London Underground Traffic Patterns Before and After the Pandemic

In [25]:

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [26]:

```
EE_2020_by_day = pd.read_csv('entries_exits_2020_by_day_NUMBAT.csv')
EE_2020_by_timeblock = pd.read_csv('entries_exits_2020_by_time_block_NUMBAT.csv')
EE_2022_by_day = pd.read_csv('entries_exits_2022_by_day_NUMBAT.csv')
EE_2022_by_timeblock = pd.read_csv('entries_exits_2022_by_time_block_NUMBAT.csv')
```

Summary Statistics

In [27]:

```
summary_2020_by_day = EE_2020_by_day.describe()
summary_2020_by_timeblock = EE_2020_by_timeblock.describe()
summary_2022_by_day = EE_2022_by_day.describe()
summary_2022_by_timeblock = EE_2022_by_timeblock.describe()

print("2020 Entries and Exits by Day Summary:")
print(summary_2020_by_day)
print("\n2020 Entries and Exits by Time Block Summary:")
print(summary_2020_by_timeblock)
print("\n2022 Entries and Exits by Day Summary:")
print(summary_2022_by_day)
print("\n2022 Entries and Exits by Time Block Summary:")
print(summary_2022_by_timeblock)
```

```
2020 Entries and Exits by Day Summary:
                nlc  total_mon_to_thur_entries  total_friday_entries  \
count    464.000000                  464.000000            464.000000
mean    2423.213362                 3183.625000           3170.556034
std     3015.621155                 3503.830109           3503.928749
min      500.000000                    0.000000              0.000000
25%      616.750000                 1174.750000           1173.500000
50%      742.500000                 2117.000000           2116.000000
75%     3170.250000                 3742.750000           3663.000000
max     9587.000000                25787.000000          26275.000000

       total_saturday_entries  total_sunday_entries  total_mon_to_thur_exits  \
count              464.000000            464.000000               464.000000
mean              2061.756466           1432.030172              3198.355603
std               2459.827900           1798.576622              3697.123858
min                  0.000000              0.000000                 0.000000
25%                722.500000            495.000000              1171.250000
50%               1344.000000            927.000000              2070.000000
75%               2428.750000           1653.500000              3625.250000
max              23599.000000          19547.000000             26479.000000

       total_friday_exits  total_saturday_exits  total_sunday_exits  \
count          464.000000            464.000000          464.000000
mean          3183.495690           2035.810345         1380.603448
std           3654.206975           2601.757722         1809.928310
min              0.000000              0.000000            0.000000
25%           1163.000000            630.250000          395.750000
50%           2068.000000           1287.500000          841.500000
75%           3607.750000           2330.250000         1565.000000
max          27444.000000          23859.000000        19026.000000
```

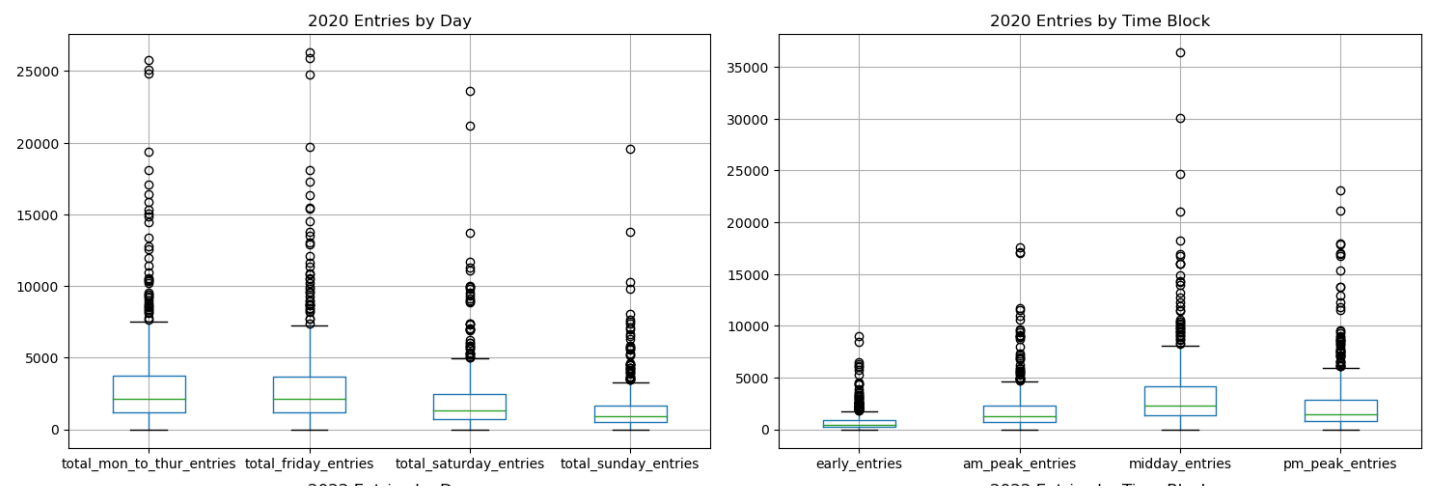
```
       net_mon_to_thur   net_friday  net_saturday   net_sunday
count       464.000000   464.000000    464.000000   464.000000
mean        -14.730603   -12.939655     25.946121    51.426724
std         477.816785   390.326718    432.853009   364.597484
min       -6935.000000 -2356.000000  -4224.000000 -1944.000000
25%         -77.250000   -95.500000    -45.250000   -50.250000
50%          23.500000    18.500000     21.000000     7.000000
75%         128.500000   119.250000    126.250000    81.250000
max        1689.000000  2218.000000   2616.000000  4816.000000

2020 Entries and Exits by Time Block Summary:
              nlc  total_entries  early_entries  am_peak_entries  \
count   464.000000     464.000000     464.000000       464.000000
mean   2423.213362    9847.967672     781.829741      1939.601293
std    3015.621155   11116.324984    1117.905908      2243.547696
min     500.000000       0.000000       0.000000         0.000000
25%     616.750000    3662.000000     206.750000       738.500000
50%     742.500000    6587.000000     399.000000      1293.000000
75%    3170.250000   11362.500000     845.750000      2308.250000
max    9587.000000   94163.000000    9033.000000     17559.000000

       midday_entries  pm_peak_entries  evening_entries  late_entries  \
count      464.000000         464.00000       464.000000    464.000000
mean      3485.071121        2490.31681       890.025862    261.383621
std       3861.676830        3135.63935      1184.089177    356.148960
min          0.000000           0.00000         0.000000      0.000000
25%       1381.500000         766.00000       243.750000     63.000000
50%       2282.000000        1479.00000       479.000000    138.000000
75%       4125.250000        2877.50000      1093.750000    300.500000
max      36371.000000       23067.00000      8615.000000   2507.000000

        total_exits   early_exits  ...  pm_peak_exits  evening_exits  \
count    464.000000    464.000000  ...     464.000000     464.000000
mean    9798.265086    459.816810  ...    2547.062500    1047.952586
std    11606.252949    899.529442  ...    3035.909677    1253.808774
min        0.000000      0.000000  ...       0.000000       0.000000
25%     3510.500000     81.750000  ...     920.500000     352.750000
50%     6316.500000    187.000000  ...    1616.500000     666.500000
75%    11329.250000    407.250000  ...    3039.000000    1332.250000
max    96424.000000  10162.000000  ...   25990.000000   10804.000000

        late_exits    net_total    net_early   net_am_peak   net_midday  \
count   464.000000   464.000000   464.000000    464.000000   464.000000
mean    329.702586    49.702586   322.012931   -144.392241   154.834052
std     428.776820  1322.326500  1081.694621   2130.793163   729.868419
min       0.000000 -9736.000000 -8803.000000 -14703.000000 -4809.000000
25%      96.000000  -238.250000   -19.250000   -356.250000   -26.250000
50%     200.000000    86.500000   165.000000    181.000000   156.500000
75%     388.500000   441.750000   500.500000    693.500000   432.000000
max    3932.000000  7972.000000  5952.000000   6492.000000  4325.000000

        net_pm_peak  net_evening     net_late
count    464.000000   464.000000   464.000000
mean     -56.745690  -157.926724   -68.318966
std     1885.719707   745.692531   268.167289
min    -7512.000000 -3492.000000 -1718.000000
25%     -783.500000  -417.000000  -139.500000
50%     -190.000000  -131.500000   -40.500000
75%      326.750000    41.250000    13.000000
max    10176.000000  4673.000000  1135.000000

[8 rows x 22 columns]

2022 Entries and Exits by Day Summary:
              nlc  total_mon_entries  total_tue_wed_thur_entries  \
count   471.000000         471.000000                  471.000000
mean   2455.645435        8943.774947                10177.870488
std    3032.124956       13881.923344                16401.902020
min     500.000000           0.000000                    0.000000
25%     618.500000        2520.500000                 2743.500000
50%     747.000000        4962.000000                 5391.000000
75%    3173.000000        8970.000000                10009.500000
```

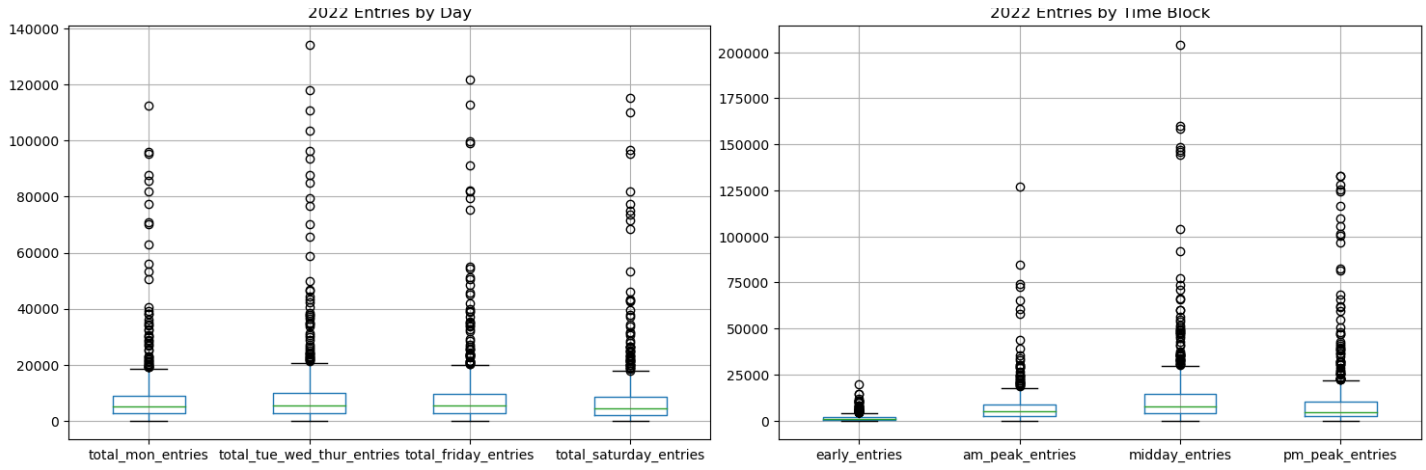
```
75%    3173.000000        8970.000000                 10009.500000
max    9846.000000      112656.000000                134435.000000

       total_friday_entries  total_saturday_entries  total_sunday_entries  \
count            471.000000              471.000000            471.000000
mean            9542.649682             8490.292994           5857.481953
std            14991.974632            14179.190509          10199.568397
min                0.000000                0.000000              0.000000
25%             2497.500000             2080.000000           1366.000000
50%             5234.000000             4419.000000           2971.000000
75%             9533.500000             8363.000000           5384.500000
max           121730.000000           115331.000000          88440.000000

       total_mon_thur_entries  total_mon_exits  total_tue_wed_thur_exits  \
count              471.000000       471.000000                471.000000
mean             19121.645435      8957.653928              10196.711253
std              30238.000535     14663.864508              17108.150220
min                  0.000000         0.000000                  0.000000
25%               5421.000000      2549.000000               2786.500000
50%              10358.000000      4734.000000               5239.000000
75%              18797.000000      9052.500000              10008.500000
max             247091.000000    128949.000000             152288.000000

       total_friday_exits  total_saturday_exits  total_sunday_exits  \
count          471.000000            471.000000          471.000000
mean          9559.152866           8512.800425         5871.397028
std          15826.013643          15092.128067        10728.685754
min              0.000000              0.000000            0.000000
25%           2477.000000           2015.000000         1353.500000
50%           4958.000000           3993.000000         2841.000000
75%           9432.500000           7975.000000         5601.500000
max         137675.000000         119643.000000        98708.000000

       total_mon_thur_exits  net_mon_to_thur    net_friday  net_saturday  \
count             471.00000       471.000000    471.000000    471.000000
mean            19154.36518       -32.719745    -16.503185    -22.507431
std             31730.67533      2983.067821   1533.154043   1875.886665
min                 0.00000    -34146.000000 -15945.000000 -20473.000000
25%              5336.00000      -237.500000   -108.500000     -5.500000
50%             10024.00000       141.000000     92.000000    186.000000
75%             18982.50000       736.500000    431.000000    509.500000
max            281237.00000     10398.000000   5712.000000   7641.000000

         net_sunday
count    471.000000
mean     -13.915074
std     1295.891710
min   -17905.000000
25%      -60.500000
50%       32.000000
75%      221.000000
max     5096.000000

2022 Entries and Exits by Time Block Summary:
               nlc  total_entries  early_entries  am_peak_entries  \
count   471.000000     471.000000     471.000000       471.000000
mean   2455.645435   43012.070064    1690.494692      7674.409766
std    3032.124956   68708.987439    2298.890495     11031.734208
min     500.000000       0.000000       0.000000         0.000000
25%     618.500000   11393.000000     450.000000      2533.500000
50%     747.000000   23253.000000     908.000000      4826.000000
75%    3173.000000   41896.000000    1949.000000      8910.000000
max    9846.000000  522764.000000   19960.000000    127220.000000

       midday_entries  pm_peak_entries  evening_entries  late_entries  \
count      471.000000       471.000000       471.000000    471.000000
mean     14396.711253     11210.212314      5374.904459   2665.464968
std      22737.920591     20112.650556     11113.410543   6614.170009
min          0.000000         0.000000         0.000000      0.000000
25%       4003.500000      2312.000000       826.000000    226.500000
50%       7892.000000      4741.000000      1907.000000    627.000000
75%      14385.500000     10232.500000      4463.500000   2028.000000
max     204292.000000    133026.000000     87014.000000  60127.000000
```

```
max      204292.000000     135020.000000     87014.000000     80127.000000

          total_exits   early_exits  ...  pm_peak_exits  evening_exits  \
count      471.000000    471.000000  ...     471.000000     471.000000
mean     43097.715499    983.065817  ...   11124.407643    6046.462845
std      72312.439957   2108.655679  ...   18537.145777    9112.163838
min          0.000000      0.000000  ...       0.000000       0.000000
25%      11361.500000    128.500000  ...    3174.000000    1750.000000
50%      22063.000000    349.000000  ...    5955.000000    3526.000000
75%      41623.000000    887.000000  ...   11055.500000    6227.000000
max     576315.000000  17697.000000  ...  185967.000000   89679.000000

          late_exits     net_total     net_early   net_am_peak    net_midday  \
count     471.000000    471.000000    471.000000    471.000000    471.000000
mean     3187.407643    -85.645435    707.428875     58.186837    256.426752
std      4779.475828   6932.751127   2148.728922  11006.404133   8918.721113
min         0.000000 -60362.000000 -11296.000000 -89560.000000 -96409.000000
25%       876.500000   -398.000000      0.000000   -182.000000      9.000000
50%      1800.000000    463.000000    427.000000   1315.000000   1327.000000
75%      3457.000000   1827.500000   1163.000000   3699.000000   2748.500000
max     50310.000000  26050.000000  11057.000000  49334.000000  22822.000000

         net_pm_peak   net_evening      net_late
count     471.000000    471.000000    471.000000
mean       85.804671   -671.558386   -521.942675
std      8199.687481   5752.897271   5076.624972
min    -60288.000000 -31946.000000 -17584.000000
25%     -2261.500000  -2575.000000  -1947.000000
50%      -663.000000  -1058.000000   -787.000000
75%       326.500000      0.000000     -0.500000
max     70216.000000  46821.000000  49835.000000

[8 rows x 22 columns]
```

In [28]:

```
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))
plt.tight_layout(pad=2.0)

EE_2020_by_day[['total_mon_to_thur_entries', 'total_friday_entries', 'total_saturday_entries', 'total_sunday_entries']].boxplot(ax=axes[0, 0])
axes[0, 0].set_title('2020 Entries by Day')

EE_2020_by_timeblock[['early_entries', 'am_peak_entries', 'midday_entries', 'pm_peak_entries']].boxplot(ax=axes[0, 1])
axes[0, 1].set_title('2020 Entries by Time Block')

EE_2022_by_day[['total_mon_entries', 'total_tue_wed_thur_entries', 'total_friday_entries', 'total_saturday_entries']].boxplot(ax=axes[1, 0])
axes[1, 0].set_title('2022 Entries by Day')

EE_2022_by_timeblock[['early_entries', 'am_peak_entries', 'midday_entries', 'pm_peak_entries']].boxplot(ax=axes[1, 1])
axes[1, 1].set_title('2022 Entries by Time Block')

plt.show()
```





Subplots showing the distribution of each dataset for each set of days and time blocks

In [30]:

```
time_blocks = ['early_entries', 'am_peak_entries', 'midday_entries', 'pm_peak_entries',
'evening_entries', 'late_entries']
time_blocks = ['early_entries', 'am_peak_entries', 'midday_entries', 'pm_peak_entries',
'evening_entries', 'late_entries']

averages_2020 = EE_2020_by_timeblock[time_blocks].mean()
averages_2022 = EE_2022_by_timeblock[time_blocks].mean()

percentage_changes = ((averages_2022 - averages_2020) / averages_2020) * 100
percentage_changes_df = pd.DataFrame({
    'Time Block': time_blocks,
    'Percentage Change': percentage_changes.values
})

percentage_changes_df.to_csv('percentage_changes.csv', index=False)
```

In [31]:

```
pip install circlify
```

```
Requirement already satisfied: circlify in c:\users\dimin\anaconda3\lib\site-packages (0.
15.0)
Note: you may need to restart the kernel to use updated packages.
```

Circular Packing Chart

In [32]:

```
import circlify
df = pd.DataFrame({
    'Time Block': ['Early Entries', 'AM Peak Entries', 'Midday Entries', 'PM Peak Entries
', 'Evening Entries', 'Late Entries'],
    'Percentage Change': [116, 296, 313, 350, 504, 920]
})

circles = circlify.circlify(
    df['Percentage Change'].tolist(),
    show_enclosure=False,
    target_enclosure=circlify.Circle(x=0, y=0, r=1)
)
fig, ax = plt.subplots(figsize=(10,10))
ax.set_title('Percentage Change in each Time Block from 2020 to 2022')
ax.axis('off')

lim = max(
    max(
        abs(circle.x) + circle.r,
```

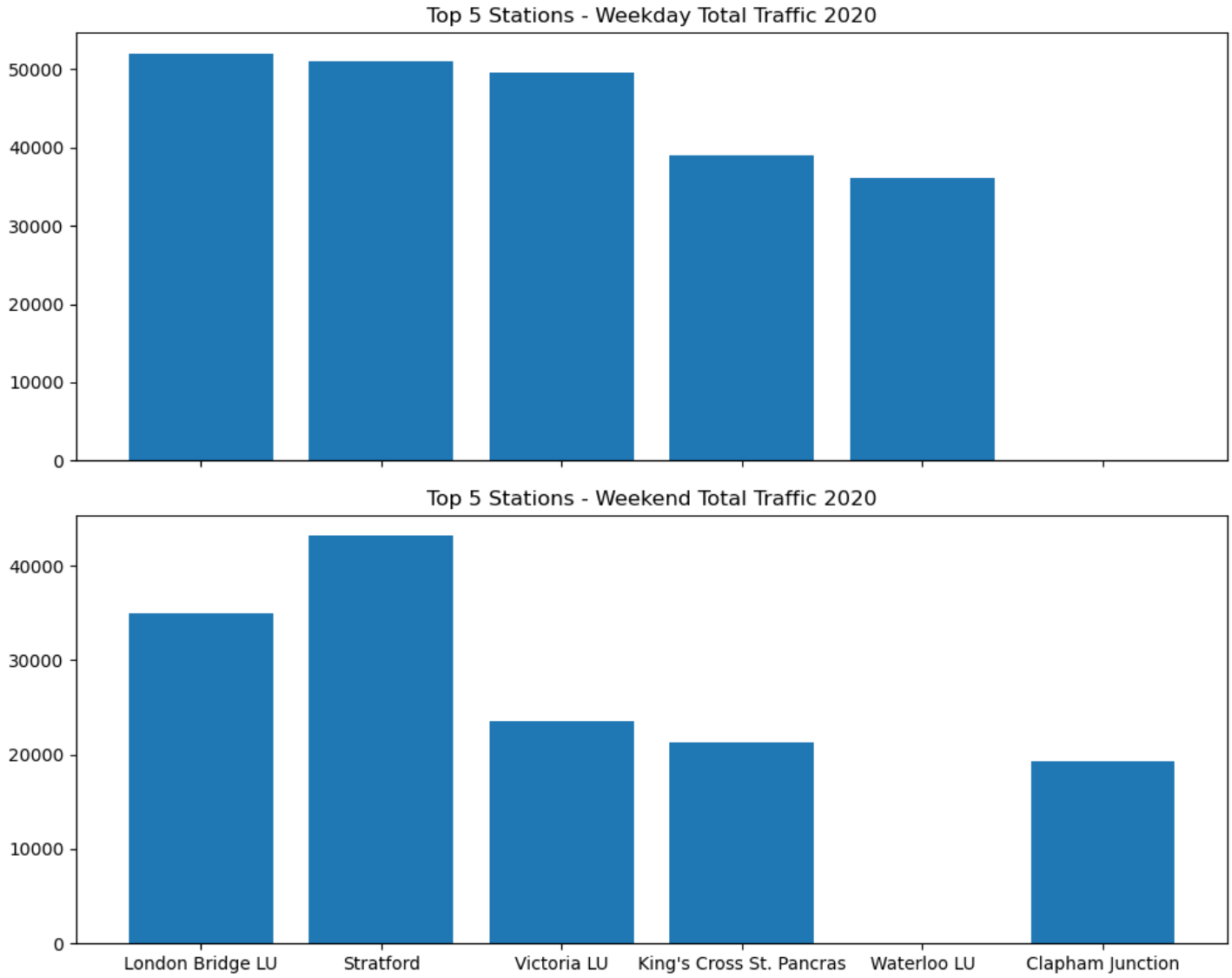
```
        abs(circle.y) + circle.r,
    )
    for circle in circles
)
plt.xlim(-lim, lim)
plt.ylim(-lim, lim)


labels = df['Time Block']
for circle, label in zip(circles, labels):
    x, y, r = circle
    ax.add_patch(plt.Circle((x, y), r*0.9, alpha=0.98, linewidth=2, facecolor="#e48c5c",
edgecolor="None"))
    plt.annotate(label, (x,y ) ,va='center', ha='center', bbox=dict(facecolor='white', e
dgecolor='None', boxstyle='round', pad=.5))
```

Bar Chart for Weekdays and Weekends Traffic

In [34]:

```
data_2020 = pd.read_csv('entries_exits_2020_by_day_NUMBAT.csv')
data_2022 = pd.read_csv('entries_exits_2022_by_day_NUMBAT.csv')
```

```
data_2020['weekday_total_2020'] = data_2020[['total_mon_to_thur_entries', 'total_friday_e
ntries']].sum(axis=1)
data_2020['weekend_total_2020'] = data_2020[['total_saturday_entries', 'total_sunday_entr
ies']].sum(axis=1)
data_2022['weekday_total_2022'] = data_2022[['total_mon_entries', 'total_tue_wed_thur_ent
ries', 'total_friday_entries']].sum(axis=1)
data_2022['weekend_total_2022'] = data_2022[['total_saturday_entries', 'total_sunday_entr
ies']].sum(axis=1)

top_5_weekday_2020 = data_2020.groupby('station')['weekday_total_2020'].sum().nlargest(5)
top_5_weekend_2020 = data_2020.groupby('station')['weekend_total_2020'].sum().nlargest(5)
top_5_weekday_2022 = data_2022.groupby('station')['weekday_total_2022'].sum().nlargest(5)
top_5_weekend_2022 = data_2022.groupby('station')['weekend_total_2022'].sum().nlargest(5)

fig, axs = plt.subplots(2, 1, figsize=(10, 8), sharex=True)

axs[0].bar(top_5_weekday_2020.index, top_5_weekday_2020.values)
axs[0].set_title('Top 5 Stations - Weekday Total Traffic 2020')
axs[0].tick_params(labelbottom=False)

axs[1].bar(top_5_weekend_2020.index, top_5_weekend_2020.values)
axs[1].set_title('Top 5 Stations - Weekend Total Traffic 2020')
axs[1].tick_params(labelrotation=0)

plt.tight_layout()
plt.show()
```

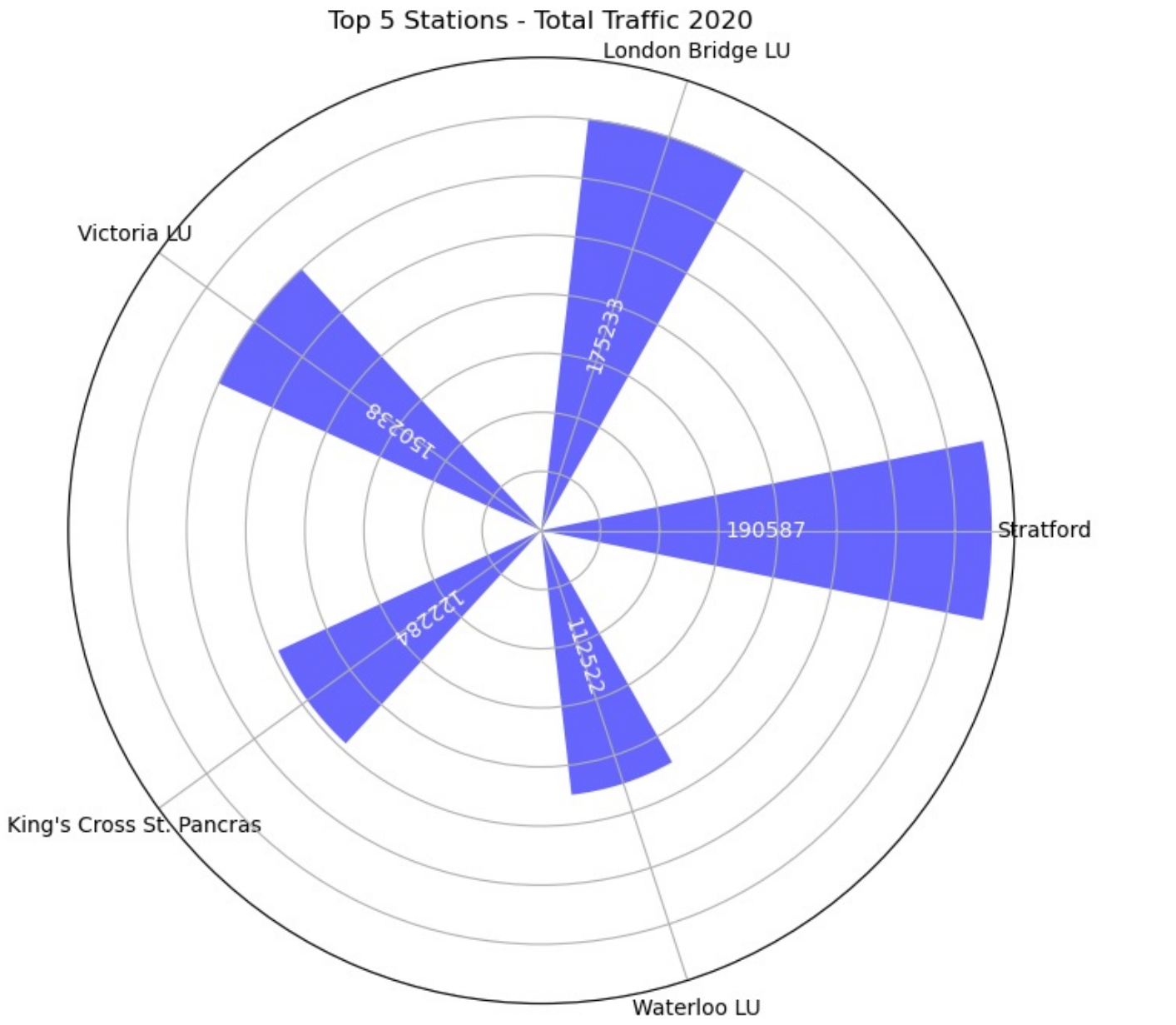


Radial Charts

In [37]:

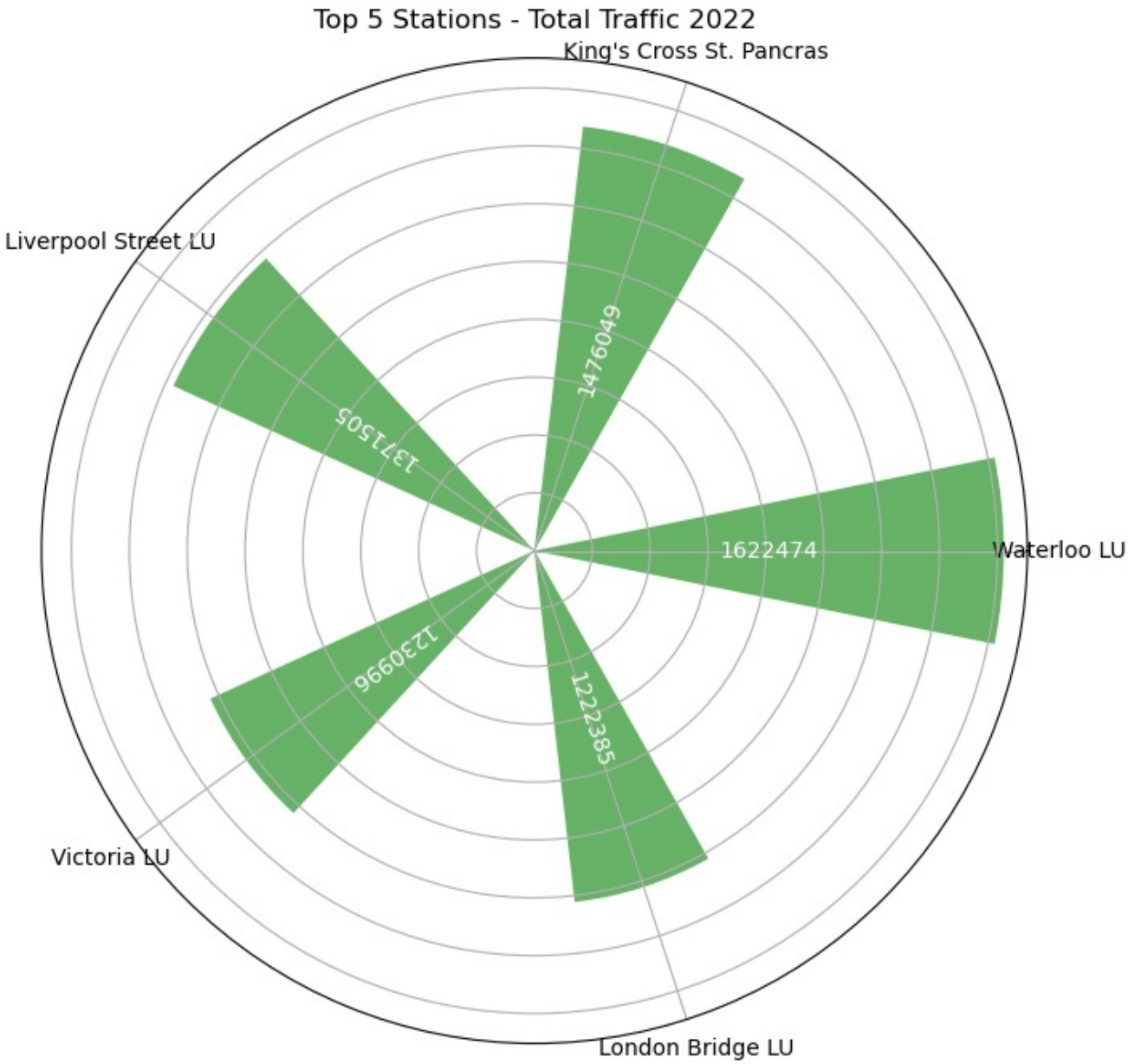
```
data_2020['total_traffic'] = data_2020.filter(like='_entries').sum(axis=1) + data_2020.f
```

```
ilter(like='_exits').sum(axis=1)
data_2022['total_traffic'] = data_2022.filter(like='_entries').sum(axis=1) + data_2022.f
ilter(like='_exits').sum(axis=1)

total_traffic_2020 = data_2020.groupby('station')['total_traffic'].sum()
total_traffic_2022 = data_2022.groupby('station')['total_traffic'].sum()

top_5_stations_2020 = total_traffic_2020.nlargest(5)
top_5_stations_2022 = total_traffic_2022.nlargest(5)

stations_2020 = top_5_stations_2020.index
values_2020 = top_5_stations_2020.values

angles = np.linspace(0, 2 * np.pi, len(stations_2020) + 1)
fig, ax = plt.subplots(figsize=(8, 8), subplot_kw=dict(polar=True))

for angle, value in zip(angles[:-1], values_2020):
    ax.bar(angle, value, color='blue', alpha=0.6, width=0.4)
    rotation = np.degrees(angle)
    ax.text(angle, value / 2, str(value), color='white', ha='center', va='center', rotat
ion=rotation)

ax.set_yticklabels([])
ax.set_xticks(angles[:-1])
ax.set_xticklabels(stations_2020)
ax.set_title('Top 5 Stations - Total Traffic 2020')

plt.show()
```



In [38]:

```
stations_2022 = top_5_stations_2022.index
values_2022 = top_5_stations_2022.values

angles = np.linspace(0, 2 * np.pi, len(stations_2022) + 1)
fig, ax = plt.subplots(figsize=(8, 8), subplot_kw=dict(polar=True))

for angle, value in zip(angles[:-1], values_2022):
    ax.bar(angle, value, color='green', alpha=0.6, width=0.4)
    rotation = np.degrees(angle)
    ax.text(angle, value / 2, str(value), color='white', ha='center', va='center', rotat
ion=rotation)

ax.set_yticklabels([])
ax.set_xticks(angles[:-1])
ax.set_xticklabels(stations_2022)
ax.set_title('Top 5 Stations - Total Traffic 2022')

plt.show()
```

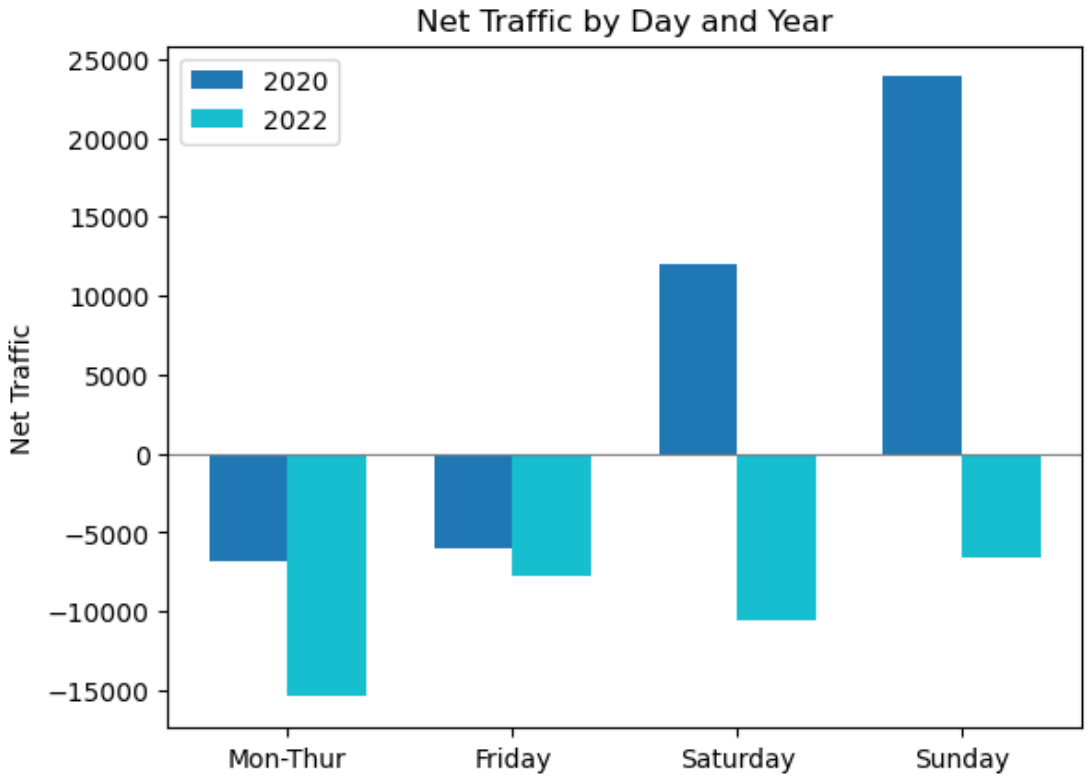


Bar Chart for Net Traffic by Day and Year

In [39]:

```
net_traffic_2020 = data_2020[['net_mon_to_thur', 'net_friday', 'net_saturday', 'net_sund
ay']].sum()
net_traffic_2022 = data_2022[['net_mon_to_thur', 'net_friday', 'net_saturday', 'net_sund
ay']].sum()
```

```
labels = ['Mon-Thur', 'Friday', 'Saturday', 'Sunday']
x = np.arange(len(labels))
width = 0.35

colors_2020 = ['#1f77b4']
colors_2022 = ['#17becf']

fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, net_traffic_2020, width, label='2020', color=colors_2020)
rects2 = ax.bar(x + width/2, net_traffic_2022, width, label='2022', color=colors_2022)

ax.axhline(0, color='grey', linewidth=0.8)
ax.set_ylabel('Net Traffic')
ax.set_title('Net Traffic by Day and Year')
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()

plt.show()
```



Radar Chart

In [103]:

```
data = {
    'Stratford': [4604, 14754, 36339, 25990, 10804, 3932],
    'London Bridge LU': [4130, 15631, 34885, 22778, 8398, 2376],
    'Victoria LU': [4394, 15745, 25118, 22246, 7714, 1934],
    'Waterloo LU': [4983, 12084, 17808, 15099, 5565, 1525],
    'Kings Cross St. Pancras': [4346, 13168, 23413, 13997, 5262, 1705]
}
time_blocks = ['early_exits', 'am_peak_exits', 'midday_exits', 'pm_peak_exits', 'evening_
exits', 'late_exits']

df = pd.DataFrame(data, index=time_blocks)
num_vars = len(df)

angles = np.linspace(0, 2 * np.pi, num_vars, endpoint=False).tolist()
angles += angles[:1]
fig, ax = plt.subplots(figsize=(8, 8), subplot_kw=dict(polar=True))

for station in df.columns:
    values = df.loc[:, station].tolist()
```

```
    values += values[:1]
    ax.plot(angles, values, label=station)
    ax.fill(angles, values, alpha=0.25)

range_scale = df.values.flatten().max()
ax.set_xticks(angles[:-1])
ax.set_xticklabels(time_blocks)
ax.set_ylim(0, range_scale + (range_scale * 0.1))
ax.legend(loc='upper center', bbox_to_anchor=(0.5, -0.05), ncol=len(df.columns))
ax.set_title('Traffic Volume by Time Block for Key Stations', pad=20)

plt.show()
```