METRO-ANALYSIS (/github/DISHANKKARAMPUDI/METRO-ANALYSIS/tree/main)
/ Metro.ipynb (/github/DISHANKKARAMPUDI/METRO-ANALYSIS/tree/main/Metro.ipynb)

Import Pandas to handle with dataframes

In [1]:
```
import pandas as pd
```

Load the dataset

In [2]:
```
df=pd.read_csv("metro.csv")
df
```

Out[2]:

	Station ID	Station Name	Distance from Start (km)	Line	Opening Date	Station Layout	Latitude	Longitude
0	1	Jhil Mil	10.3	Red line	2008-04-06	Elevated	28.675790	77.312390
1	2	Welcome [Conn: Red]	46.8	Pink line	2018-10-31	Elevated	28.671800	77.277560
2	3	DLF Phase 3	10.0	Rapid Metro	2013-11-14	Elevated	28.493600	77.093500
3	4	Okhla NSIC	23.8	Magenta line	2017-12-25	Elevated	28.554483	77.264849
4	5	Dwarka Mor	10.2	Blue line	2005-12-30	Elevated	28.619320	77.033260
...	...	...	...	...	...	...	...	...
280	281	Bata Chowk	38.3	Voilet line	2015-06-09	Elevated	28.385836	77.313462
281	282	Dwarka Sector 12	5.8	Blue line	2006-01-04	Elevated	28.592320	77.040510
282	283	Noida Sector 18	43.6	Blue line	2009-12-11	Elevated	28.570810	77.326120
283	284	Knowledge Park II	21.4	Aqua line	2019-01-25	Elevated	28.456867	77.500054
284	285	Mayur Vihar Extention	39.5	Blue line	2009-12-11	Elevated	28.594158	77.294589

285 rows × 8 columns

Other libraries used in the analysis process

In [3]:
```
import folium
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
import plotly.io as pio
pio.templates.default='plotly_white'
metro_data=df
```

In [4]:
```
metro_data.head()
```

Out[4]:

	Station ID	Station Name	Distance from Start (km)	Line	Opening Date	Station Layout	Latitude	Longitude
0	1	Jhil Mil	10.3	Red line	2008-04-06	Elevated	28.675790	77.312390
1	2	Welcome [Conn: Red]	46.8	Pink line	2018-10-31	Elevated	28.671800	77.277560
2	3	DLF Phase 3	10.0	Rapid Metro	2013-11-14	Elevated	28.493600	77.093500
3	4	Okhla NSIC	23.8	Magenta line	2017-12-25	Elevated	28.554483	77.264849
4	5	Dwarka Mor	10.2	Blue line	2005-12-30	Elevated	28.619320	77.033260

Check for missing values

In [5]:
```
metro_data.isnull().sum()
```

Out[5]:
```
Station ID                  0
Station Name                0
Distance from Start (km)    0
Line                        0
Opening Date                0
Station Layout              0
Latitude                    0
Longitude                   0
dtype: int64
```

In [6]:
```
metro_data.dtypes
```

Out[6]:
```
Station ID                   int64
Station Name                object
Distance from Start (km)    float64
Line                         object
Opening Date                 object
Station Layout               object
Latitude                    float64
Longitude                   float64
dtype: object
```

Convert date data-type from object to datetime

In [7]:
```
metro_data['Opening Date']=pd.to_datetime(metro_data['Opening Date'])
```

Assign a dictionary to make colours for various metro lines

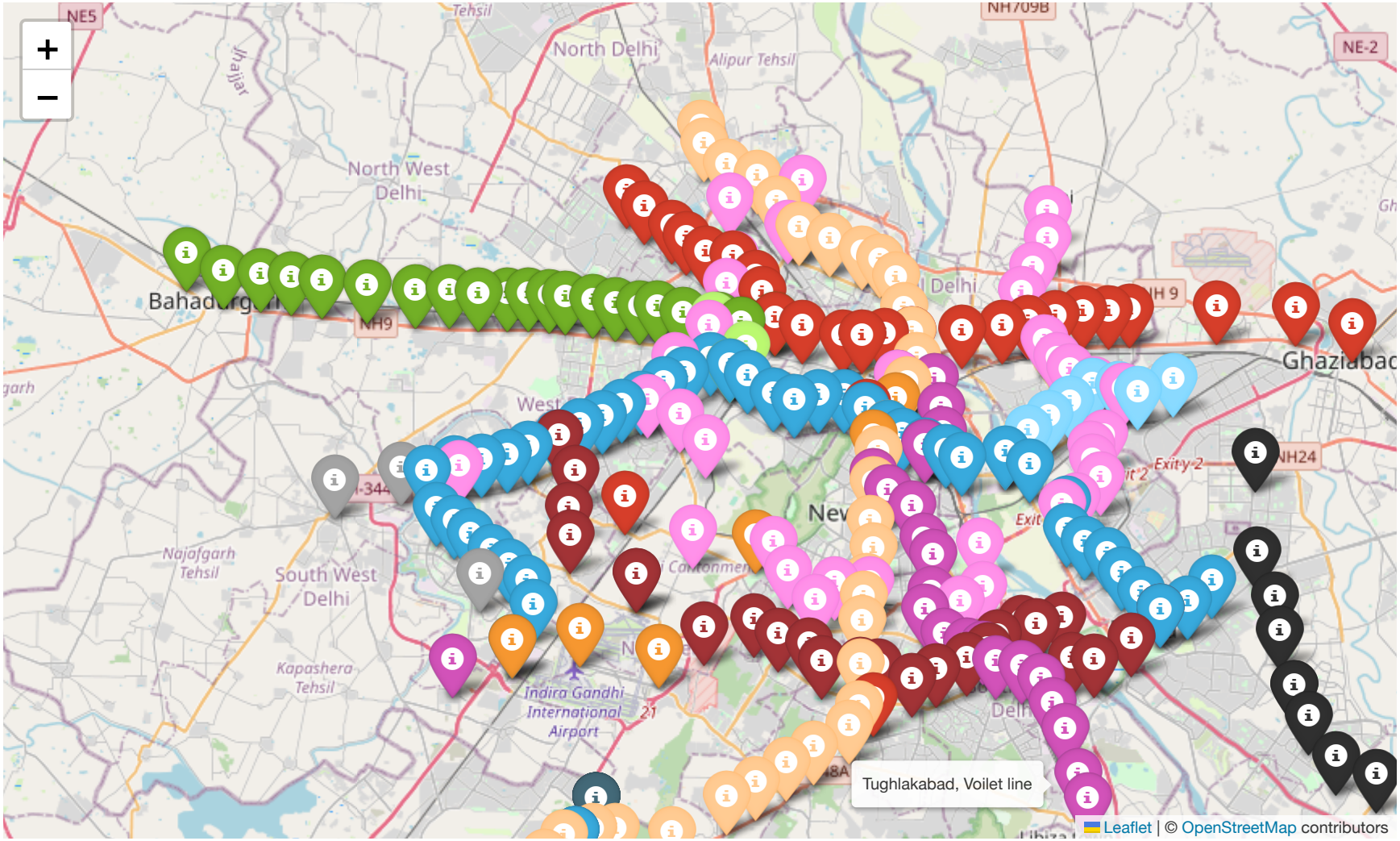
In [8]:
```
line_colors = {
    'Red line': 'red',
    'Blue line': 'blue',
    'Yellow line': 'beige',
    'Green line': 'green',
    'Voilet line': 'purple',
    'Pink line': 'pink',
    'Magenta line': 'darkred',
    'Orange line': 'orange',
    'Rapid Metro': 'cadetblue',
    'Aqua line': 'black',
    'Green line branch': 'lightgreen',
    'Blue line branch': 'lightblue',
    'Gray line': 'lightgray'
}
```

Get a map(visual pictures) of the metro network

In [9]:
```
delhi_map_with_line_tooltip = folium.Map(location=[28.7041, 77.1025], zoom_start=11)
for index,row in metro_data.iterrows():
    line=row['Line']
    color = line_colors.get(line, 'black')
    folium.Marker(
        location=[row['Latitude'], row['Longitude']],
        popup=f"{row['Station Name']}",
        tooltip=f"{row['Station Name']}, {line}",
        icon=folium.Icon(color=color)
    ).add_to(delhi_map_with_line_tooltip)
```

In [10]:
```
delhi_map_with_line_tooltip
```

Out[10]:



In [11]:
```
metro_data['Opening Year'] = metro_data['Opening Date'].dt.year
```

In [12]:
```
metro_data['Opening Year']
```

Out[12]:
```
0      2008
1      2018
2      2013
3      2017
4      2005
       ...
280    2015
281    2006
282    2009
283    2019
284    2009
Name: Opening Year, Length: 285, dtype: int32
```

In [13]:
```
number_year=metro_data['Opening Year'].value_counts().sort_index()
```

In [14]:
```
number_year
```

Out[14]:
```
Opening Year
2002     6
2003     4
2004    11
2005    28
2006     9
2008     3
2009    17
2010    54
2011    13
2013     5
2014     3
2015    13
2017    18
2018    64
2019    37
Name: count, dtype: int64
```

In [15]:
```
df1=number_year.reset_index()
```

In [16]:
```
df1
```

Out[16]:

	Opening Year	count
0	2002	6
1	2003	4
2	2004	11
3	2005	28
4	2006	9
5	2008	3
6	2009	17
7	2010	54
8	2011	13
9	2013	5
10	2014	3
11	2015	13
12	2017	18
13	2018	64
14	2019	37

In [17]:
```
df1.columns = ['Year', 'Number of Stations']
```

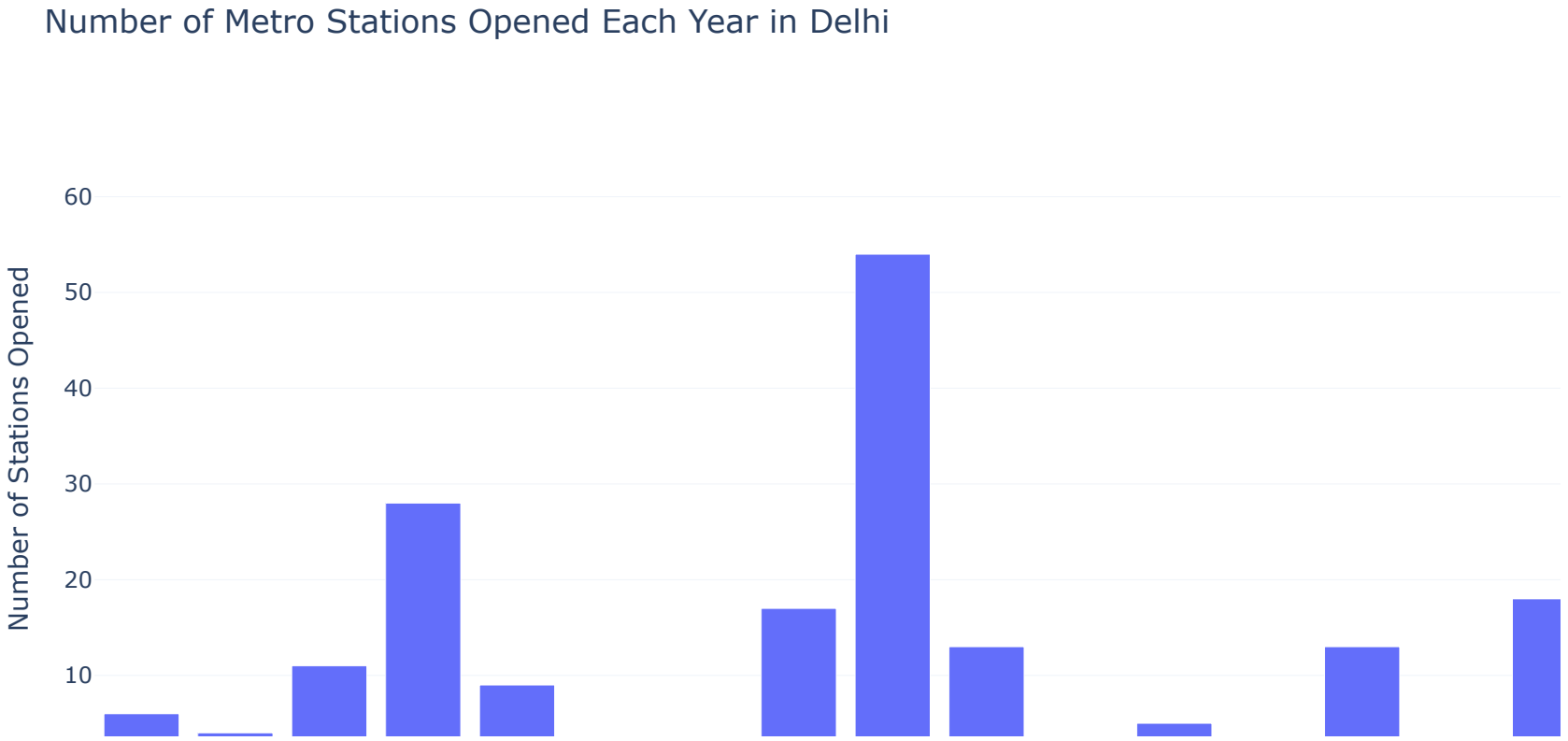
In [18]:
```
df1
```

Out[18]:

	Year	Number of Stations
0	2002	6
1	2003	4
2	2004	11
3	2005	28
4	2006	9
5	2008	3
6	2009	17
7	2010	54
8	2011	13
9	2013	5
10	2014	3
11	2015	13
12	2017	18
13	2018	64
14	2019	37

In [19]:
```
fig = px.bar(df1, x='Year', y='Number of Stations',
             title="Number of Metro Stations Opened Each Year in Delhi",
             labels={'Year': 'Year', 'Number of Stations': 'Number of Stations Opened'})
```

In [20]:
```
fig.update_layout(xaxis_tickangle=-45, xaxis=dict(tickmode='linear'),
                  yaxis=dict(title='Number of Stations Opened'),
                  xaxis_title="Year")
fig.show()
```



The above plot gives the information about number of metro lines opened per year

In [21]:
```
y=metro_data.groupby('Line')['Distance from Start (km)'].max()
```

In [22]:
```
x=metro_data['Line'].value_counts()
```

In [23]:
```
x
```

Out[23]:
```
Line
Blue line            49
Pink line            38
Yellow line          37
Voilet line          34
Red line             29
Magenta line         25
Aqua line            21
Green line           21
Rapid Metro          11
Blue line branch      8
Orange line           6
Gray line             3
Green line branch     3
Name: count, dtype: int64
```

In [24]:
```
y/(x-1)
```

Out[24]:
```
Line
Aqua line            1.355000
Blue line            1.097917
Blue line branch     1.157143
Gray line            1.950000
Green line           1.240000
Green line branch    1.050000
Magenta line         1.379167
Orange line          4.160000
Pink line            1.421622
Rapid Metro          1.000000
Red line             1.167857
Voilet line          1.318182
Yellow line          1.269444
dtype: float64
```

In [25]:
```
line_analysis = pd.DataFrame({
    'Line': x.index,
    'Number of Stations': x.values,
    'Average Distance Between Stations (km)': y/(x-1)
})

line_analysis = line_analysis.sort_values(by='Number of Stations', ascending=False)

line_analysis.reset_index(drop=True, inplace=True)
print(line_analysis)
```

```
                 Line  Number of Stations  \
0           Blue line                  49
1           Pink line                  38
2         Yellow line                  37
3         Voilet line                  34
4            Red line                  29
5        Magenta line                  25
6           Aqua line                  21
7          Green line                  21
8         Rapid Metro                  11
9    Blue line branch                   8
10        Orange line                   6
11          Gray line                   3
12  Green line branch                   3

    Average Distance Between Stations (km)
0                                 1.355000
1                                 1.097917
2                                 1.157143
3                                 1.950000
4                                 1.240000
5                                 1.050000
6                                 1.379167
7                                 4.160000
8                                 1.421622
9                                 1.000000
10                                1.167857
11                                1.318182
12                                1.269444
```

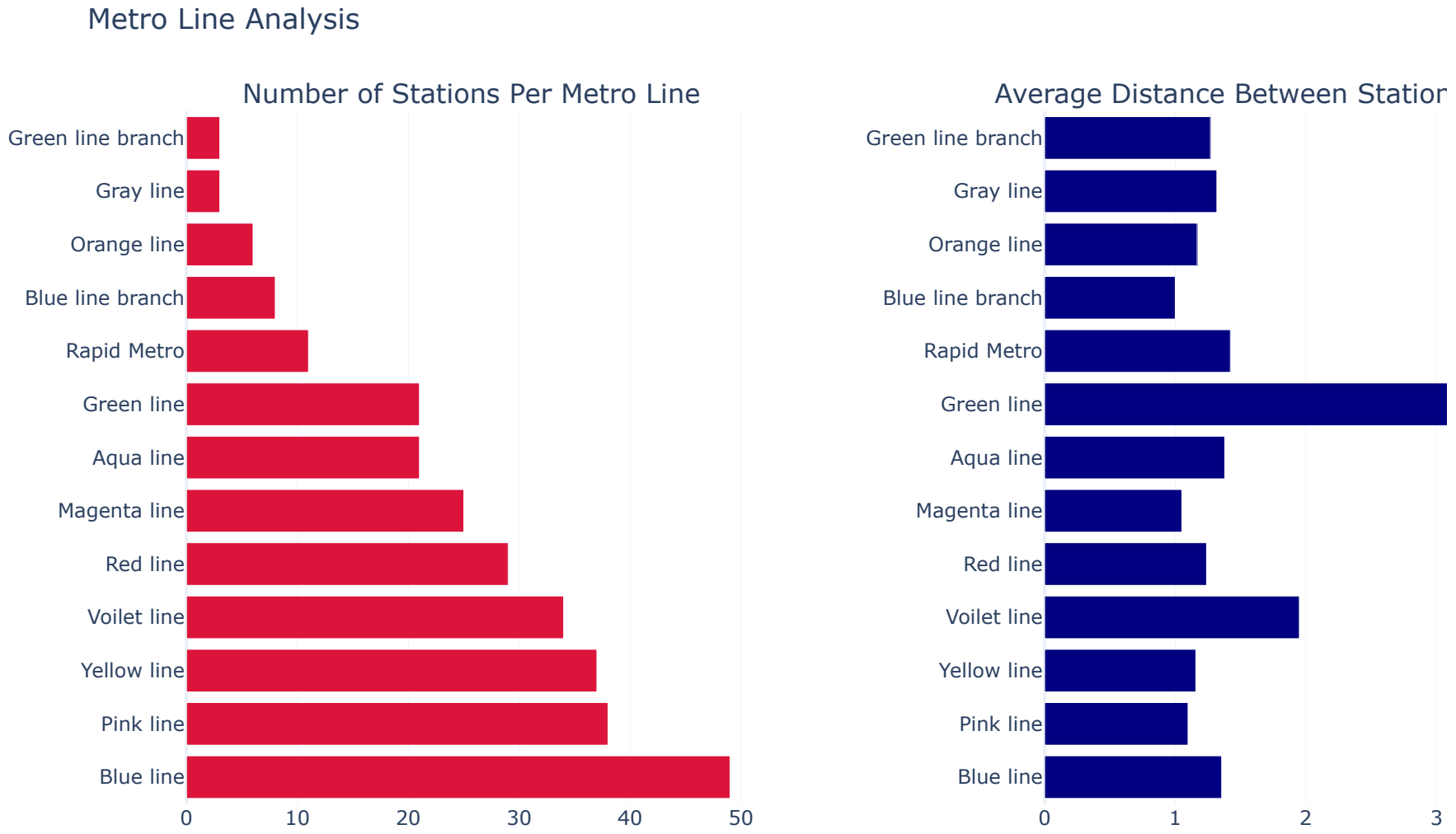
In [26]:
```
fig1 = make_subplots(rows=1, cols=2, subplot_titles=('Number of Stations Per Metro Line',
                                                     'Average Distance Between Stations Per Metro Line'),
                     horizontal_spacing=0.2)

fig1.add_trace(
    go.Bar(y=line_analysis['Line'], x=line_analysis['Number of Stations'],
           orientation='h', name='Number of Stations', marker_color='crimson'),
    row=1, col=1
)

fig1.add_trace(
    go.Bar(y=line_analysis['Line'], x=line_analysis['Average Distance Between Stations (km)'],
           orientation='h', name='Average Distance (km)', marker_color='navy'),
    row=1, col=2
)

fig1.update_layout(height=600, width=1200, title_text="Metro Line Analysis", template="plotly_white")

fig1.show()
```



The above plots, gives info about each metro line length and the total number of stations in that line

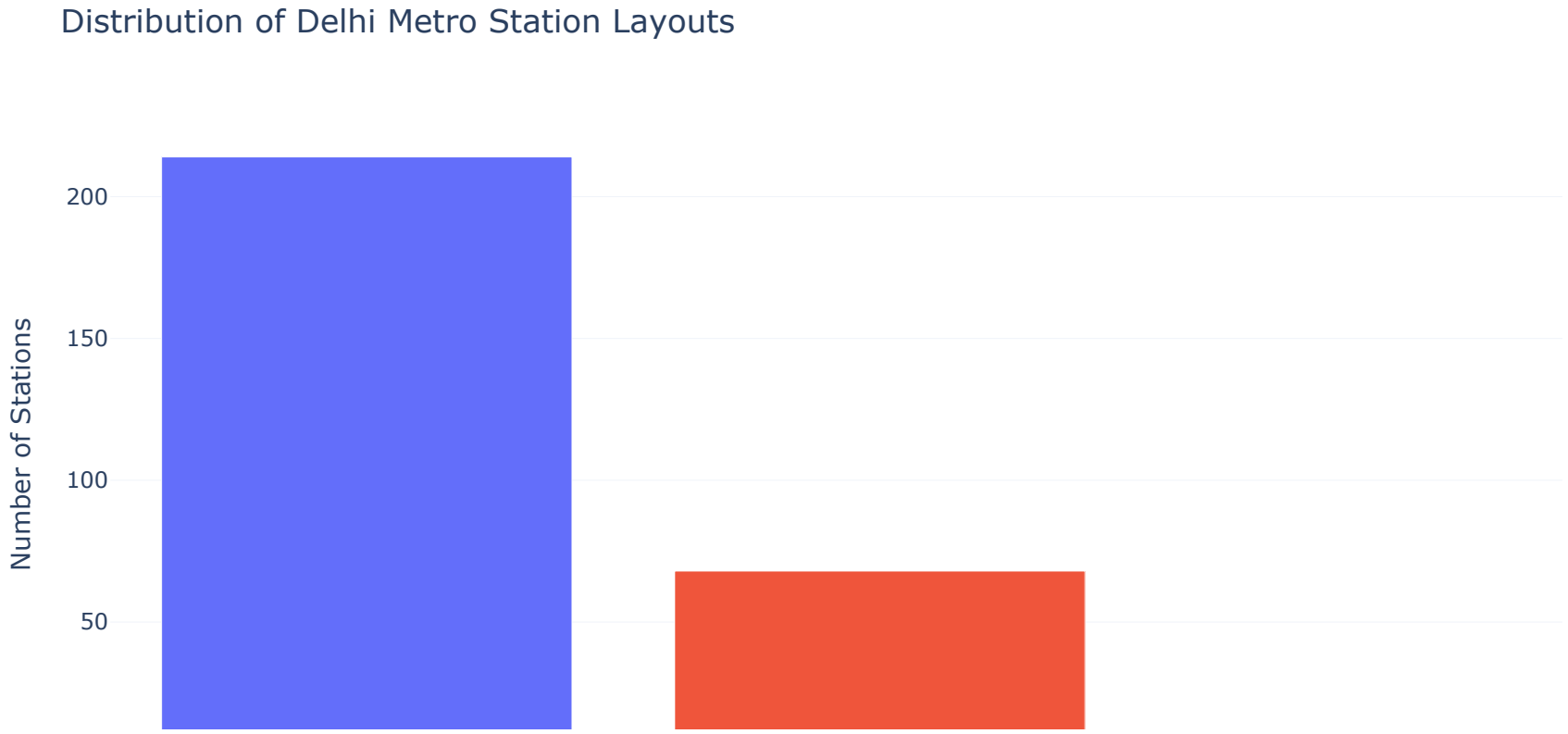
In [27]:
```
layout_counts = metro_data['Station Layout'].value_counts()
```

In [28]:
```
layout_counts
```

Out[28]:
```
Station Layout
Elevated       214
Underground     68
At-Grade         3
Name: count, dtype: int64
```

In [29]:
```
fig2 = px.bar(x=layout_counts.index, y=layout_counts.values,
              labels={'x': 'Station Layout', 'y': 'Number of Stations'},
              title='Distribution of Delhi Metro Station Layouts',
              color=layout_counts.index,
              )
```

In [30]:
```
fig2.show()
```



This plot gives info about number of elevated/underground/on-road metro stations

In [ ]: