In [3]:

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import plotly.graph_objects as go
import plotly.express as px
import seaborn as sns
%matplotlib inline
```

In [4]:

```
air_df = pd.read_csv('air_traffic_data.csv' )
air_df
```

Out[4]:

	Activity Period	Operating Airline	Operating Airline IATA Code	Published Airline	Published Airline IATA Code	GEO Summary	GEO Region	Activity Type Code	Price Category Code	Terminal
0	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Deplaned	Low Fare	Terminal 1
1	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Enplaned	Low Fare	Terminal 1
2	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Thru / Transit	Low Fare	Terminal 1
3	200507	Air Canada	AC	Air Canada	AC	International	Canada	Deplaned	Other	Terminal 1
4	200507	Air Canada	AC	Air Canada	AC	International	Canada	Enplaned	Other	Terminal 1
...	...	...	...	...	...	...	...	...	...	..
15002	201603	Virgin America	VX	Virgin America	VX	Domestic	US	Enplaned	Low Fare	Terminal 2
15003	201603	Virgin America	VX	Virgin America	VX	International	Mexico	Deplaned	Low Fare	Internationa
15004	201603	Virgin America	VX	Virgin America	VX	International	Mexico	Enplaned	Low Fare	Terminal 2
15005	201603	Virgin Atlantic	VS	Virgin Atlantic	VS	International	Europe	Deplaned	Other	Internationa
15006	201603	Virgin Atlantic	VS	Virgin Atlantic	VS	International	Europe	Enplaned	Other	Internationa

15007 rows × 16 columns

In [5]:

```
air_df.shape
```

Out[5]:

```
(15007, 16)
```

In [6]:

```
air_df.describe()
```

Out[6]:

	Activity Period	Passenger Count	Adjusted Passenger Count	Year

	Activity Period	Passenger Count	Adjusted Passenger Count	Year
count	15007.000000	15007.000000	15007.000000	15007.000000
mean	201045.073366	29240.521090	29331.917105	2010.385220
std	313.336196	58319.509284	58284.182219	3.137589
min	200507.000000	1.000000	1.000000	2005.000000
25%	200803.000000	5373.500000	5495.500000	2008.000000
50%	201011.000000	9210.000000	9354.000000	2010.000000
75%	201308.000000	21158.500000	21182.000000	2013.000000
max	201603.000000	659837.000000	659837.000000	2016.000000

In [7]:
```
air_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15007 entries, 0 to 15006
Data columns (total 16 columns):
 #   Column                        Non-Null Count  Dtype 
---  ------                        --------------  ----- 
 0   Activity Period               15007 non-null  int64 
 1   Operating Airline             15007 non-null  object
 2   Operating Airline IATA Code   14953 non-null  object
 3   Published Airline             15007 non-null  object
 4   Published Airline IATA Code   14953 non-null  object
 5   GEO Summary                   15007 non-null  object
 6   GEO Region                    15007 non-null  object
 7   Activity Type Code            15007 non-null  object
 8   Price Category Code           15007 non-null  object
 9   Terminal                      15007 non-null  object
 10  Boarding Area                 15007 non-null  object
 11  Passenger Count               15007 non-null  int64 
 12  Adjusted Activity Type Code   15007 non-null  object
 13  Adjusted Passenger Count      15007 non-null  int64 
 14  Year                          15007 non-null  int64 
 15  Month                         15007 non-null  object
dtypes: int64(4), object(12)
memory usage: 1.8+ MB
```

In [8]:
```
air_df.drop_duplicates()
```

Out[8]:

	Activity Period	Operating Airline	Operating Airline IATA Code	Published Airline	Published Airline IATA Code	GEO Summary	GEO Region	Activity Type Code	Price Category Code	Terminal
0	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Deplaned	Low Fare	Terminal 1
1	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Enplaned	Low Fare	Terminal 1
2	200507	ATA Airlines	TZ	ATA Airlines	TZ	Domestic	US	Thru / Transit	Low Fare	Terminal 1
3	200507	Air Canada	AC	Air Canada	AC	International	Canada	Deplaned	Other	Terminal 1

	Activity Period	Operating Airline	Operating Airline IATA Code	Published Airline	Published Airline IATA Code	GEO Summary	GEO Region	Activity Type Code	Price Category Code	Terminal
4	200507	Air Canada	AC	Air Canada	AC	International	Canada	Enplaned	Other	Terminal 1
...	...	...	...	...	...	...	...	...	...	..
15002	201603	Virgin America	VX	Virgin America	VX	Domestic	US	Enplaned	Low Fare	Terminal 2
15003	201603	Virgin America	VX	Virgin America	VX	International	Mexico	Deplaned	Low Fare	Internationa
15004	201603	Virgin America	VX	Virgin America	VX	International	Mexico	Enplaned	Low Fare	Terminal 2
15005	201603	Virgin Atlantic	VS	Virgin Atlantic	VS	International	Europe	Deplaned	Other	Internationa
15006	201603	Virgin Atlantic	VS	Virgin Atlantic	VS	International	Europe	Enplaned	Other	Internationa

15007 rows × 16 columns

In [9]:
```
#find missing values

print(air_df.isnull())
```

```
       Activity Period  Operating Airline  Operating Airline IATA Code  \
0                False              False                        False
1                False              False                        False
2                False              False                        False
3                False              False                        False
4                False              False                        False
...                ...                ...                          ...
15002            False              False                        False
15003            False              False                        False
15004            False              False                        False
15005            False              False                        False
15006            False              False                        False

       Published Airline  Published Airline IATA Code  GEO Summary  \
0                  False                        False        False
1                  False                        False        False
2                  False                        False        False
3                  False                        False        False
4                  False                        False        False
...                  ...                          ...          ...
15002              False                        False        False
15003              False                        False        False
15004              False                        False        False
15005              False                        False        False
15006              False                        False        False

       GEO Region  Activity Type Code  Price Category Code  Terminal  \
0           False               False                False     False
1           False               False                False     False
2           False               False                False     False
3           False               False                False     False
4           False               False                False     False
...           ...                 ...                  ...       ...
15002       False               False                False     False
```

```
15003       False                False                False     False
15004       False                False                False     False
15005       False                False                False     False
15006       False                False                False     False

       Boarding Area  Passenger Count  Adjusted Activity Type Code  \
0              False            False                         False
1              False            False                         False
2              False            False                         False
3              False            False                         False
4              False            False                         False
...              ...              ...                           ...
15002          False            False                         False
15003          False            False                         False
15004          False            False                         False
15005          False            False                         False
15006          False            False                         False

       Adjusted Passenger Count   Year  Month
0                         False  False  False
1                         False  False  False
2                         False  False  False
3                         False  False  False
4                         False  False  False
...                         ...    ...    ...
15002                     False  False  False
15003                     False  False  False
15004                     False  False  False
15005                     False  False  False
15006                     False  False  False

[15007 rows x 16 columns]
```

In [11]:

```
#drop missing values
air_df.dropna(inplace=True)
air_df.shape
```

Out[11]:

```
(14953, 16)
```

In [13]:

```
y=air_df['Operating Airline'].value_counts()
y
```

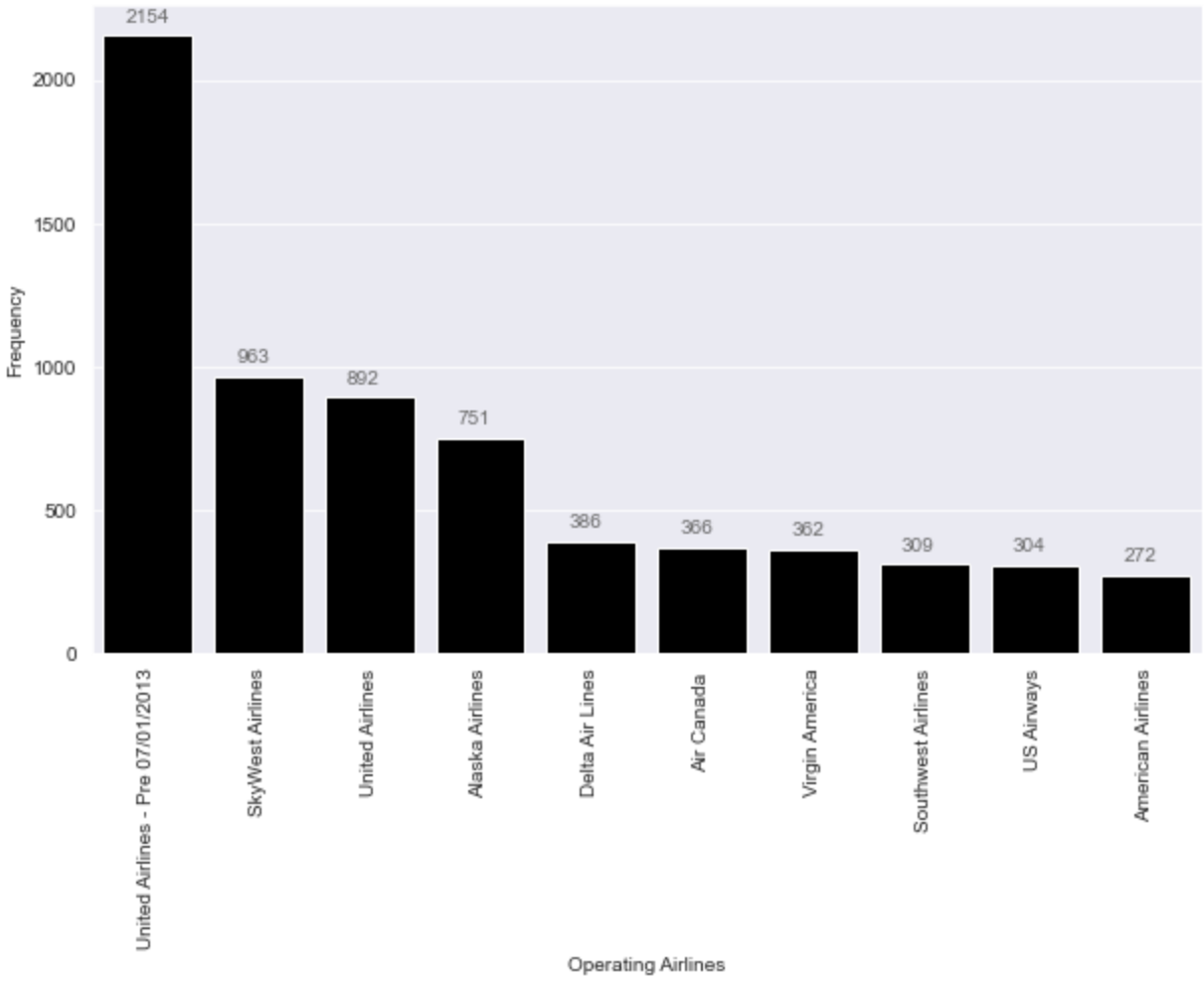
Out[13]:

```
United Airlines - Pre 07/01/2013    2154
SkyWest Airlines                     963
United Airlines                      892
Alaska Airlines                      751
Delta Air Lines                      386
                                    ...
Air India Limited                      8
World Airways                          3
Evergreen International Airlines       2
Atlas Air, Inc                         2
Xtra Airways                           2
Name: Operating Airline, Length: 73, dtype: int64
```

In [79]:

```
#United Airlines 07/01/2013 most popular choice of flight
plt.figure(figsize=(10,6))
sns.set_style('darkgrid')
ax = sns.countplot(x='Operating Airline',data=air_df, order=air_df['Operating Airline'].va
plt.xticks(rotation=90)
plt.xlabel('Operating Airlines')
plt.ylabel('Frequency')
```

```
for p in ax.patches:
        ax.annotate((p.get_height()), (p.get_x()+0.2, p.get_height()+50),color='dimgrey')
```

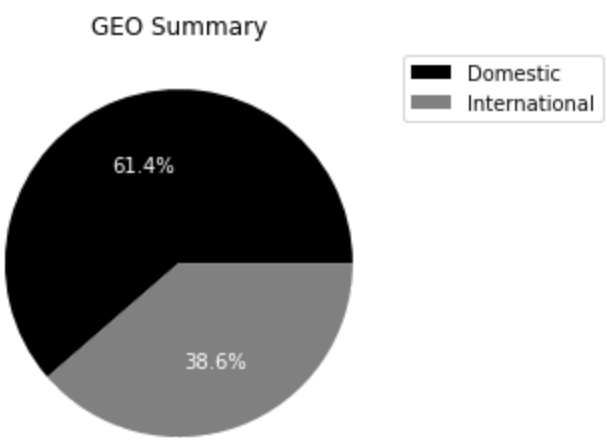


In [26]:

```
#Domestic travellers were more
y=air_df['GEO Summary'].value_counts()
y
mylabels = ["Domestic", "International"]
mycolors = ["black", "grey"]
fig1, ax1 = plt.subplots()
wedges, texts, autotexts = ax1.pie(y, labels = mylabels, autopct='%1.1f%%',radius=1,colors
ax1.set_title("GEO Summary")
ax1.legend(loc ="upper right",bbox_to_anchor=(1, 0, 0.5, 1))
```

Out[26]:

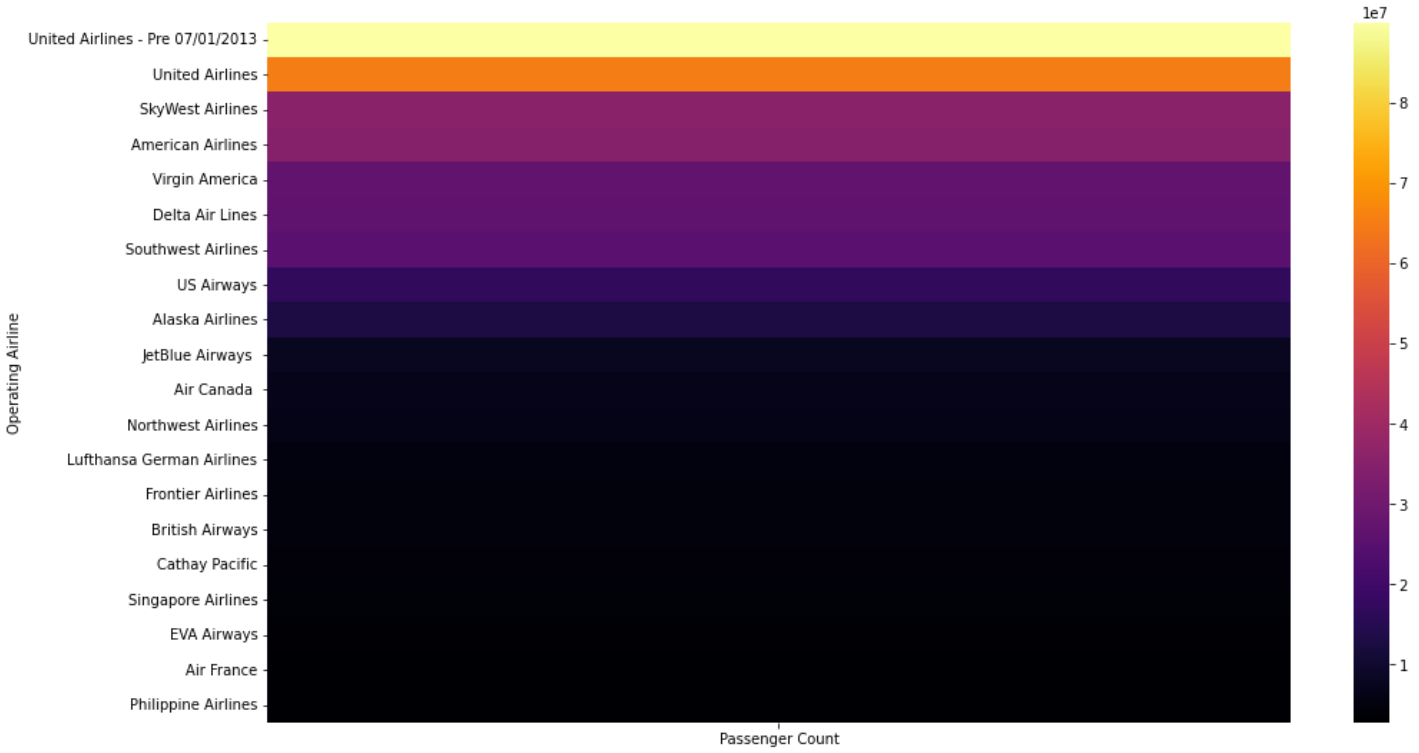
```
<matplotlib.legend.Legend at 0x197dc455190>
```



In [40]:
```
#Most number of passengers took United Airlines - Pre 07/01/2013 flight
plt.figure(figsize=(16,9))
selected_columns = air_df[["Operating Airline","Passenger Count"]]
new_df = selected_columns.copy()
h=new_df.groupby('Operating Airline').sum()
q=h.sort_values('Passenger Count', ascending=False)
print(q.head(20))
w=new_df.set_index('Operating Airline')
sns.heatmap(data=q.head(20), cmap="inferno", robust = True)
```

```
                                Passenger Count
Operating Airline
United Airlines - Pre 07/01/2013        105363917
United Airlines                          64876996
SkyWest Airlines                         35711737
American Airlines                        34588714
Virgin America                           26934738
Delta Air Lines                          26440420
Southwest Airlines                       25087141
US Airways                               16816616
Alaska Airlines                          12955980
JetBlue Airways                           7827973
Air Canada                                6680071
Northwest Airlines                        6266220
Lufthansa German Airlines                 4979907
Frontier Airlines                         4624796
British Airways                           4547282
Cathay Pacific                            4417302
Singapore Airlines                        3804635
EVA Airways                               3384020
Air France                                2989982
Philippine Airlines                       2644148
<AxesSubplot:ylabel='Operating Airline'>
```

Out[40]:

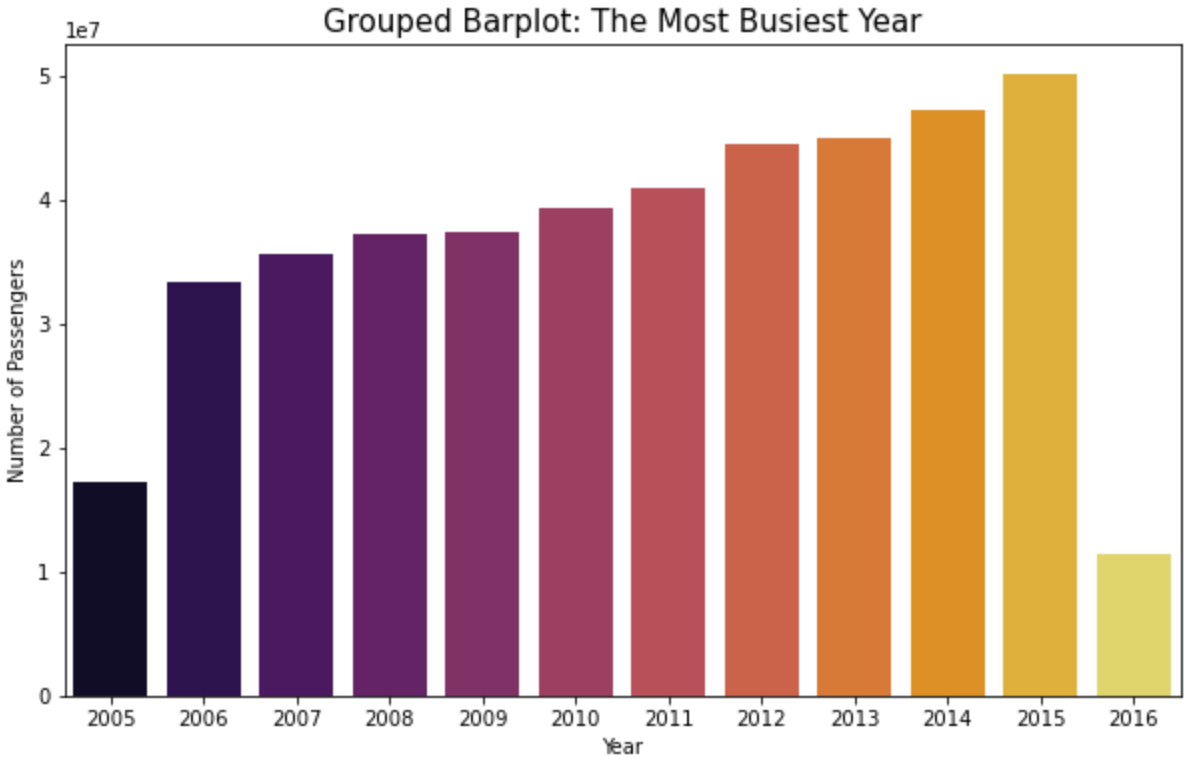


In [ ]:
```
#Which year was the busiest and showed highest number of passengers?
#which month was the busiest and showed highest number of passengers?
#which is the busiest time of year for airlines?
#GEO SUummary vs GEO Region vs Operating Airline
#Operating Airline vs Boarding Area
#Operating Airline vs Price Category
```

In [74]:
```
#In 2015 the passenger count was high
o=air_df.groupby('Year')['Passenger Count'].sum().reset_index().rename(columns={'index':'Y
print(o.head(15))
plt.figure(figsize=(10,6))
ax=sns.barplot(x='Passen', y='Passenger Count', data=o,palette = "inferno")
plt.ylabel("Number of Passengers", size=10)
plt.xlabel("Year", size=10)
plt.title("Grouped Barplot: The Most Busiest Year", size=15)
```

```
    Passen  Passenger Count
0     2005         17222033
1     2006         33332970
2     2007         35554082
3     2008         37234678
4     2009         37338942
5     2010         39253999
6     2011         40927786
7     2012         44399885
8     2013         44945760
9     2014         47114631
10    2015         50057887
11    2016         11429847
```

Out[74]:
```
Text(0.5, 1.0, 'Grouped Barplot: The Most Busiest Year')
```

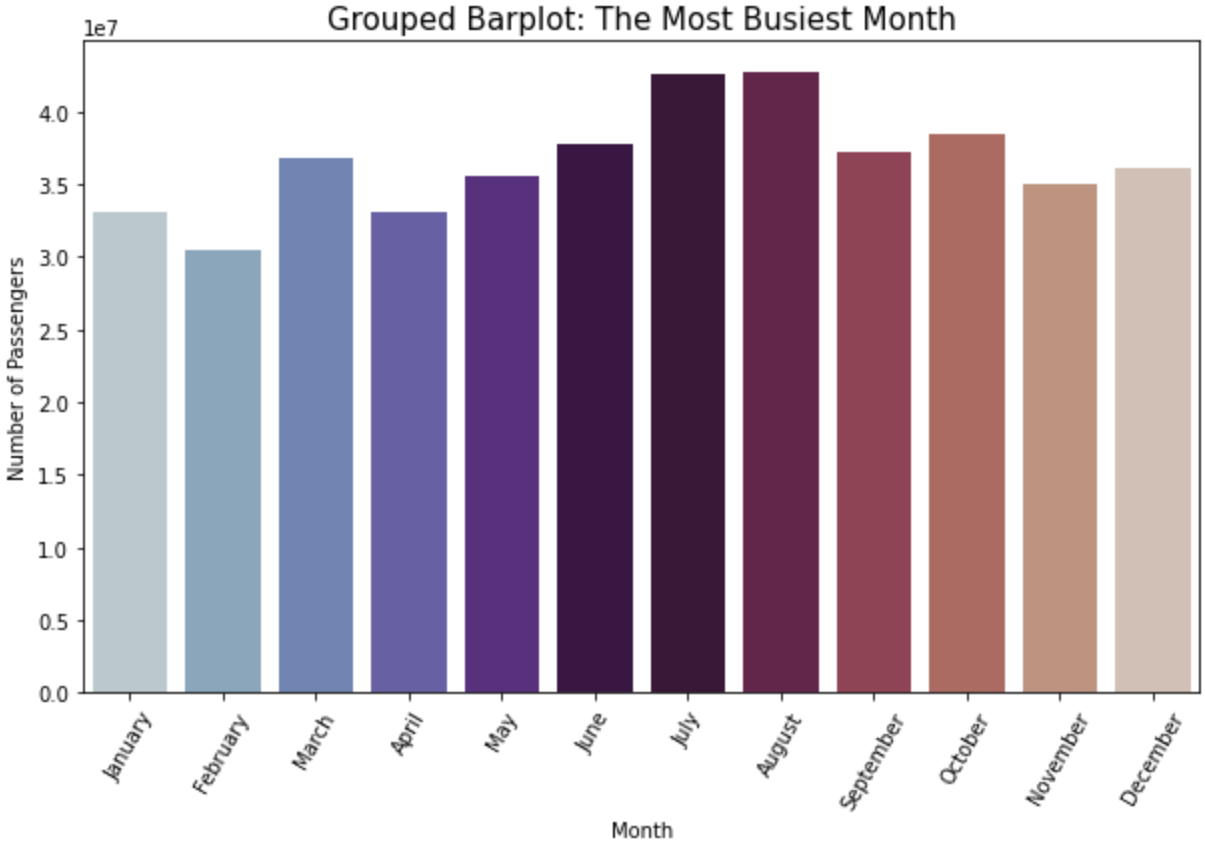


In [76]:
```
#August is the busiest Month
o=air_df.groupby('Month')['Passenger Count'].sum().reset_index().rename(columns={'index':'
print(o)
plt.figure(figsize=(10,6))
Months = ['January', 'February', 'March', 'April', 'May', 'June',
          'July', 'August', 'September', 'October', 'November', 'December']
ax=sns.barplot(x='Passen', y='Passenger Count', data=o,palette = "twilight", order=Months)
plt.ylabel("Number of Passengers", size=10)
plt.xlabel("Month", size=10)
plt.title("Grouped Barplot: The Most Busiest Month", size=15)
plt.xticks(rotation=60)
```

```
      Passen  Passenger Count
```

```
0       April          33106801
1      August          42753659
2    December          36162507
3    February          30444302
4     January          33087662
5        July          42552115
6        June          37721908
7       March          36787839
8         May          35558969
9    November          34988575
10    October          38391386
11  September          37256777
```

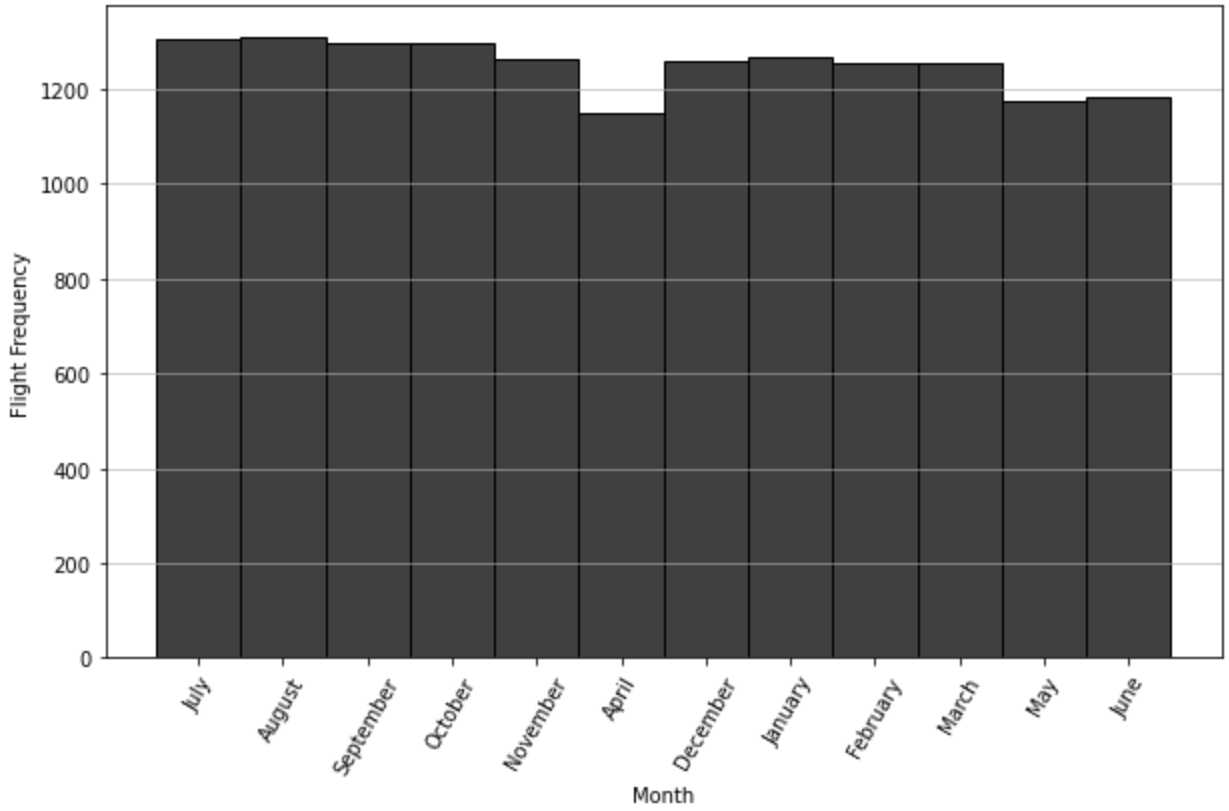
Out[76]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11]),
 [Text(0, 0, 'January'),
  Text(1, 0, 'February'),
  Text(2, 0, 'March'),
  Text(3, 0, 'April'),
  Text(4, 0, 'May'),
  Text(5, 0, 'June'),
  Text(6, 0, 'July'),
  Text(7, 0, 'August'),
  Text(8, 0, 'September'),
  Text(9, 0, 'October'),
  Text(10, 0, 'November'),
  Text(11, 0, 'December')])
```



In [78]:
```
#August is the most busiest month
plt.figure(figsize=(10,6))
Month=air_df['Month']
Month
sns.histplot(data=Month,color="black")
plt.xlabel('Month')
plt.ylabel('Flight Frequency')
plt.grid(axis='y', alpha=0.75)
plt.xticks(rotation=60)
```

Out[78]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
 [Text(0, 0, ''),
```

```
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, ''),
Text(0, 0, '')])
```



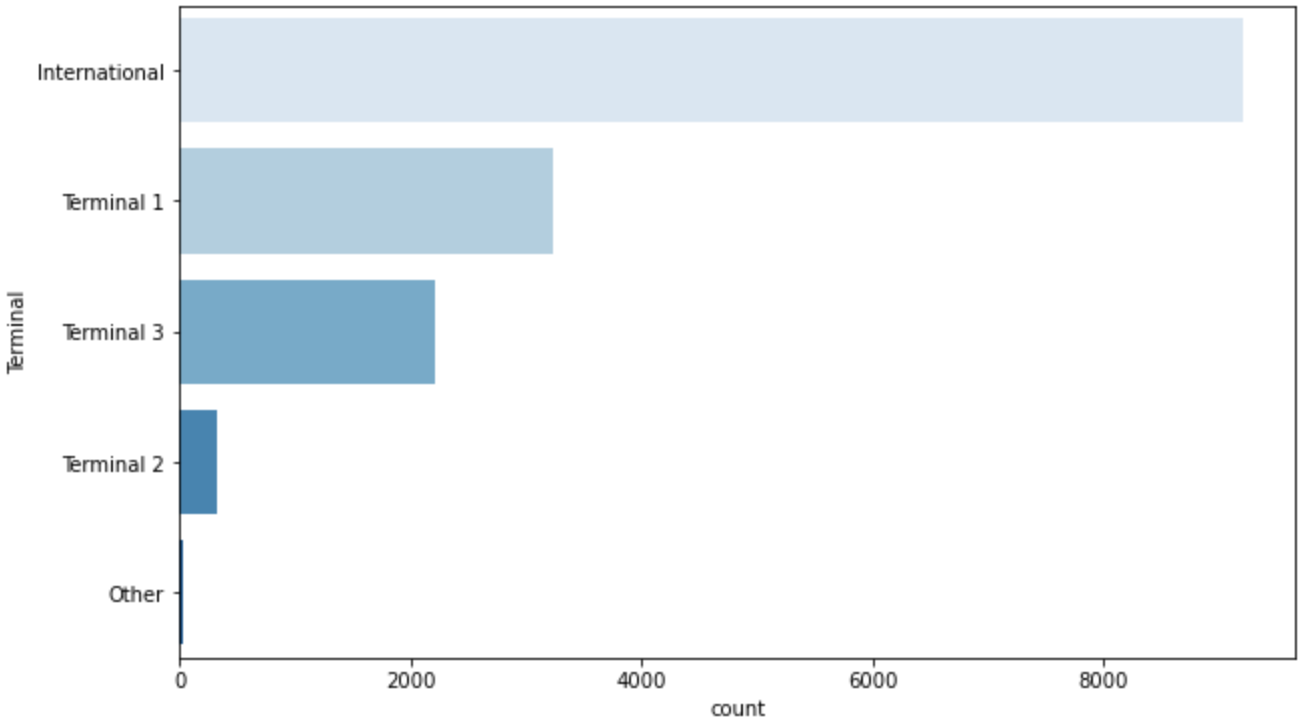
In [77]:

```
#Flights took off more frequently from the international terminal
Term =air_df['Terminal'].value_counts().reset_index().rename(columns={'index':'Terminal','
Term

plt.figure(figsize=(10,6))
sns.barplot(x='count', y='Terminal', data=Term, palette = "Blues")
```

Out[77]:

```
<AxesSubplot:xlabel='count', ylabel='Terminal'>
```



In [72]:

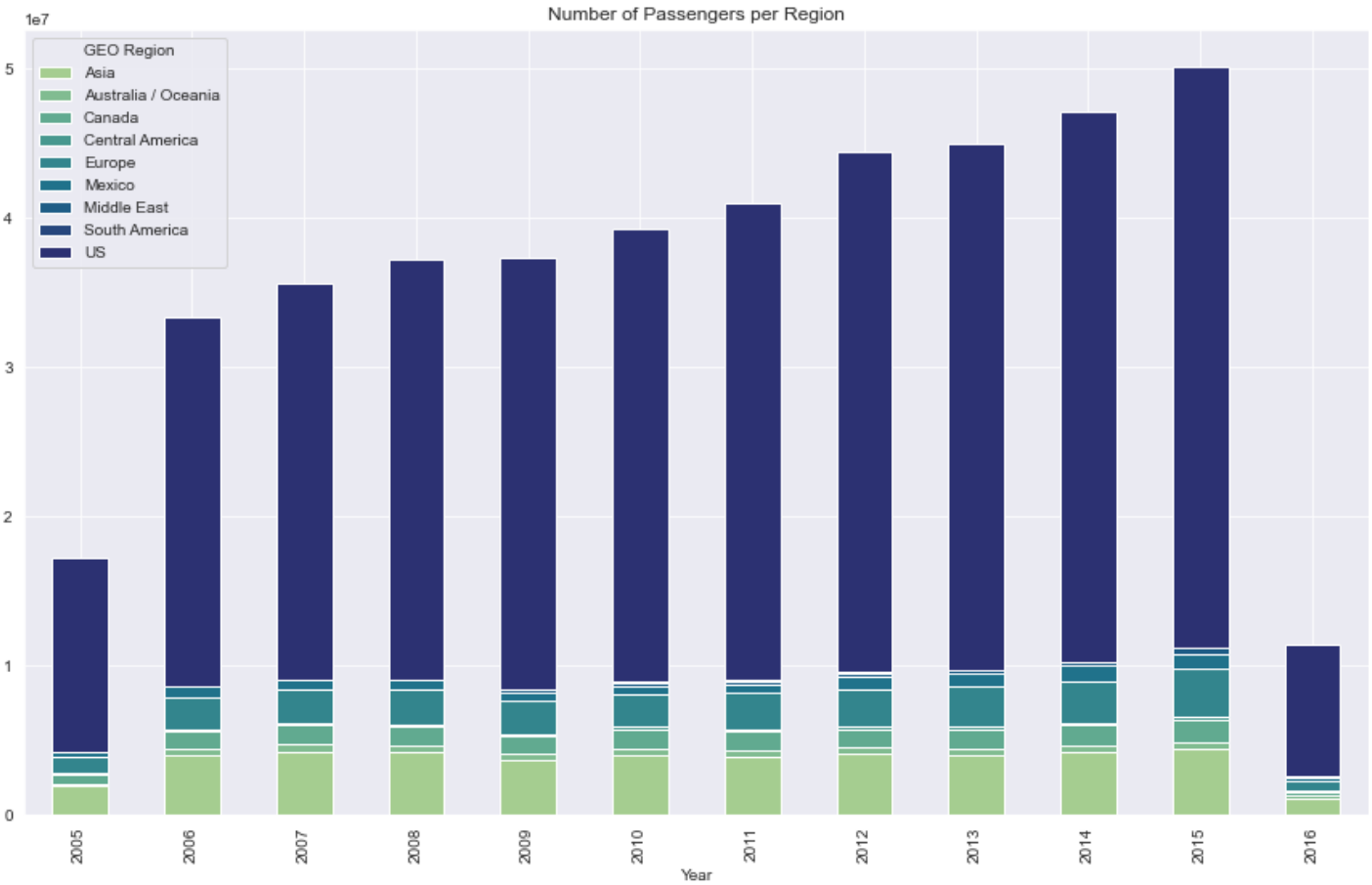
```
o=air_df.groupby(['Year', 'GEO Region']).agg(Passenger_Count = ('Passenger Count', 'sum'))
print(o.head(15))
```

```
                          Passenger_Count
Year GEO Region
2005 Asia                         1955732
     Australia / Oceania           164991
     Canada                        589467
     Central America                61994
     Europe                       1147195
     Mexico                        305087
     US                          12997567
2006 Asia                         3978377
     Australia / Oceania           453420
     Canada                       1194203
     Central America               111255
     Europe                       2163664
     Mexico                        686118
     US                          24745933
2007 Asia                         4207750
```

In [84]:

```
o=air_df.groupby(['Year', 'GEO Region']).agg(Passenger_Count = ('Passenger Count', 'sum'))
print(o)
```

```
AxesSubplot(0.125,0.125;0.775x0.755)
```



In [5]:
```
y=air_df['Operating Airline']
y
```

Out[5]:
```
0            ATA Airlines
1            ATA Airlines
2            ATA Airlines
3             Air Canada
4             Air Canada
               ...
15002     Virgin America
15003     Virgin America
15004     Virgin America
15005    Virgin Atlantic
15006    Virgin Atlantic
Name: Operating Airline, Length: 15007, dtype: object
```

In [ ]: