

```
In [74]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import datetime as dt

In [75]: pd.set_option('display.max_columns',100)

In [76]: df=pd.read_csv(r'C:\Users\berid\OneDrive\Desktop\mydata\Rusian export\RUSToWorldTrade.csv')

C:\Users\berid\AppData\Local\Programs\Python\Python39\lib\site-packages\IPython\core\interactiveshell.py:3251: DtypeWarning:

Columns (11) have mixed types.Specify dtype option on import or set low_memory=False.

In [77]: df.columns=df.columns.str.strip().str.title().str.replace(' ','')

In [78]: df.head()
```

Out[78]:

	Index	Classification	Year	AggregateLevel	IsLeafCode	ReporterCode	Reporter	ReporterIso	PartnerCode	Partner	PartnerIso	CommodityCode	Commodity	QtyUnitCode	QtyUnit	Qty	Netweight(Kg)	TradeValue(Us\$)	
0	0	S4	2011		1	0	643	Russian Federation	RUS	136	Cayman Isds	CYM	6	Manufactured goods classified chiefly by material	1	No Quantity	NaN	NaN	7559
1	1	S4	2012		1	0	643	Russian Federation	RUS	136	Cayman Isds	CYM	6	Manufactured goods classified chiefly by material	1	No Quantity	NaN	NaN	2777
2	2	S4	2015		1	0	643	Russian Federation	RUS	136	Cayman Isds	CYM	6	Manufactured goods classified chiefly by material	1	No Quantity	NaN	NaN	213157
3	3	S4	2016		1	0	643	Russian Federation	RUS	136	Cayman Isds	CYM	6	Manufactured goods classified chiefly by material	1	No Quantity	0.0	0.0	905554
4	4	S4	2017		1	0	643	Russian Federation	RUS	136	Cayman Isds	CYM	6	Manufactured goods classified chiefly by material	1	No Quantity	0.0	0.0	1995259

```
In [79]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1384603 entries, 0 to 1384602
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Index                 1384603 non-null int64
1   Classification        1384603 non-null object
2   Year                 1384603 non-null int64
3   AggregateLevel       1384603 non-null int64
4   IsLeafCode           1384603 non-null int64
5   ReporterCode         1384603 non-null int64
6   Reporter             1384603 non-null object
7   ReporterIso          1384603 non-null object
8   PartnerCode          1384603 non-null int64
9   Partner              1384603 non-null object
10  PartnerIso           1375914 non-null object
11  CommodityCode        1384603 non-null object
12  Commodity            1384603 non-null object
13  QtyUnitCode          1384603 non-null int64
14  QtyUnit              1384603 non-null object
15  Qty                  1283894 non-null float64
16  Netweight(Kg)        1323567 non-null float64
17  TradeValue(Us$)     1384603 non-null int64
dtypes: float64(2), int64(8), object(8)
memory usage: 190.1+ MB
```

```
In [80]: df.shape
```

Out[80]: (1384603, 18)

Plot export value by country for each year

```
In [81]: grouped=df.groupby(['Year', 'Partner'])['TradeValue(Us$)'].sum().reset_index().sort_values(['Year', 'TradeValue(Us$)'],ascending=[True,False])\
        .groupby('Year').head(30)
grouped['Billion']=(round(grouped['TradeValue(Us$)']/10**9,1)).astype(str)

fig=px.bar(grouped,x='Partner',y='TradeValue(Us$)',
           animation_frame='Year',
           range_y=(0,grouped['TradeValue(Us$)'].max()),
           text='Billion'
           )

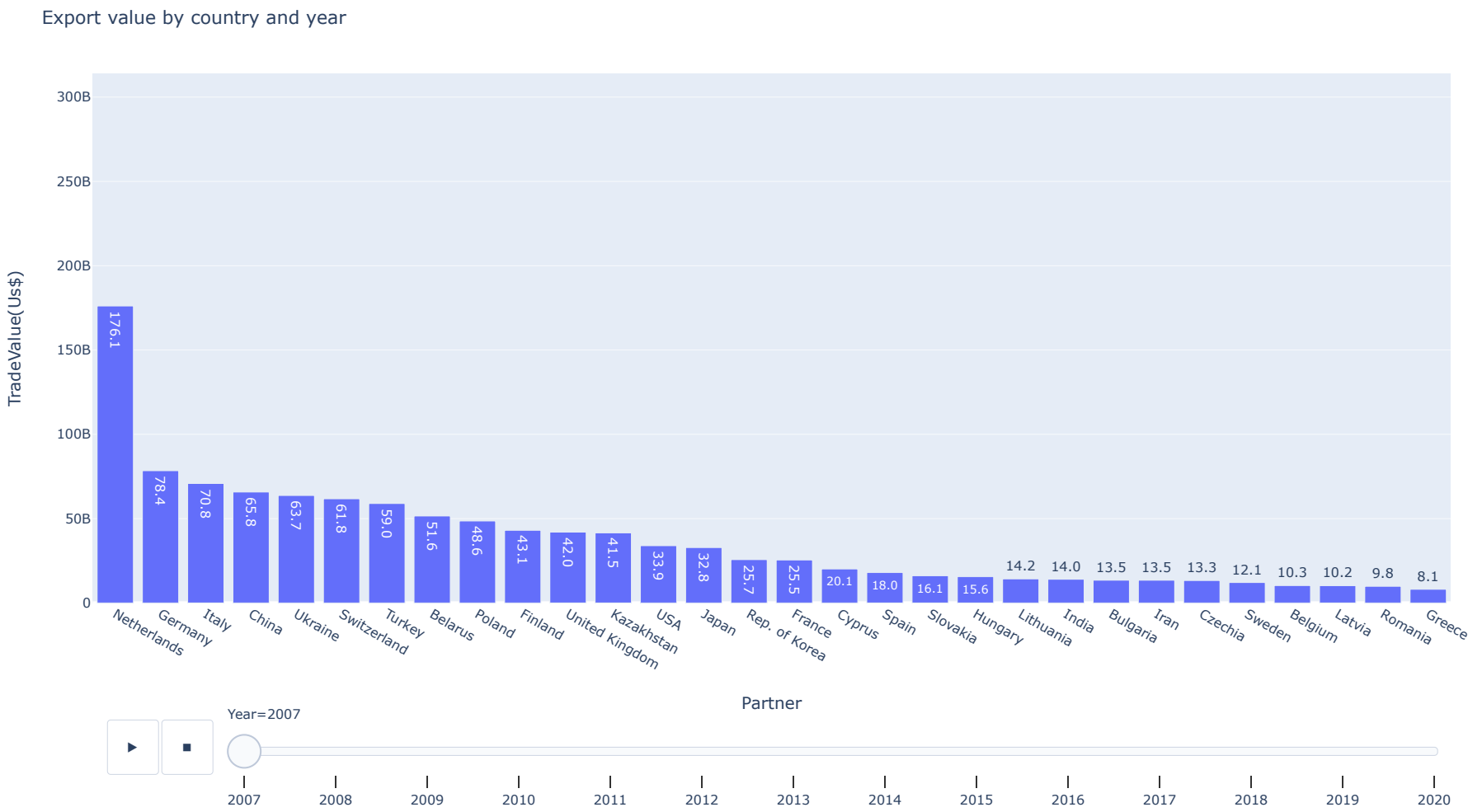
fig.update_layout(title_text='Export value by country and year',font=dict(size=8))
fig.layout.updatemenus[0].buttons[0].args[1]["frame"]["duration"] = 2000
fig.show()

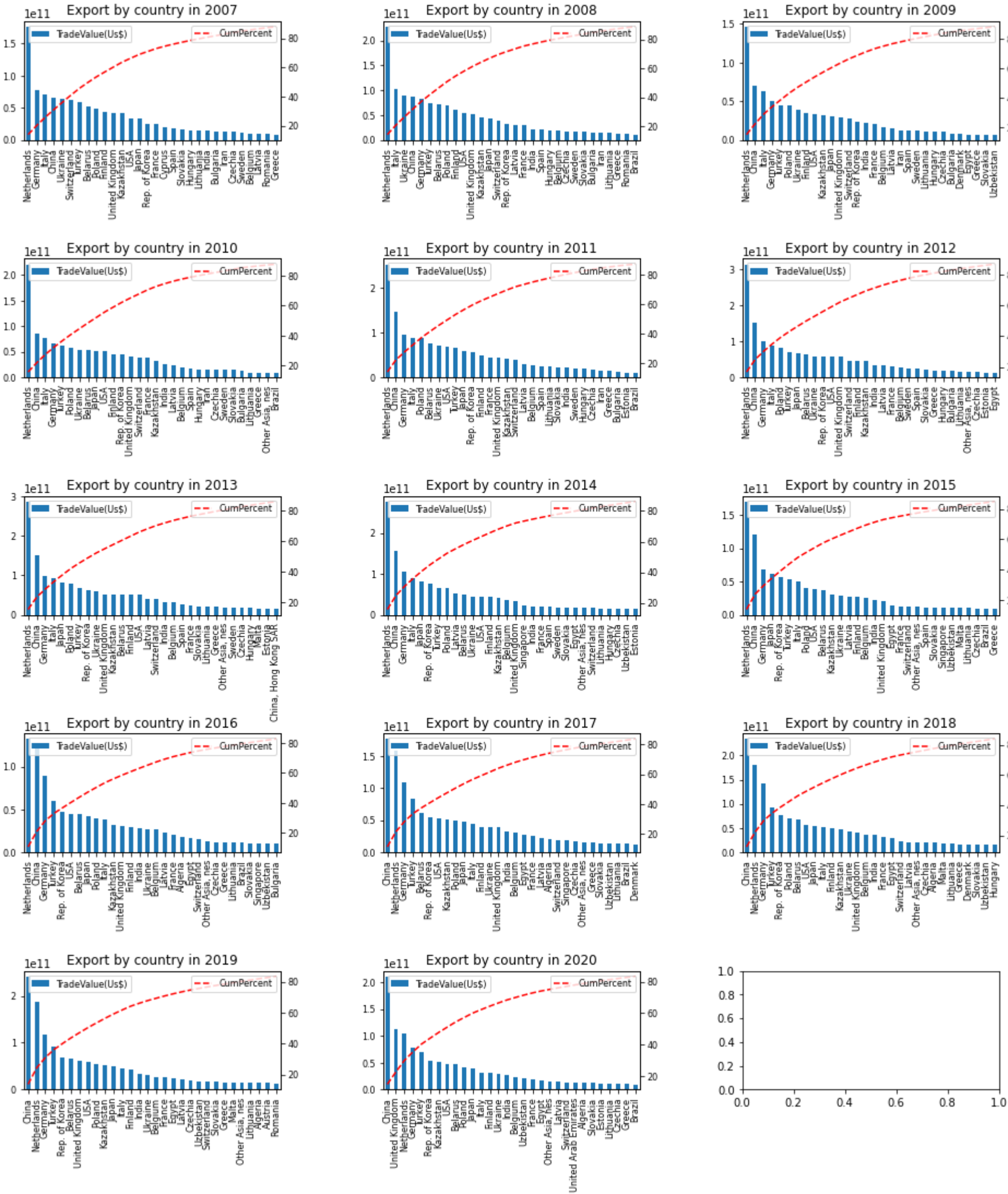
years=sorted(df.Year.unique())
fig,axes=plt.subplots(5,3,figsize=(16,18))
for ax,year in zip(axes.ravel(),years):
    gr=df[df.Year==year].groupby(['Year', 'Partner'])['TradeValue(Us$)'].sum().reset_index().sort_values(['Year', 'TradeValue(Us$)'],ascending=[True,False])
    gr['CumPercent']=(gr['TradeValue(Us$)']/gr['TradeValue(Us$)'].sum()*100).cumsum()
    gr=gr.groupby('Year').head(30)

    gr.plot(ax=ax,kind='bar',x='Partner',y='TradeValue(Us$)',fontsize=8,xlabel='')
    ax.set_title('Export by country in '+str(year))
    ax.legend(fontsize=8,loc='upper left')

    ax2=ax.twinx()
    gr.plot(ax=ax2,kind='line',x='Partner',y='CumPercent',fontsize=8,xlabel='',ls='--',color='r')
    ax2.set_label('Cumulative Percent')
    ax2.legend(fontsize=8,loc='upper right')

plt.subplots_adjust(hspace=1,wspace=0.4)
plt.show()
```





```
In [82]: # In 2009, 2015 and 2016, 2020 russian export subsided.
# This must have been caused by sanctions after wars against Georgia and Ukraine in 2008 and 2014. In 2020 due to pandemic.
# Export was subsided for only one year. After one year Russian export reached the point that was before the wars.

# From 2007 till 2017 Netherlands had the Largest share among Russian export countries. Since 2017 China has the Largest share.
```

Share of commodity in Russian export by year

```
In [83]: from plotly.subplots import make_subplots
import plotly.graph_objects as go
import math

fig = make_subplots(rows=7, cols=2,
                    specs=[[{"type": "domain"} for _ in range(2)] for _ in range(7)],
                    subplot_titles=[str(year) for year in years])

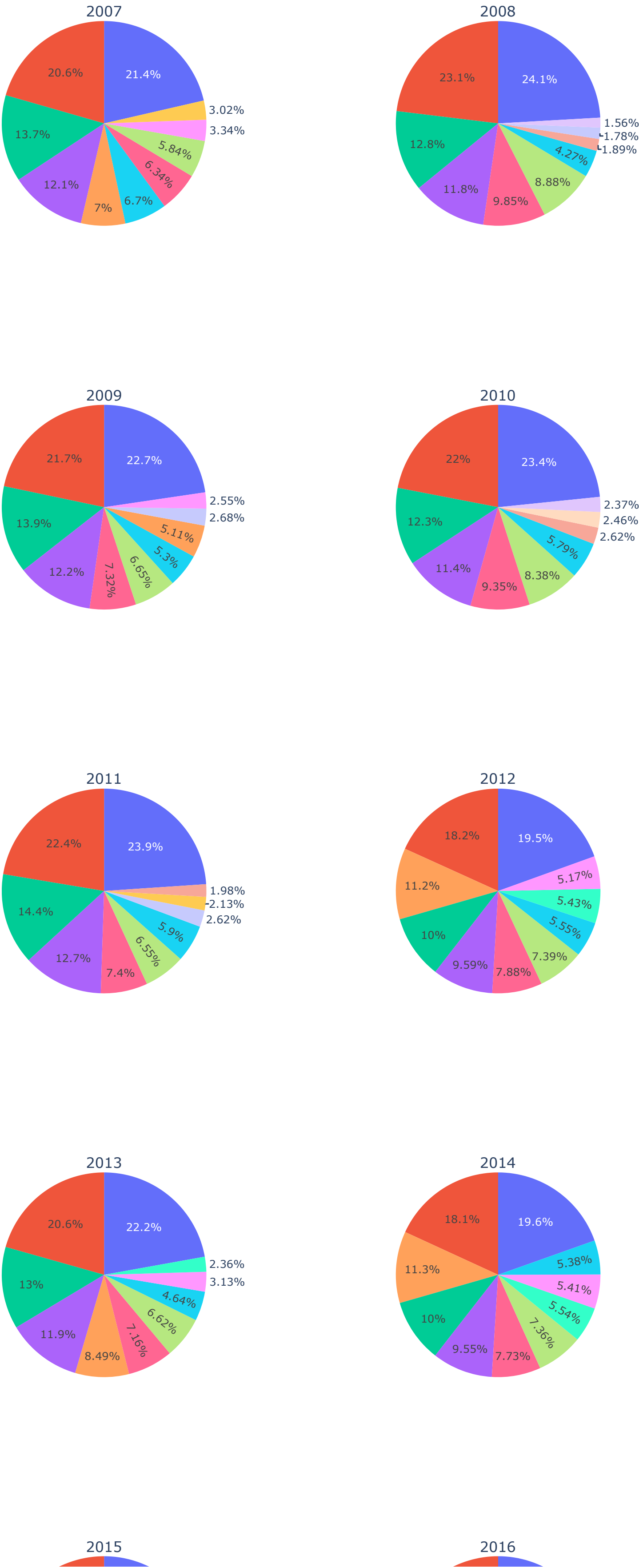
for i,year in enumerate(sorted(df.Year.unique())):

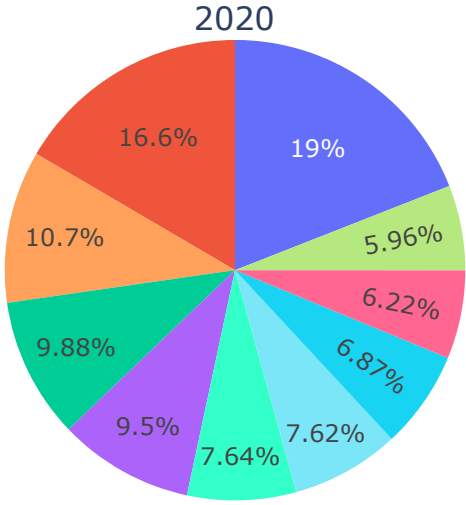
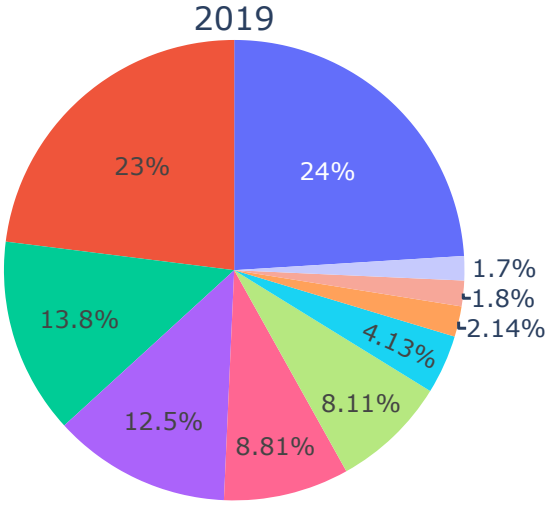
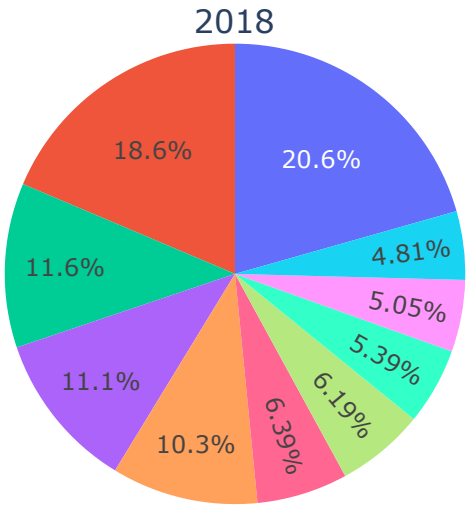
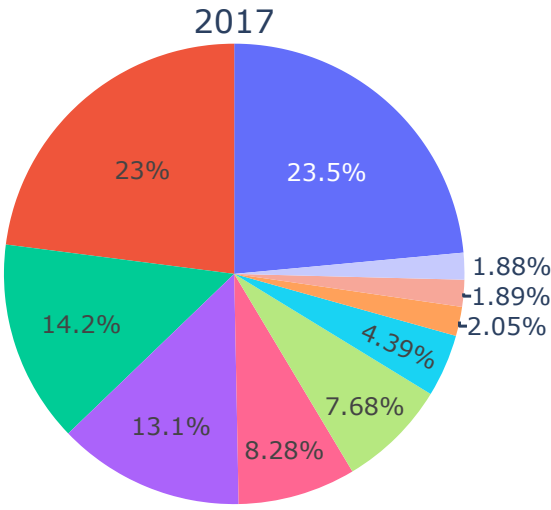
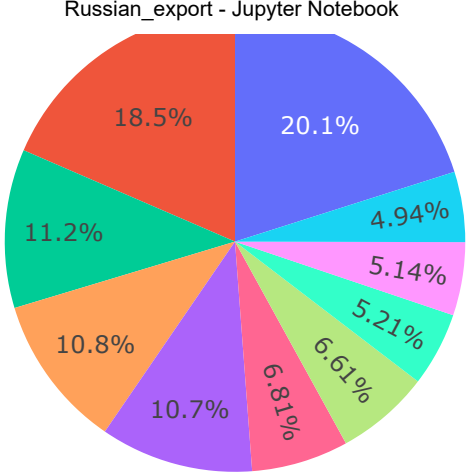
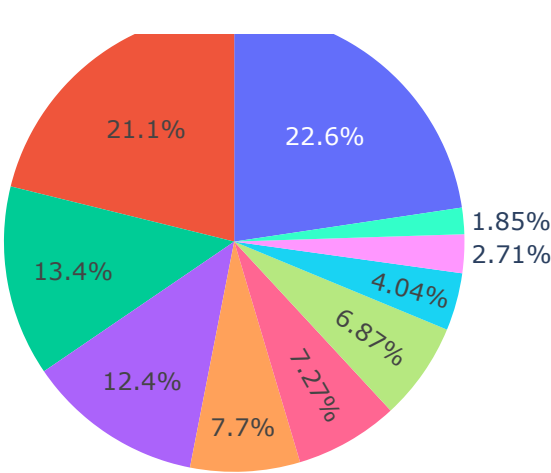
    grouped=df[df.Year==year].groupby('Commodity')['TradeValue(Us$)'].sum().reset_index().sort_values('TradeValue(Us$)',ascending=False).head(10)

    fig.add_trace(go.Pie(values=grouped['TradeValue(Us$)'],labels=grouped['Commodity'],name=str(year)),
                  row=i%7+1,col=math.floor(i/7+1))

    fig.update_layout(height=3000, width=1000, title_text="Commodity share for each year",showlegend=False,paper_bgcolor="white")
fig.show()
```

Commodity share for each year





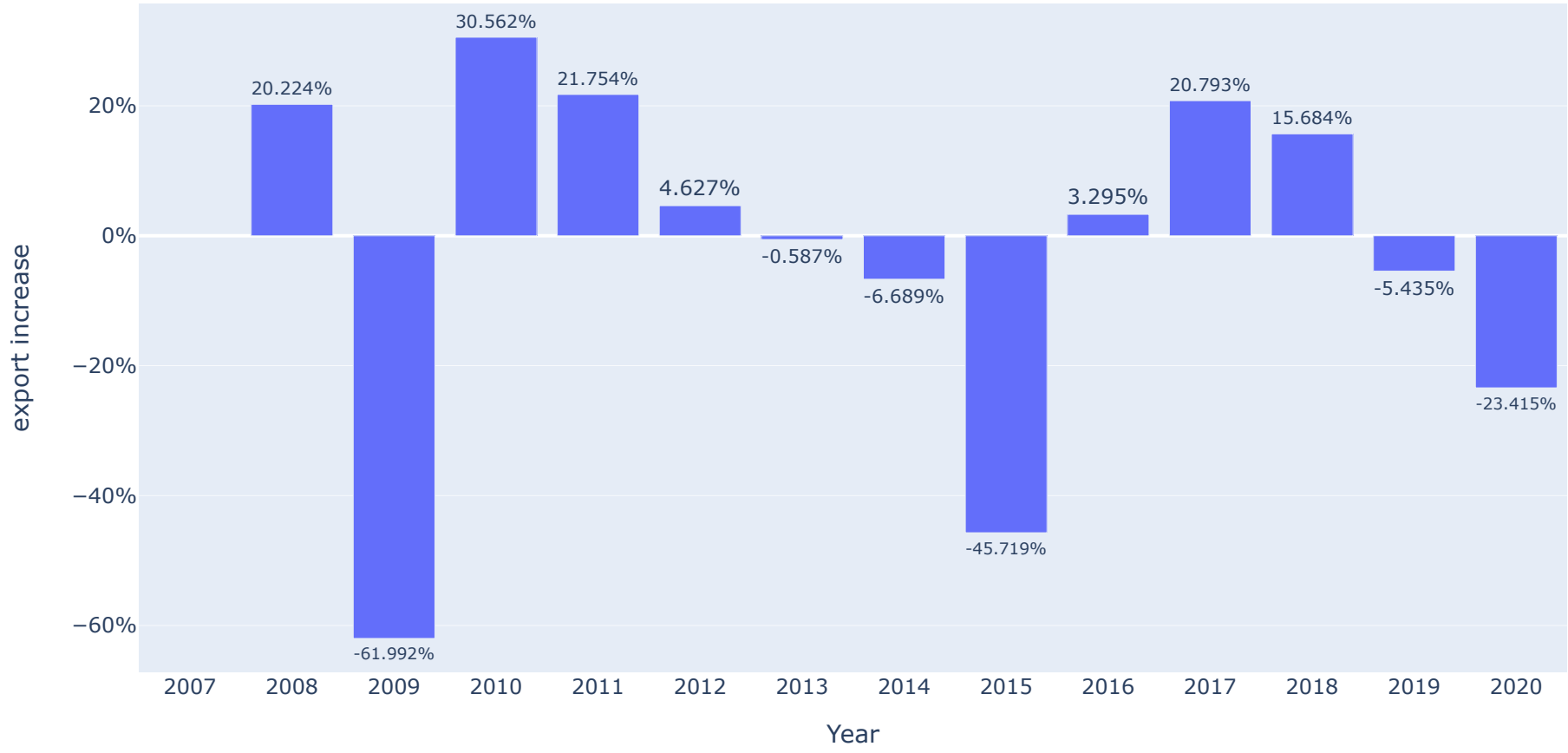
How Russian export was increasing year by year

```
In [84]: grouped=df.groupby(['Year'])['TradeValue(Us$)'].sum().reset_index().sort_values('Year',ascending=True)
grouped['Previous']=grouped['TradeValue(Us$)'].shift(1)
grouped['Increase']=(grouped['TradeValue(Us$)']-grouped['Previous'])/grouped['TradeValue(Us$)']

fig=px.bar(x=grouped.Year.astype(str),y=grouped['Increase'],text_auto='.3%')
fig.update_layout(title_text='Value of Russian export by year')
fig.layout.yaxis.tickformat = '.0%'
fig.update_xaxes(title_text='Year')
fig.update_yaxes(title_text='export increase')
fig.update_traces(textposition='outside')

fig.show()
```

Value of Russian export by year



```
In [ ]:
```