

Importing libraries

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [44]: data= pd.read_csv("merchandise_values_annual_dataset.csv", encoding= 'latin-1',
```

View dataset

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

Out[3]:

	IndicatorCategory	IndicatorCode	Indicator	ReporterCode	ReporterISO3A	Rep
0	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	533	ABW	Nether with re
1	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	320	GTM	Guat
2	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	528	NLD	Nether
3	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	296	KIR	K
4	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	624	GNB	Gu I

5 rows × 23 columns



About the Dataset

This dataset presents a comprehensive record of annual merchandise trade values for countries and regions worldwide, spanning from 1948 to 2024. It covers both imports and exports, providing detailed breakdowns by product classification (based on the SITC3

– Standard International Trade Classification, Revision 3), trade partners, and estimated versus official reporting.

The data is reported in Million US Dollars and standardized to an annual frequency, making it well-suited for time series analysis, international trade research, and economic trend modeling. Users can explore trade patterns over more than seven decades, identify shifts in product demand, compare regional trade balances, or model the impact of global events on merchandise trade flows.

Column Description:

- **IndicatorCategory** : Broad category of trade metric — always "Merchandise trade values".
- **IndicatorCode** : Internal code representing the indicator type (e.g., ITS_MTV_AM).
- **Indicator** : Full description of the indicator — often "Merchandise imports by product group – annual".
- **ReporterCode** : Numeric/textual code for the reporting country or region.
- **ReporterISO3A** : ISO-3 alpha country code of the reporting entity (e.g., IND, USA), missing for some rows.
- **Reporter** : Name of the country/region reporting the data.
- **PartnerCode** : Numeric/textual code of the trade partner (e.g., 000 for World).
- **PartnerISO3A** : ISO-3 alpha code of the partner country/region — largely missing due to aggregated "World" category.
- **Partner** : Name of the trade partner (commonly "World" in aggregated data).
- **ProductClassificationCode** : Classification code of the product categorization system — always SITC3 here.
- **ProductClassification** : Full name of the classification system used — "Standard International Trade Classification, Revision 3".
- **ProductCode** : Code of the specific product or commodity category.
- **Product** : Full name/description of the product category.
- **PeriodCode** **Period** : code — always A for Annual.
- **Period** **Frequency** : description — "Annual".
- **FrequencyCode** : Code for frequency — always A (Annual).
- **Frequency** : Full frequency description — always "Annual".
- **UnitCode** : Code for unit of measurement — USM (Million US dollars).
- **Unit** : Unit description — "Million US dollar".
- **Year** : Reporting year (1948–2024).
- **ValueFlagCode** : Code indicating special conditions or estimates.
- **ValueFlag** : Description of value flag (e.g., "Estimate"), mostly missing.
- **Value** : Trade value in million USD.

Issues with the dataset

- Mixed data type in *ReporterCode* column . **consistency issue**
- Missing values in *ReporterISO3A* column. **completeness issue**

- Over 2.3 Lakhs rows in *PartnerCode* column have values as "000". completeness
- More than 95% Missing Values in *PartnerISO3A*, *ValueFlagCode* and *ValueFlag* columns. completeness

Creating a copy of dataframe

```
In [4]: df= data.copy()
```

```
In [5]: # view dataset  
df.head()
```

```
Out[5]:
```

	IndicatorCategory	IndicatorCode	Indicator	ReporterCode	ReporterISO3A	Rep
--	-------------------	---------------	-----------	--------------	---------------	-----

0	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	533	ABW	Nether with re
---	--------------------------	------------	---	-----	-----	----------------

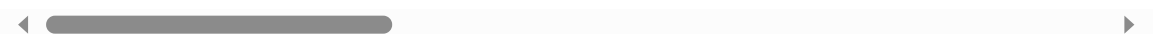
1	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	320	GTM	Guat
---	--------------------------	------------	---	-----	-----	------

2	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	528	NLD	Nether
---	--------------------------	------------	---	-----	-----	--------

3	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	296	KIR	K
---	--------------------------	------------	---	-----	-----	---

4	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	624	GNB	Gu I
---	--------------------------	------------	---	-----	-----	---------

5 rows × 23 columns



Summary of the dataframe

```
In [6]: df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 231924 entries, 0 to 231923
Data columns (total 23 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   IndicatorCategory                     231924 non-null object
1   IndicatorCode                         231924 non-null object
2   Indicator                             231924 non-null object
3   ReporterCode                         231924 non-null object
4   ReporterISO3A                        222189 non-null object
5   Reporter                             231924 non-null object
6   PartnerCode                          231924 non-null object
7   PartnerISO3A                         866 non-null    object
8   Partner                             231924 non-null object
9   ProductClassificationCode            231924 non-null object
10  ProductClassification                 231924 non-null object
11  ProductCode                          231924 non-null object
12  Product                              231924 non-null object
13  PeriodCode                           231924 non-null object
14  Period                               231924 non-null object
15  FrequencyCode                        231924 non-null object
16  Frequency                            231924 non-null object
17  UnitCode                             231924 non-null object
18  Unit                                 231924 non-null object
19  Year                                 231924 non-null int64
20  ValueFlagCode                        11798 non-null  object
21  ValueFlag                            11798 non-null  object
22  Value                                231924 non-null int64
dtypes: int64(2), object(21)
memory usage: 40.7+ MB

```

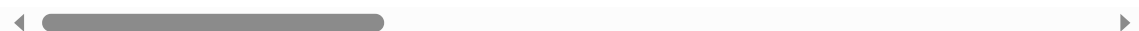
Sample of dataset

In [7]: `df.sample(3)`

Out[7]:

	IndicatorCategory	IndicatorCode	Indicator	ReporterCode	ReporterISO3A
162152	Merchandise trade values	ITS_MTV_AX	Merchandise exports by product group - annual	780	TTO
160688	Merchandise trade values	ITS_MTV_AX	Merchandise exports by product group - annual	690	SYC
68819	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	400	JOR

3 rows × 23 columns



List of column names

```
In [8]: df.columns
```

```
Out[8]: Index(['IndicatorCategory', 'IndicatorCode', 'Indicator', 'ReporterCode',  
              'ReporterISO3A', 'Reporter', 'PartnerCode', 'PartnerISO3A', 'Partner',  
              'ProductClassificationCode', 'ProductClassification', 'ProductCode',  
              'Product', 'PeriodCode', 'Period', 'FrequencyCode', 'Frequency',  
              'UnitCode', 'Unit', 'Year', 'ValueFlagCode', 'ValueFlag', 'Value'],  
             dtype='object')
```

Check for duplicate values

```
In [9]: df[df.duplicated()].sum()
```

```
Out[9]: IndicatorCategory      0  
IndicatorCode                0  
Indicator                   0  
ReporterCode                0  
ReporterISO3A               0  
Reporter                   0  
PartnerCode                 0  
PartnerISO3A                0  
Partner                    0  
ProductClassificationCode    0  
ProductClassification        0  
ProductCode                 0  
Product                    0  
PeriodCode                  0  
Period                      0  
FrequencyCode                0  
Frequency                   0  
UnitCode                    0  
Unit                       0  
Year                        0  
ValueFlagCode                0  
ValueFlag                   0  
Value                       0  
dtype: object
```

Data Cleaning

ReporterCode column

```
In [10]: # code  
df['ReporterCode'] = df['ReporterCode'].astype(str).str.strip().str.lstrip("'")
```

- Changing data type to string
- removing leading and trailing spaces
- removing apostrophe symbol

```
In [11]: # imputation
df['ReporterISO3A'] = df['ReporterISO3A'].fillna('World')
```

```
In [12]: # test
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 231924 entries, 0 to 231923
Data columns (total 23 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   IndicatorCategory                     231924 non-null object
1   IndicatorCode                         231924 non-null object
2   Indicator                             231924 non-null object
3   ReporterCode                         231924 non-null object
4   ReporterISO3A                       231924 non-null object
5   Reporter                             231924 non-null object
6   PartnerCode                         231924 non-null object
7   PartnerISO3A                        866 non-null    object
8   Partner                             231924 non-null object
9   ProductClassificationCode           231924 non-null object
10  ProductClassification                231924 non-null object
11  ProductCode                         231924 non-null object
12  Product                             231924 non-null object
13  PeriodCode                         231924 non-null object
14  Period                             231924 non-null object
15  FrequencyCode                      231924 non-null object
16  Frequency                          231924 non-null object
17  UnitCode                          231924 non-null object
18  Unit                              231924 non-null object
19  Year                              231924 non-null int64
20  ValueFlagCode                     11798 non-null object
21  ValueFlag                         11798 non-null object
22  Value                             231924 non-null int64
dtypes: int64(2), object(21)
memory usage: 40.7+ MB
```

PartnerCode column

```
In [13]: # code

# string operations on PartnerCode column
df['PartnerCode'] = df['PartnerCode'].astype(str).str.strip().str.lstrip("")

# Replace null values with World
df['PartnerISO3A'] = df['PartnerISO3A'].fillna('World')
```

```
In [14]: # test
df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 231924 entries, 0 to 231923
Data columns (total 23 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   IndicatorCategory                     231924 non-null object
1   IndicatorCode                         231924 non-null object
2   Indicator                             231924 non-null object
3   ReporterCode                         231924 non-null object
4   ReporterIS03A                        231924 non-null object
5   Reporter                             231924 non-null object
6   PartnerCode                          231924 non-null object
7   PartnerIS03A                         231924 non-null object
8   Partner                              231924 non-null object
9   ProductClassificationCode            231924 non-null object
10  ProductClassification                 231924 non-null object
11  ProductCode                          231924 non-null object
12  Product                              231924 non-null object
13  PeriodCode                           231924 non-null object
14  Period                               231924 non-null object
15  FrequencyCode                        231924 non-null object
16  Frequency                            231924 non-null object
17  UnitCode                             231924 non-null object
18  Unit                                 231924 non-null object
19  Year                                 231924 non-null int64
20  ValueFlagCode                        11798 non-null  object
21  ValueFlag                            11798 non-null  object
22  Value                                231924 non-null int64
dtypes: int64(2), object(21)
memory usage: 40.7+ MB

```

ValueFlagCode and ValueFlag columns

```

In [15]: # checking data distribution
df['ValueFlagCode'].value_counts()

```

```

Out[15]: ValueFlagCode
E      11126
B        672
Name: count, dtype: int64

```

```

In [16]: # code
df['ValueFlagCode'] = df['ValueFlagCode'].fillna('R')
df['ValueFlag'] = df['ValueFlag'].fillna('Reported')

```

```

In [17]: # Checking null values
df['ValueFlagCode'].isnull().sum()

```

```

Out[17]: 0

```

```

In [18]: # test
df.info()

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 231924 entries, 0 to 231923
Data columns (total 23 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   IndicatorCategory                     231924 non-null object
1   IndicatorCode                         231924 non-null object
2   Indicator                             231924 non-null object
3   ReporterCode                         231924 non-null object
4   ReporterISO3A                        231924 non-null object
5   Reporter                             231924 non-null object
6   PartnerCode                          231924 non-null object
7   PartnerISO3A                         231924 non-null object
8   Partner                              231924 non-null object
9   ProductClassificationCode            231924 non-null object
10  ProductClassification                 231924 non-null object
11  ProductCode                          231924 non-null object
12  Product                              231924 non-null object
13  PeriodCode                           231924 non-null object
14  Period                               231924 non-null object
15  FrequencyCode                        231924 non-null object
16  Frequency                            231924 non-null object
17  UnitCode                             231924 non-null object
18  Unit                                 231924 non-null object
19  Year                                 231924 non-null int64
20  ValueFlagCode                        231924 non-null object
21  ValueFlag                            231924 non-null object
22  Value                                231924 non-null int64
dtypes: int64(2), object(21)
memory usage: 40.7+ MB

```

Write to excel

```

In [41]: with pd.ExcelWriter("merchandise_dataset(cleaned).xlsx") as writer:
          df.to_excel(writer, sheet_name= "raw data", index= False )

```

Exploratory Data Analysis

```

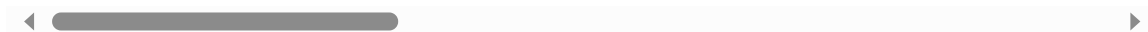
In [22]: df.head()

```


Out[22]:

	IndicatorCategory	IndicatorCode	Indicator	ReporterCode	ReporterISO3A	Rep
0	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	533	ABW	Nether with re
1	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	320	GTM	Guat
2	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	528	NLD	Nether
3	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	296	KIR	K
4	Merchandise trade values	ITS_MTV_AM	Merchandise imports by product group - annual	624	GNB	Gu I

5 rows × 23 columns



Total Import/Export by country

In [23]:

```
# pivot table
imp_exp_by_country= df.pivot_table(index= 'Reporter'
                                     , values='Value'
                                     , columns= 'Indicator'
                                     , aggfunc= 'sum'
                                     , margins= True
                                     , margins_name= "Grand Total").reset_index()

# rename columns
imp_exp_by_country.columns = [ 'Country', 'Total Exports', 'Total Imports', 'Gra

imp_exp_by_country
```

Out[23]:

	Country	Total Exports	Total Imports	Grand Total
0	Afghanistan	39792	312386	352178
1	Africa	13462209	14270774	27732983
2	Africa, CIS and Middle East	48413334	38059208	86472542
3	African, Caribbean and Pacific States (ACP)	10678778	11427626	22106404
4	Albania	120045	347568	467613
...
278	Yemen, People's Democratic Republic	10633	29902	40535
279	Yugoslavia, Socialist Federal Republic of	332224	444411	776635
280	Zambia	376182	459804	835986
281	Zimbabwe	266599	477066	743665
282	Grand Total	5900925643	5966192375	11867118018

283 rows × 4 columns

Total Import/Export by year

```
In [24]: # pivot table
imp_exp_by_year= df.pivot_table(index= 'Year',
                                values='Value',
                                columns= 'Indicator',
                                aggfunc= 'sum',
                                margins= True,
                                margins_name= "Grand Total").reset_index()

# rename columns
imp_exp_by_year.columns = [ 'Year', 'Total Exports', 'Total Imports', 'Grand Total' ]

imp_exp_by_year
```

Out[24]:

	Year	Total Exports	Total Imports	Grand Total
0	1948	250799	256246	507045
1	1949	248169	255410	503579
2	1950	260007	264306	524313
3	1951	347149	365669	712818
4	1952	330275	362780	693055
...
73	2021	299960421	302961902	602922323
74	2022	333863435	343739834	677603269
75	2023	323092774	326921799	650014573
76	2024	194059115	195422772	389481887
77	Grand Total	5900925643	5966192375	11867118018

78 rows × 4 columns

Total Import/Export by Product

```
In [25]: imp_exp_by_product= df.pivot_table(index= 'Product'
                                             , values='Value'
                                             , columns= 'Indicator'
                                             , aggfunc= 'sum'
                                             , margins= True
                                             , margins_name= "Grand Total").reset_index()
imp_exp_by_product.columns = [ 'Product Category', 'Total Exports', 'Total Imports', 'Grand Total']
imp_exp_by_product
```

Out[25]:

	Product Category	Total Exports	Total Imports	Grand Total
0	Agricultural products	108608284	110499425	219107709
1	Automotive products	92593791	86607956	179201747
2	Chemicals	129680610	127821235	257501845
3	Clothing	27796514	29243911	57040425
4	Electronic data processing and office equipment	33933440	36106223	70039663
5	Food	90710679	91486338	182197017
6	Fuels	120611110	140284412	260895522
7	Fuels and mining products	167826332	186935251	354761583
8	Integrated circuits and electronic components	31692682	37295308	68987990
9	Iron and steel	28549064	28995514	57544578
10	Machinery and transport equipment	381195553	375574055	756769608
11	Manufactures	775251773	761964580	1537216353
12	Office and telecom equipment	113119555	124353176	237472731
13	Pharmaceuticals	37473561	35145012	72618573
14	Telecommunications equipment	37688151	40700961	78389112
15	Textiles	19402119	18730565	38132684
16	Total merchandise	3584234419	3622814090	7207048509
17	Transport equipment	120558006	111634363	232192369
18	Grand Total	5900925643	5966192375	11867118018

Top 5 Product Section Globally

```
In [26]: top_5_global_products = df.groupby('Product')['Value'].sum().sort_values(ascending=False)

print(" Top 5 Product Section Globally")
print("-----")
print(top_5_global_products)
```

```
Top 5 Product Section Globally
-----
Product
Total merchandise      7207048509
Manufactures           1537216353
Machinery and transport equipment  756769608
Fuels and mining products  354761583
Fuels                  260895522
Name: Value, dtype: int64
```

Top 5 Product Section for India

```
In [27]: Top5_Indian_Products= df[df['Reporter'] == 'India'].groupby('Product')['Value'].
Top5_Indian_Products

print("Top 5 Indian Product Section")
print("-----")
print(Top5_Indian_Products)
```

Top 5 Indian Product Section

Product	
Total merchandise	15915479
Manufactures	7902265
Fuels and mining products	4461915
Fuels	3666282
Machinery and transport equipment	2598754

Name: Value, dtype: int64

Country with highest imports from India in 2022

```
In [28]: result= df[ (df['Reporter'] == 'India') &
                    (df['Indicator'].str.contains('exports')) &
                    (df['Year'] == 2022) ].groupby('Partner')['Value'].sum().sort_value

print("Country with highest imports from India in 2022:")
print("-----")
print(result)
```

Country with highest imports from India in 2022:

Partner	
World	1349161

Name: Value, dtype: int64

India's export trend for Machinery from 2015–2022

```
In [29]: # Filter data for India
india_exp = df[(df['Reporter'] == 'India') &
               df['Product'].str.contains('Machinery', case = False) &
               df['Year'].between(2015, 2022)
              ]

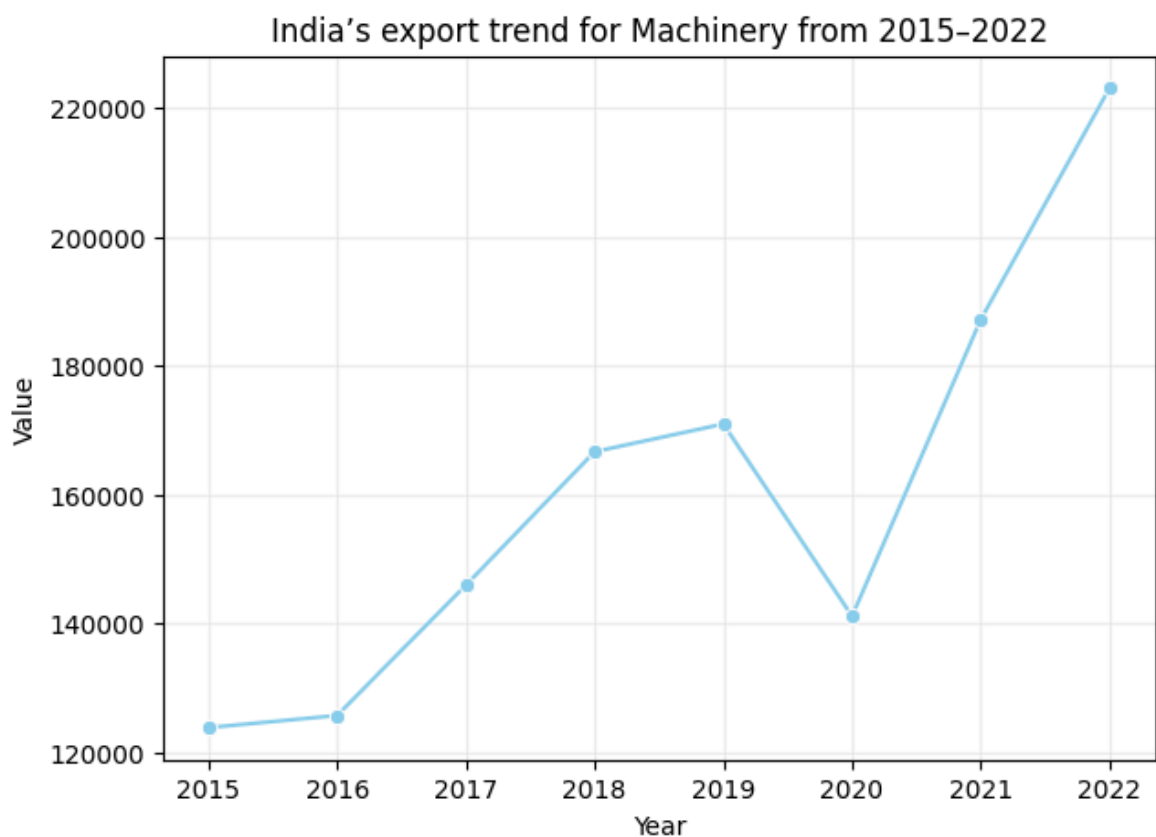
# Group the data by year and values
india_exp_trend = india_exp.groupby('Year')['Value'].sum().reset_index()
india_exp_trend
```

Out[29]:

	Year	Value
0	2015	123770
1	2016	125650
2	2017	145876
3	2018	166603
4	2019	170907
5	2020	141078
6	2021	187156
7	2022	223075

```
In [40]: # Plot the line chart
plt.figure(figsize=(7, 5))
sns.lineplot(data= india_exp_trend, x= 'Year', y= 'Value', color= 'skyblue', mar

plt.title("India's export trend for Machinery from 2015-2022")
plt.grid(alpha=0.2)
```



Observation

From 2015 to 2022, India's machinery exports showed an overall upward trend, with steady growth until 2019, a noticeable dip in 2020 likely due to pandemic disruptions, and a sharp recovery afterward, reaching their highest value in 2022.

Top 5 export commodities for India in 2022.

```
In [31]: # filter data for india
india_df= df[ (df['Reporter'] == 'India') &
              (df['Indicator'].str.contains('exports')) &
              (df['Year'] == 2022)
            ]

# group by commodities and export values
Top5_commodities = india_df.groupby('Product')['Value'].sum().reset_index().sort
Top5_commodities
```

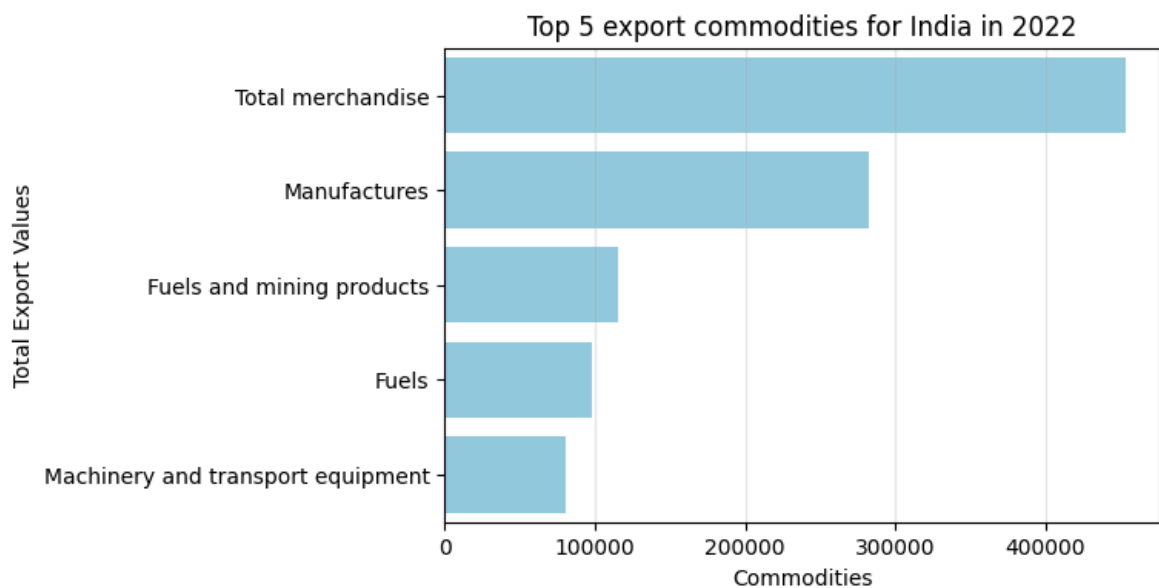
```
Out[31]:
```

	Product	Value
16	Total merchandise	453415
11	Manufactures	281864
7	Fuels and mining products	115338
6	Fuels	98470
10	Machinery and transport equipment	80798

```
In [37]: # plot bar chart

plt.figure(figsize=(6, 4))
sns.barplot(data= Top5_commodities, y= 'Product', x='Value', color='skyblue')

plt.title("Top 5 export commodities for India in 2022")
plt.xlabel("Commodities")
plt.ylabel("Total Export Values")
plt.grid(axis= 'x',alpha= 0.3)
```



Observation

In 2022, Total merchandise accounted for the highest export value from India, followed by Manufactures. Fuels and mining products and Fuels contributed moderate shares,

while Machinery and transport equipment had the lowest among the top five. This distribution suggests that India's export economy in 2022 was heavily driven by merchandise and manufactured goods, with energy-related products also playing a significant role.

In []: