

Package Python

```
In [ ]: import pandas as pd
import numpy as np
import xarray as xr
import os as os
import seaborn as sns
import plotly.express as px
import matplotlib.pyplot as plt
%matplotlib inline
from pandas.plotting import autocorrelation_plot
from sklearn import linear_model
```

Settings Jupyter

```
In [ ]: # set the max columns to none
pd.set_option('display.max_columns', None)
```

Functions

```
In [ ]: def plotar(titulo: str, labelx: str, labely: str, x: str, y: str, dataset: dict, fontt: int, fontlx: int, fontly: int):
    """Função para plotagem de gráfico"""
    sns.set_palette('husl')
    sns.set_style('whitegrid')
    ax = sns.lineplot(x = x, y = y, data = dataset, hue=hue, style=style)
    ax.figure.set_size_inches(16, 12)
    ax.set_title(titulo, loc='left', fontsize=fontt)
    ax.set_xlabel(labelx, fontsize=fontlx)
    ax.set_ylabel(labely, fontsize=fontly)
    ax = ax
    return ax
```

```
In [ ]: def plot_comparacao(x, y1,y2,y3, y4=None, params=False, dataset=None, titulo=None):
    plt.figure(figsize=(16,12))
    if params:
        ax = plt.subplot(4,1,1)
        ax.set_title(titulo, fontsize=18, loc='left')
        sns.lineplot(x = x, y = y1, data = dataset)
        ax = plt.subplot(4,1,2)
        sns.lineplot(x = x, y = y2, data = dataset)
        ax = plt.subplot(4,1,3)
        sns.lineplot(x = x, y = y3, data = dataset)
        ax = plt.subplot(4,1,4)
        sns.lineplot(x = x, y = y4, data = dataset)
    else:
        ax = plt.subplot(3,1,1)
        ax.set_title(titulo, fontsize=18, loc='left')
        sns.lineplot(x = x, y = y1, data = dataset)
        ax = plt.subplot(3,1,2)
        sns.lineplot(x = x, y = y2, data = dataset)
        ax = plt.subplot(3,1,3)
        sns.lineplot(x = x, y = y3, data = dataset)
    ax = ax
    return ax
```

Objective of this notebook

This analyze aims to explore the ANAC database of flights from January 2022. It was used basic database for this study.

Reading of bases

```
In [ ]: _path = os.getcwd()
        _path
```

```
Out[ ]: 'c:\\git\\Gol_Cases'
```

```
In [ ]: _path1 = os.path.join(_path, 'data', 'basica2022-01.txt')
        _path1
```

```
Out[ ]: 'c:\\git\\Gol_Cases\\data\\basica2022-01.txt'
```

```
In [ ]: _df = pd.read_csv(_path1, sep=';', encoding='latin1')
        _df.shape
```

C:\Users\conta\AppData\Local\Programs\Python\Python39\lib\site-packages\IPython\core\interactiveshell.py:3441: DtypeWarning: Columns (8,10) have mixed types.Specify dtype option on import or set low_memory=False.

```
Out[ ]: exec(code_obj, self.user_global_ns, self.user_ns)
        (69597, 110)
```

- Cleaning not necessary data for analyze.

```
In [ ]: _df = _df.drop(columns=[
        'sg_empresa_icao',
        'sg_empresa_iata',
        'nr_singular',
        'id_di',
        'cd_di',
        'ds_di',
        'ds_grupo_di',
        'nr_ano_referencia',
        'nr_semestre_referencia',
        'nr_trimestre_referencia',
        'nr_ano_mes_referencia',
        'id_tipo_linha',
        'cd_tipo_linha',
        'ds_tipo_linha',
        'ds_natureza_tipo_linha',
        'ds_servico_tipo_linha',
        'nr_ano_partida_real',
        'nr_semestre_partida_real',
        'nm_semestre_partida_real',
        'nr_trimestre_partida_real',
        'nm_trimestre_partida_real',
        'nr_mes_partida_real',
        'nm_mes_partida_real',
        'nr_semana_partida_real',
        'nm_dia_semana_partida_real',
        'nr_ano_mes_partida_real',
        'sg_icao_origem',
        'sg_uf_origem',
```

```

'nm_regiao_origem',
'nr_ano_chegada_real',
'nr_semestre_chegada_real',
'nm_semestre_chegada_real',
'nr_trimestre_chegada_real',
'nm_trimestre_chegada_real',
'nr_mes_chegada_real',
'nm_mes_chegada_real',
'nr_ano_mes_chegada_real',
'ds_matricula',
'sg_uf_destino',
'nm_regiao_destino',
'nr_escala_destino',
'lt_combustivel',
'kg_bagagem_livre',
'kg_bagagem_excesso',
'nr_decolagem',
'id_arquivo',
'nm_arquivo',
'nr_linha',
'dt_sistema'])
_df.shape

```

Out[]: (69597, 61)

```
In [ ]: _df = _df.rename(columns={'nr_assentos_ofertados': 'pax_ofertados', 'nr_passag_pagos'
```

```
In [ ]: _df.dtypes
```

```

Out[ ]: id_basica          int64
id_empresa          int64
nm_empresa          object
nm_pais              object
ds_tipo_empresa      object
...
nr_bagagem_gratis_km int64
nr_ask               int64
rpk                  int64
nr_atk               int64
nr_rtk               int64
Length: 61, dtype: object

```

Verifying nan values in the columns.

```
In [ ]: _df.describe()
```

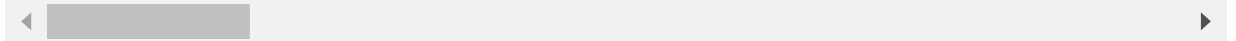
```

Out[ ]:

```

	id_basica	id_empresa	nr_voo	nr_mes_referencia	nr_semana_referencia	nr_dia_re
count	6.959700e+04	6.959700e+04	69597.000000	69597.0	69597.000000	69597
mean	2.264191e+07	1.000946e+06	3490.833944	1.0	3.571246	1.0
std	2.053645e+04	3.945483e+02	2255.404553	0.0	1.300965	0.0
min	2.260455e+07	1.000002e+06	2.000000	1.0	1.000000	1.0
25%	2.262400e+07	1.000854e+06	1802.000000	1.0	2.000000	1.0
50%	2.264201e+07	1.000909e+06	3178.000000	1.0	4.000000	1.0
75%	2.265981e+07	1.001093e+06	4432.000000	1.0	5.000000	2.0

	id_basica	id_empresa	nr_voo	nr_mes_referencia	nr_semana_referencia	nr_dia_re
max	2.267771e+07	1.007484e+06	9998.000000	1.0	6.000000	3



```
In [ ]: _counts_cols = _df.count().reset_index()
        _counts_cols.columns = ['columns', 'values']
        _counts_cols
```

```
Out[ ]:
```

	columns	values
0	id_basica	69597
1	id_empresa	69597
2	nm_empresa	69597
3	nm_pais	69597
4	ds_tipo_empresa	69597
...
56	nr_bagagem_gratis_km	69597
57	nr_ask	69597
58	rpk	69597
59	nr_atk	69597
60	nr_rtk	69597

61 rows × 2 columns

```
In [ ]: _counts_cols[_counts_cols['values']!=69597]
```

```
Out[ ]:
```

	columns	values
19	sg_iata_origem	69588
25	hr_chegada_real	69595
27	nr_semana_chegada_real	69596
28	nm_dia_semana_chegada_real	69596
29	nr_dia_chegada_real	69596
35	sg_iata_destino	69588
48	nr_horas_voadas	69595
50	nr_velocidade_media	69595

```
In [ ]: _df = _df.fillna(0)
```

Removing white spaces.

```
In [ ]: _cols = _df.select_dtypes(include='object')
        _cols = _cols.columns.to_list()
```

```
for i in _cols:
    _df[i] = _df[i].str.strip()
```

```
In [ ]: _df.loc[19].to_list()[7]
```

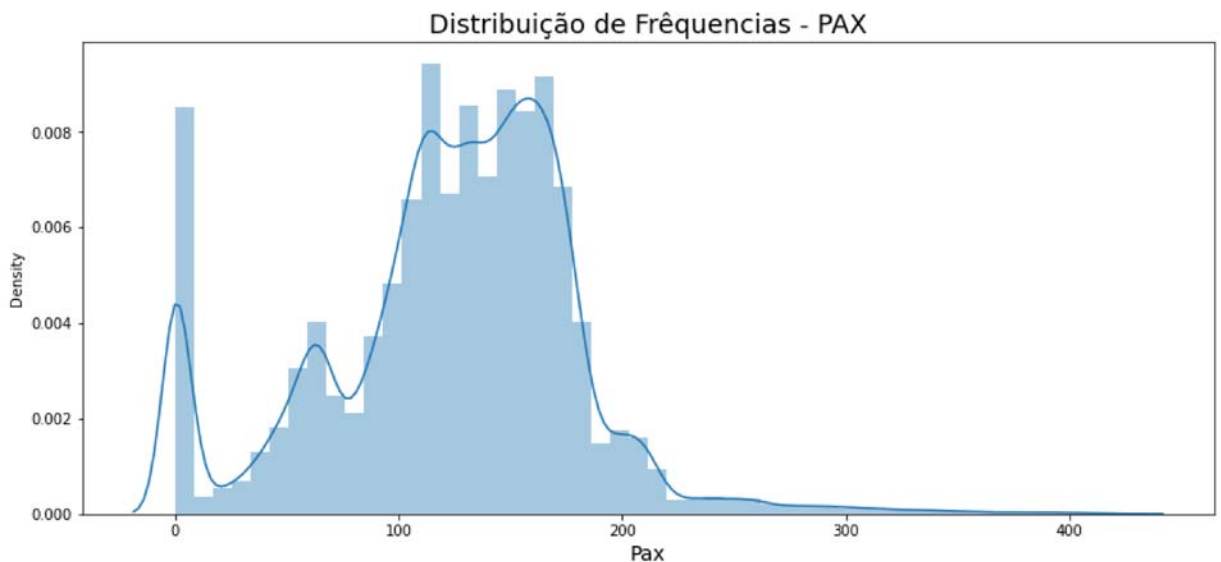
```
Out[ ]: '1º SEMESTRE'
```

```
In [ ]: histograma = sns.distplot(_df['pax'])
histograma.figure.set_size_inches(14,6)
histograma.set_title('Distribuição de Frequências - PAX', fontsize=18)
histograma.set_xlabel('Pax', fontsize=14)
```

C:\Users\conta\AppData\Local\Programs\Python\Python39\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

```
warnings.warn(msg, FutureWarning)
```

```
Out[ ]: Text(0.5, 0, 'Pax')
```

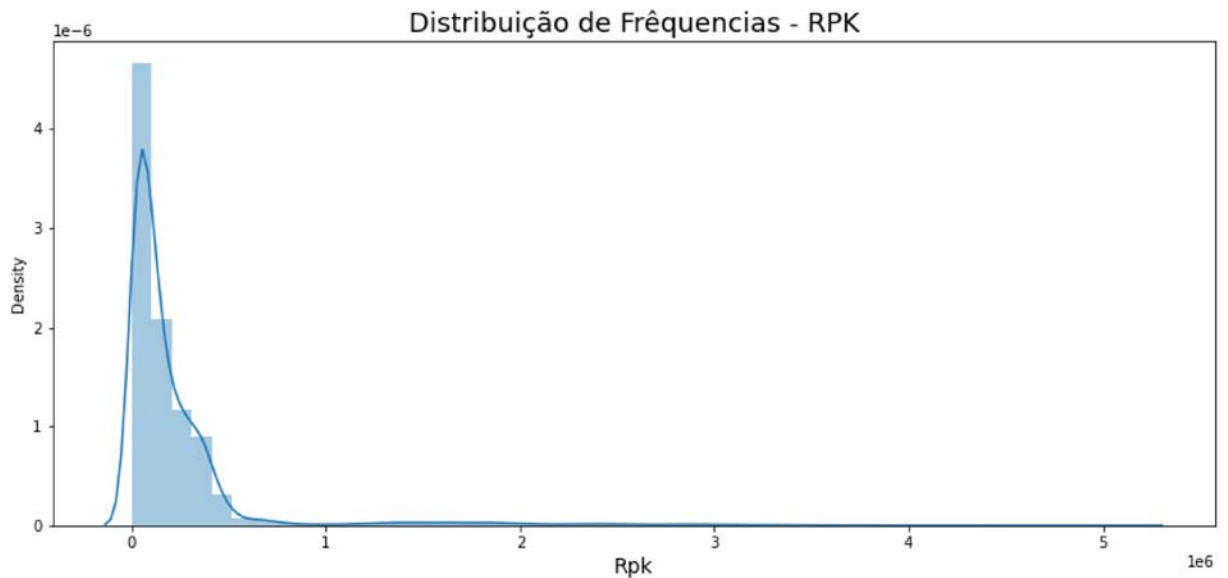


```
In [ ]: histograma = sns.distplot(_df['rpk'])
histograma.figure.set_size_inches(14,6)
histograma.set_title('Distribuição de Frequências - RPK', fontsize=18)
histograma.set_xlabel('Rpk', fontsize=14)
```

C:\Users\conta\AppData\Local\Programs\Python\Python39\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

```
warnings.warn(msg, FutureWarning)
```

```
Out[ ]: Text(0.5, 0, 'Rpk')
```

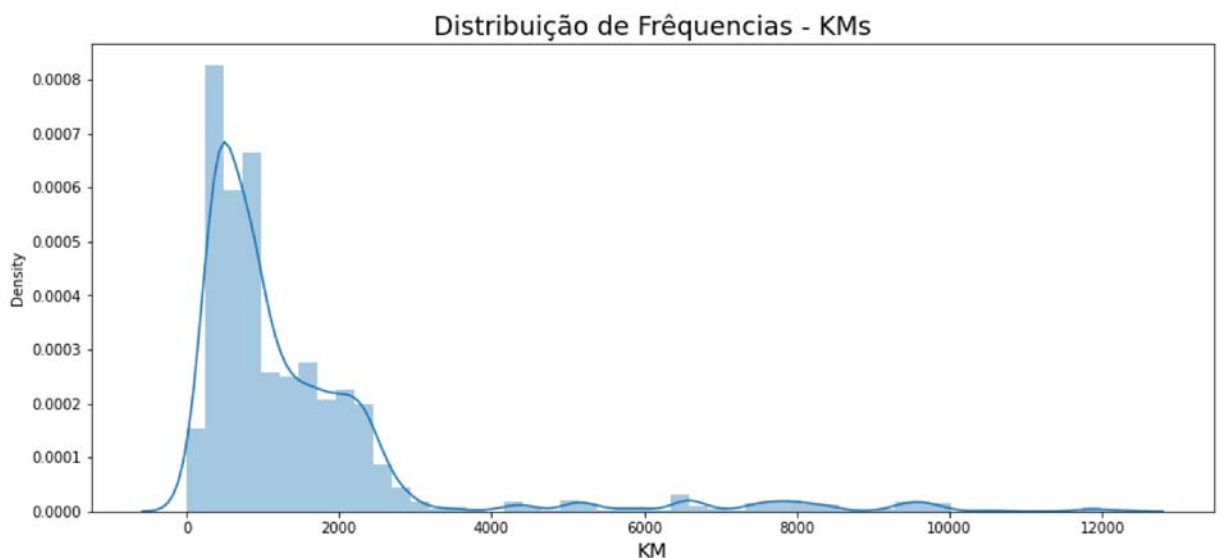


```
In [ ]: histograma = sns.distplot(_df['km_distancia'])
        histograma.figure.set_size_inches(14,6)
        histograma.set_title('Distribuição de Frêquências - KMs', fontsize=18)
        histograma.set_xlabel('KM', fontsize=14)
```

C:\Users\conta\AppData\Local\Programs\Python\Python39\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)

```
Out[ ]: Text(0.5, 0, 'KM')
```



```
In [ ]: _df['nr_horas_voadas'] = _df['nr_horas_voadas'].str.replace(',', '.').astype(float)
        _df['nr_velocidade_media'] = _df['nr_velocidade_media'].str.replace(',', '.').astype(float)
```

```
In [ ]: _df.head(3)
```

```
Out[ ]: id_basica id_empresa nm_empresa nm_pais ds_tipo_empresa nr_voo dt_referencia nm_sem
```

id_basica	id_empresa	nm_empresa	nm_pais	ds_tipo_empr	ds_uf	ds_estado	nm_ciu
1	1000000	AEROLINHAS BRASILEIRAS S.A.	BRASIL	ESTADUAL	AC	2000-01-01	
2	1000000	AEROLINEAS ARGENTINAS S/A	ARGENTINA	ESTADUAL	BA	2000-01-01	
3	1000000	AEROSUL TÁXI AÉREO LTDA (EX.: AUSTEN TÁXI AÉREO LTDA)	BRASIL	ESTADUAL	AC	2000-01-01	
4	1000000	AEROVIAS DEL CONTINENTE AMERICANO S.A. AVIANCA	COLOMBIA	ESTADUAL	BA	2000-01-01	
5	1000000	AEROVÍAS DE MÉXICO S.A DE C.V - AEROMÉXICO	MEXICO	ESTADUAL	BA	2000-01-01	
6	1000000	AIR CANADA	CANADA	ESTADUAL	BA	2000-01-01	
7	1000000	AIR EUROPA LINEAS AEREAS SOCIEDAD ANONIMA	ESPAÑA	ESTADUAL	BA	2000-01-01	
8	1000000	AMERICAN AIRLINES, INC.	ESTADOS UNIDOS	ESTADUAL	BA	2000-01-01	
9	1000000	ATLAS AIR INC	CANADA	ESTADUAL	BA	2000-01-01	
10	1000000	AZUL CONECTA LTDA. (EX TWO TAXI AEREO LTDA)	BRASIL	ESTADUAL	AC	2000-01-01	
11	1000000	AZUL LINHAS AÉREAS BRASILEIRAS S/A	BRASIL	ESTADUAL	AC	2000-01-01	
12	1000000	BOA BOLIVIANA DE AVIACIÓN	BOLIVIA	ESTADUAL	BA	2000-01-01	
13	1000000	BRITISH AIRWAYS PLC	REINO UNIDO	ESTADUAL	BA	2000-01-01	
14	1000000	CARGOJET AIRWAYS LTD.	DINAMARCA	ESTADUAL	BA	2000-01-01	
15	1000000	CARGOLUX AIRLINES INTERNATIONAL S/A	LUZEMBURGO	ESTADUAL	BA	2000-01-01	
16	1000000	COMPANHIA DE AVIACION PARAGUAYA SOCIEDAD ANONIMA - PARANAIR (EX- AMASZONAS PARA GUAY)	PARAGUAY	ESTADUAL	BA	2000-01-01	
17	1000000	COMPANHIA DE SERVICIOS DE TRANSPORTE AEREO AMASZONAS S/A.	BRASIL	ESTADUAL	AC	2000-01-01	
18	1000000	COMPANHIA PANAMEÑA DE AVIACION S.A. (COPA AIRLINES)	PANAMA	ESTADUAL	BA	2000-01-01	
19	1000000	DELTA AIR LINES INC.	ESTADOS UNIDOS	ESTADUAL	BA	2000-01-01	
20	1000000	DEUTSCHE LUFTHANSA A.G.	ALEMANIA	ESTADUAL	BA	2000-01-01	
21	1000000	EMIRATES	EMIRATOS ARABES UNIDOS	ESTADUAL	BA	2000-01-01	
22	1000000	ETHIOPIAN AIRLINES GROUP	ETIOPIA	ESTADUAL	BA	2000-01-01	
23	1000000	EUROATLANTIC AIRWAYS	IRLANDA	ESTADUAL	BA	2000-01-01	
24	1000000	FB LÍNEAS AÉREAS S.A (FLYBONDI)	ARGENTINA	ESTADUAL	BA	2000-01-01	
25	1000000	FEDEX - FEDERAL EXPRESS CORPORATION	ESTADOS UNIDOS	ESTADUAL	BA	2000-01-01	
26	1000000	FLY ALLWAYS N.V.	PAISES BAJOS	ESTADUAL	BA	2000-01-01	
27	1000000	FLY PRO S.R.L.	ARGENTINA	ESTADUAL	BA	2000-01-01	
28	1000000	GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS S.A.)	BRASIL	ESTADUAL	AC	2000-01-01	
29	1000000	IBÉRIA LINEAS AEREAS DE ESPAÑA SOCIEDAD ANONIMA OPERADORA	ESPAÑA	ESTADUAL	BA	2000-01-01	
30	1000000	ITALIA TRANSPORTO AEREO S.P.A.	ITALIA	ESTADUAL	BA	2000-01-01	
31	1000000	JETSMART AIRLINES SPA - CHILE	CHILE	ESTADUAL	BA	2000-01-01	

```
In [ ]:
_empresa = {}
empresa = _df['nm_empresa'].sort_values().unique()
for i in range(len(empresa)):
    _empresa[i] = empresa[i]
_empresa
```

```
Out[ ]: {0: 'ABSA - AEROLINHAS BRASILEIRAS S.A.',
1: 'AEROLINEAS ARGENTINAS S/A',
2: 'AEROSUL TÁXI AÉREO LTDA (EX.: AUSTEN TÁXI AÉREO LTDA)',
3: 'AEROVIAS DEL CONTINENTE AMERICANO S.A. AVIANCA',
4: 'AEROVÍAS DE MÉXICO S.A DE C.V - AEROMÉXICO',
5: 'AIR CANADA',
6: 'AIR EUROPA LINEAS AEREAS SOCIEDAD ANONIMA',
7: 'AMERICAN AIRLINES, INC.',
8: 'ATLAS AIR INC',
9: 'AZUL CONECTA LTDA. (EX TWO TAXI AEREO LTDA)',
10: 'AZUL LINHAS AÉREAS BRASILEIRAS S/A',
11: 'BOA BOLIVIANA DE AVIACIÓN',
12: 'BRITISH AIRWAYS PLC',
13: 'CARGOJET AIRWAYS LTD.',
14: 'CARGOLUX AIRLINES INTERNATIONAL S/A',
15: 'COMPANHIA DE AVIACION PARAGUAYA SOCIEDAD ANONIMA - PARANAIR (EX- AMASZONAS PARA GUAY)',
16: 'COMPANHIA DE SERVICIOS DE TRANSPORTE AEREO AMASZONAS S/A.',
17: 'COMPANHIA PANAMEÑA DE AVIACION S.A. (COPA AIRLINES)',
18: 'DELTA AIR LINES INC.',
19: 'DEUTSCHE LUFTHANSA A.G.',
20: 'EMIRATES',
21: 'ETHIOPIAN AIRLINES GROUP',
22: 'EUROATLANTIC AIRWAYS',
23: 'FB LÍNEAS AÉREAS S.A (FLYBONDI)',
24: 'FEDEX - FEDERAL EXPRESS CORPORATION',
25: 'FLY ALLWAYS N.V.',
26: 'FLY PRO S.R.L.',
27: 'GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS S.A.)',
28: 'IBÉRIA LINEAS AEREAS DE ESPAÑA SOCIEDAD ANONIMA OPERADORA',
29: 'ITALIA TRANSPORTO AEREO S.P.A.',
30: 'JETSMART AIRLINES SPA - CHILE',
```

```

31: 'KALITTA AIR LLC',
32: 'KLM CIA. REAL HOLANDESA DE AVIAÇÃO',
33: 'KOREAN AIR LINES CO. LTD.',
34: 'LAN CARGO S.A.',
35: 'LAN PERU S.A.',
36: 'LATAM AIRLINES GROUP (EX - LAN AIRLINES S/A)',
37: 'LUFTHANSA CARGO AG',
38: 'LÍNEA AÉREA CARGUEIRA DE COLOMBIA S.A - LANCO',
39: 'MAP TRANSPORTES AÉREOS LTDA.',
40: 'MARTINAIR HOLLAND N.V.',
41: 'MODERN TRANSPORTE AEREO DE CARGA S.A',
42: 'NEOS S.P.A',
43: 'OMNI AIR INTERNATIONAL',
44: 'OMNI TÁXI AÉREO S.A.',
45: 'PASSAREDO TRANSPORTES AÉREOS S.A.',
46: 'PRIVILEGE STYLE S.A.',
47: 'QATAR AIRWAYS GROUP',
48: 'SATA INTERNACIONAL - AZORES AIRLINES',
49: 'SIDERAL LINHAS AÉREAS LTDA.',
50: 'SKY AIRLINES S.A.',
51: 'SOCIÉTÉ AIR FRANCE',
52: 'SURINAM AIRWAYS LTD',
53: 'SWIFT AIR, LLC',
54: 'SWISS INTERNATIONAL AIR LINES LTD.',
55: 'TAAG LINHAS AÉREAS DE ANGOLA (ANGOLA AIRLINES)',
56: 'TAM LINHAS AÉREAS S.A.',
57: 'TAMPA CARGO S.A.',
58: 'TAP - TRANSPORTES AÉREOS PORTUGUESES S/A',
59: 'TOTAL LINHAS AÉREAS S.A.',
60: 'TRANS GUYANA AIRWAYS',
61: 'TRANSPORTE AÉREOS DEL MERCOSUR S.A. (TAM MERCOSUR)',
62: 'TURKISH AIRLINES INC',
63: 'UKRAINE AIR ALLIANCE AIRLINES',
64: 'UNITED AIRLINES, INC',
65: 'UPS - UNITED PARCEL SERVICE CO.'}

```

In []:

```

_pais = {}
pais = _df['nm_pais'].sort_values().unique()
for i in range(len(pais)):
    _pais[i] = pais[i]
_pais

```

Out[]:

```

{0: 'ALEMANHA',
1: 'ANGOLA',
2: 'ARGENTINA',
3: 'BOLÍVIA',
4: 'BRASIL',
5: 'CANADÁ',
6: 'CHILE',
7: 'COLÔMBIA',
8: 'COREIA DO SUL',
9: 'EMIRADOS ÁRABES UNIDOS',
10: 'ESPANHA',
11: 'ESTADOS UNIDOS DA AMÉRICA',
12: 'ETIÓPIA',
13: 'FRANÇA',
14: 'GUIANA',
15: 'HOLANDA',
16: 'INGLATERRA',
17: 'ITÁLIA',
18: 'LUXEMBURGO',
19: 'MOLDÁVIA',
20: 'MÉXICO',

```


Flights by companies.

```

frequencia = pd.crosstab(_df['nm_pais'], _df['nm_empresa'])
frequencia.rename(columns=_pais, inplace=True)
frequencia.rename(index=_empresa, inplace=True)
frequencia

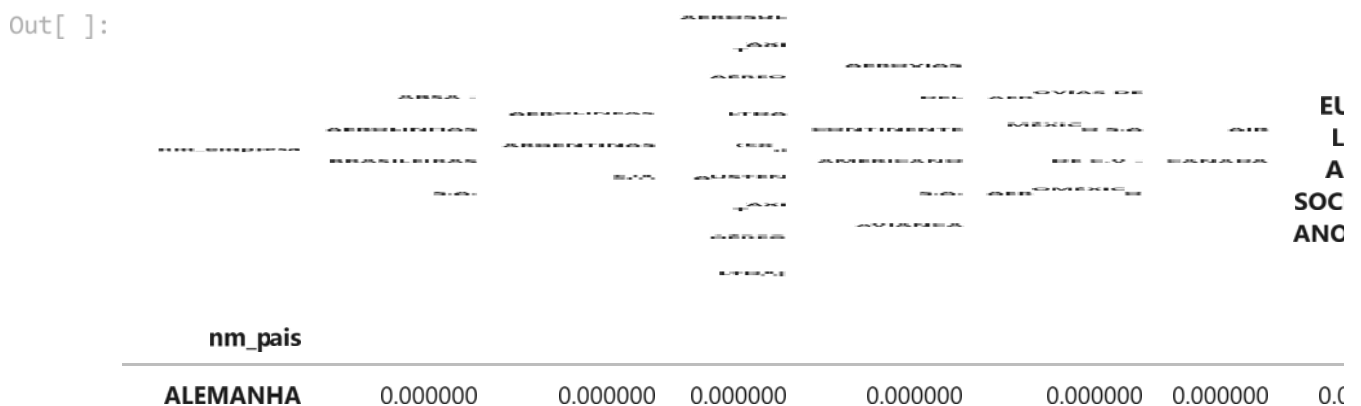
```

COUNTRY	REGIONAL						EUROPEAN LAW AND SOCIAL ANALYSIS	
	AMERICAN		AFRICAN		ASIAN			
	EUROPEAN		AFRICAN		ASIAN			
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			
COUNTRY	AMERICAN		AFRICAN		ASIAN		EUROPEAN LAW AND SOCIAL ANALYSIS	
	EUROPEAN		AFRICAN		ASIAN			



Percent of Flights by companies.

```
In [ ]: frecuencia = pd.crosstab(_df['nm_pais'], _df['nm_empresa'], normalize=True)
frecuencia.rename(columns=_pais,inplace=True)
frecuencia.rename(index=_empresa,inplace=True)
frecuencia
```



[illegible]

[illegible]

```
In [ ]: frequencia.loc['BRASIL'].sort_values(ascending=False)
```

```
Out[ ]: nm_empresa
AZUL LINHAS AÉREAS BRASILEIRAS S/A      0.303821
GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS S.A.) 0.286923
TAM LINHAS AÉREAS S.A.                   0.261980
AZUL CONECTA LTDA. (EX TWO TAXI AEREO LTDA) 0.019211
PASSAREDO TRANSPORTES AÉREOS S.A.       0.016926
...
ITALIA TRANSPORTO AEREO S.P.A.           0.000000
JETSMART AIRLINES SPA - CHILE            0.000000
KALITTA AIR LLC                          0.000000
KLM CIA. REAL HOLANDESA DE AVIAÇÃO      0.000000
UPS - UNITED PARCEL SERVICE CO.          0.000000
Name: BRASIL, Length: 66, dtype: float64
```

We can concluded that Azul Linhas Aéreas company represent 30% of the flights of Januray 2022.

Grouping of passengers by company

```
In [ ]: passenger_by_comp = _df.groupby(['nm_dia_semana_referencia', 'nm_empresa'])['pax'].s
passenger by comp
```

```
Out[ ]: nm_dia_semana_referencia nm_empresa
DOMINGO ABSA - AEROLINHAS BRASILEIRAS S.A. 0
AEROLINEAS ARGENTINAS S/A 8852
AEROVIAS DEL CONTINENTE AMERICANO S.A. AVIANCA 2992
AEROVÍAS DE MÉXICO S.A DE C.V - AEROMÉXICO 3355
AIR CANADA 4561
...
TERÇA-FEIRA TRANS GUYANA AIRWAYS 15
TRANSPORTE AÉREOS DEL MERCOSUR S.A. (TAM MERCOSUR) 855
TURKISH AIRLINES INC 2764
UNITED AIRLINES, INC 6981
UPS - UNITED PARCEL SERVICE CO. 0
Name: pax, Length: 377, dtype: int64
```

```
In [ ]: passenger_by_comp = passenger_by_comp.reset_index().rename(columns={'nm_dia_semana_r': 'Dia da Semana'})
passenger_by_comp = passenger_by_comp.sort_values('Quantidade de Passageiros', ascending=False)
# passenger by comp['Datas'] = pd.to_datetime(passenger_by_comp['Datas'])
```

```
passenger_by_comp['Quantidade de Passageiros'] = passenger_by_comp['Quantidade de Pa
passenger_by_comp
```

```
Out[ ]: 

|   | Dias da Semana | Companhias Aéreas                                 | Quantidade de Passageiros |
|---|----------------|---------------------------------------------------|---------------------------|
| 0 | SEGUNDA-FEIRA  | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 473084.0                  |
| 1 | DOMINGO        | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 462938.0                  |
| 2 | SEGUNDA-FEIRA  | TAM LINHAS AÉREAS S.A.                            | 431171.0                  |
| 3 | DOMINGO        | TAM LINHAS AÉREAS S.A.                            | 430066.0                  |
| 4 | SÁBADO         | TAM LINHAS AÉREAS S.A.                            | 416205.0                  |
| 5 | SÁBADO         | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 415139.0                  |
| 6 | SEGUNDA-FEIRA  | AZUL LINHAS AÉREAS BRASILEIRAS S/A                | 381685.0                  |
| 7 | TERÇA-FEIRA    | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 365252.0                  |
| 8 | SEXTA-FEIRA    | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 358215.0                  |
| 9 | QUARTA-FEIRA   | GOL LINHAS AÉREAS S.A. (EX- VRG LINHAS AÉREAS ... | 354522.0                  |


```

Top Ten Volumes

```
In [ ]: passenger_by_comp.dtypes
```

```
Out[ ]: Dias da Semana      object
Companhias Aéreas      object
Quantidade de Passageiros  float64
dtype: object
```

```
In [ ]: titulo = 'Passageiros por Companhia Aérea'
plotar(titulo=titulo, labelx='Datas', labely='Quantidade de Passageiros', x='Dias da
```

```
Out[ ]: <AxesSubplot:title={'left':'Passageiros por Companhia Aérea'}, xlabel='Datas', ylabel='Quantidade de Passageiros'>
```

Passageiros por Companhia Aérea

