ID	Charged Weight	Total Weight	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type	Shipper
0	1091117223244	2001806458	1.00	0.700	121003	143001	B	B	Foncha	
1	1091117229776	2001806885	1.00	0.420	121003	208019	B	B	Foncha	
2	1091117323112	2001807058	1.15	1.168	121003	140301	B	B	Foncha	
3	1091117326612	2001807814	0.79	0.607	121003	284001	B	B	Foncha	
4	1091117327275	2001807956	1.08	1.080	121003	248006	B	B	Foncha	
...
119	1091117614452	2001809383	0.50	0.607	121003	303702	B	D	Fon and cha	
120	1091120922803	2001820978	0.50	0.515	121003	313301	B	D	Foncha	
121	1091121846136	2001811305	0.50	0.750	121003	302020	B	D	Foncha	
122	1091117435661	2001808295	0.20	0.245	121003	673002	E	E	Fon and cha	
123	1091120014461	2001818390	0.80	0.841	121003	783301	E	E	Fon and cha	

124 rows × 14 columns

In [23]:

```
df.rename(columns={'Weight Slabs_x': 'Weight slab as per X (KG)', 'Weight Slabs_y': 'Weight slab as per Courier Company (KG)', 'Charged Weight': 'Total weight as per Courier Company (KG)', 'Total weight': 'Total weight as per X (KG)'}, inplace=True)
df
```

Out[23]:

	AWB Code	Order ID	Total weight as per Courier Company (KG)	Total weight as per X (KG)	Warehouse Pincode	Customer Pincode	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Ship
0	1091117223244	2001806458	1.00	0.700	121003	143001	B	B	For ch
1	1091117229776	2001806885	1.00	0.420	121003	208019	B	B	For ch
2	1091117323112	2001807058	1.15	1.168	121003	140301	B	B	For ch
3	1091117326612	2001807814	0.79	0.607	121003	284001	B	B	For ch
4	1091117327275	2001807956	1.08	1.080	121003	248006	B	B	For ch
...
119	1091117614452	2001809383	0.50	0.607	121003	303702	B	D	For anc ch
120	1091120922803	2001820978	0.50	0.515	121003	313301	B	D	For ch
121	1091121846136	2001811305	0.50	0.750	121003	302020	B	D	For ch
122	1091117435661	2001808295	0.20	0.245	121003	673002	E	E	For anc ch
123	1091120014461	2001818390	0.80	0.841	121003	783301	E	E	For anc ch

124 rows × 14 columns

In [24]:

```
df = df.drop(columns=['Warehouse Pincode', 'Customer Pincode', 'Zone_x', 'Zone_y'])

cols = [
    'Order ID', 'AWB Code', 'Total weight as per X (KG)', 'Weight slab as per X (KG)'
    'Total weight as per Courier Company (KG)', 'Weight slab charged by Courier Company',
    'Delivery Zone as per X', 'Delivery Zone charged by Courier Company', 'Type of Shipment',
    'Billing Amount (Rs.)'
]
df = df[cols]
df
```

Out[24]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	Forward charge
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	Forward charge
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	Forward charge
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	Forward charge
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	Forward charge
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	Forward and RTC charge
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	Forward charge
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	Forward charge
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	Forward and RTC charge
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	Forward and RTC charge

124 rows × 10 columns



In [25]: `df['Unit Factor']=1
df`

Out[25]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	Forward charge
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	Forward charge
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	Forward charge
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	Forward charge
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	Forward charge
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	Forward and RTC charge
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	Forward charge
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	Forward charge
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	Forward and RTC charge
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	Forward and RTC charge

124 rows × 11 columns

In [26]: `df['additional weight factor'] = np.ceil(df['Total weight as per X (KG)'] / df['Weight slab charged by Courier Company (KG)'])`

Out[26]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	Forward charge
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	Forward charge
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	Forward charge
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	Forward charge
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	Forward charge
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	Forward and RTC charge
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	Forward charge
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	Forward charge
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	Forward and RTC charge
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	Forward and RTC charge

124 rows × 12 columns

```
In [27]: def calculate_total(row, df):
    zone = row['Delivery Zone as per X']
    unit_factor = row['Unit Factor']
    add_weight_factor = row['additional weight factor']

    charges = df[df['Zone'] == zone].iloc[0]

    if row['Type of Shipment'] == 'Forward charges':
        total = (charges['Forward Fixed Charge'] * unit_factor) + (charges['Forward Ad
    elif row['Type of Shipment'] == 'Forward and RTC charges':
        total = ((charges['Forward Fixed Charge'] + charges['RTC Fixed Charge']) * uni
    else:
        total = 0

    return total
```

```
df['total'] = df.apply(calculate_total, axis=1, dF=dF)
df
```

Out[27]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Type of Shipment
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	Forward charge
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	Forward charge
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	Forward charge
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	Forward charge
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	Forward charge
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	Forward and RTC charge
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	Forward charge
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	Forward charge
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	Forward and RTC charge
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	Forward and RTC charge

124 rows × 13 columns

In [28]:

```
df.rename(columns={'Billing Amount (Rs.)': 'Charges Billed by Courier Company (Rs.)',
                  'total': 'Expected Charge as per X (Rs.)'}, inplace=True)
df=df.drop(columns=['Unit Factor', 'additional weight factor', 'Type of Shipment'])
df
```

Out[28]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Charges Billed by Courier Company (Rs.)
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	61.0
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	61.0
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	89.0
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	61.0
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	89.0
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	86.0
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	45.4
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	45.4
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	107.0
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	213.0

124 rows × 10 columns

In [29]: df['Difference Between Expected Charges and Billed Charges (Rs.)'] = (df['Charges Billed by Courier Company (Rs.)'] - df['Expected Charges (Rs.)'])		

Out[29]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Charges Billed by Courier Company (Rs.)
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	61.0
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	61.0
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	89.0
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	61.0
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	89.0
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	86.0
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	45.4
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	45.4
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	107.0
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	213.0

124 rows × 11 columns



In [30]: df['Difference Between Expected Charges and Billed Charges (Rs.)'] = df['Difference Be
df

Out[30]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Charges Billed by Courier Company (Rs.)
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	61.0
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	61.0
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	89.0
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	61.0
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	89.0
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	86.0
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	45.4
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	45.4
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	107.0
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	213.0

124 rows × 11 columns

In [31]:

```

conditions = [
    (df['Difference Between Expected Charges and Billed Charges (Rs.)'] == 0),
    (df['Difference Between Expected Charges and Billed Charges (Rs.)'] > 0),
    (df['Difference Between Expected Charges and Billed Charges (Rs.)'] < 0)
]

categories = ['Correctly Charged', 'Over Charged', 'Under Charged']

df['Charge Status'] = pd.Series([''] * len(df))

for i in range(len(conditions)):
    df.loc[conditions[i], 'Charge Status'] = categories[i]

summary = df.groupby('Charge Status').agg(
    Count=('Order ID', 'size'),
    Amount= ('Difference Between Expected Charges and Billed Charges (Rs.)', 'sum')
).reset_index()

summary = summary.rename(columns={'Charge Status': ''})
summary = summary.set_index('').reindex(['Correctly Charged', 'Over Charged', 'Under Charged'])

print(summary)

```

		Count	Amount
0	Correctly Charged	8	0.0
1	Over Charged	114	7751.2
2	Under Charged	2	-47.2

In [32]: df

Out[32]:

	Order ID	AWB Code	Total weight as per X (KG)	Weight slab as per X (KG)	Total weight as per Courier Company (KG)	Weight slab charged by Courier Company (KG)	Delivery Zone as per X	Delivery Zone charged by Courier Company	Charged Billed by Courier Company (Rs.)
0	2001806458	1091117223244	0.700	1.0	1.00	1.0	B	B	61.0
1	2001806885	1091117229776	0.420	1.0	1.00	1.0	B	B	61.0
2	2001807058	1091117323112	1.168	1.0	1.15	1.0	B	B	89.0
3	2001807814	1091117326612	0.607	1.0	0.79	1.0	B	B	61.0
4	2001807956	1091117327275	1.080	1.0	1.08	1.0	B	B	89.0
...
119	2001809383	1091117614452	0.607	1.0	0.50	1.5	B	D	86.0
120	2001820978	1091120922803	0.515	1.0	0.50	1.5	B	D	45.4
121	2001811305	1091121846136	0.750	1.0	0.50	1.5	B	D	45.4
122	2001808295	1091117435661	0.245	2.0	0.20	2.0	E	E	107.0
123	2001818390	1091120014461	0.841	2.0	0.80	2.0	E	E	213.0

124 rows × 12 columns

