In [13]:

```python
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
```

In [14]:

```python
df = pd.read_csv("co2_emission.csv")
```

In [15]:

```python
df.head()
```

Out[15]:

	Entity	Code	Year	Annual CO$_2$ emissions (tonnes)
0	Afghanistan	AFG	1949	14656.0
1	Afghanistan	AFG	1950	84272.0
2	Afghanistan	AFG	1951	91600.0
3	Afghanistan	AFG	1952	91600.0
4	Afghanistan	AFG	1953	106256.0

In [16]:

```python
df.sample(10)
```

Out[16]:

	Entity	Code	Year	Annual CO$_2$ emissions (tonnes)
20739	Zambia	ZMB	2010	2562578.27
15276	Portugal	PRT	1952	5353104.00
6550	Ethiopia	ETH	1959	336998.85
672	Americas (other)	NaN	1894	16751808.00
10171	Ireland	IRL	2017	39738353.97
11957	Maldives	MDV	2004	666848.00
6312	El Salvador	SLV	1963	1124521.13
20015	Uzbekistan	UZB	1960	47851513.16
5823	Dominican Republic	DOM	2017	21268776.28
19936	Uruguay	URY	1940	84272.00

In [17]:

```python
df.rename(columns={"Entity":"Country", "Annual CO₂ emissions (tonnes )": "CO2 Emission (tonnes)"},
inplace=True)
```

In [18]:

```python
def missing_value_of_data(df):
    total=df.isnull().sum().sort_values(ascending=False)
    percentage=round(total/df.shape[0]*100,2)
    return pd.concat([total,percentage],axis=1,keys=['Total','Percentage'])

missing_value_of_data(df)
```

Out[18]:

	Total	Percentage
Code	2207	10.58
CO2 Emission (tonnes)	0	0.00
Year	0	0.00
Country	0	0.00

In [19]:

```
df.head().T
```

Out[19]:

	0	1	2	3	4
Country	Afghanistan	Afghanistan	Afghanistan	Afghanistan	Afghanistan
Code	AFG	AFG	AFG	AFG	AFG
Year	1949	1950	1951	1952	1953
CO2 Emission (tonnes)	14656	84272	91600	91600	106256

In [5]:

```
df["Country"].unique()
```

Out[5]:

```
array(['Afghanistan', 'Africa', 'Albania', 'Algeria', 'Americas (other)',
       'Andorra', 'Angola', 'Anguilla', 'Antarctic Fisheries',
       'Antigua and Barbuda', 'Argentina', 'Armenia', 'Aruba',
       'Asia and Pacific (other)', 'Australia', 'Austria', 'Azerbaijan',
       'Bahamas', 'Bahrain', 'Bangladesh', 'Barbados', 'Belarus',
       'Belgium', 'Belize', 'Benin', 'Bermuda', 'Bhutan', 'Bolivia',
       'Bonaire Sint Eustatius and Saba', 'Bosnia and Herzegovina',
       'Botswana', 'Brazil', 'British Virgin Islands', 'Brunei',
       'Bulgaria', 'Burkina Faso', 'Burundi', 'Cambodia', 'Cameroon',
       'Canada', 'Cape Verde', 'Cayman Islands',
       'Central African Republic', 'Chad', 'Chile', 'China',
       'Christmas Island', 'Colombia', 'Comoros', 'Republic of the Congo',
       'Cook Islands', 'Costa Rica', "Cote d'Ivoire", 'Croatia', 'Cuba',
       'Curacao', 'Cyprus', 'Czech Republic', 'Czechoslovakia',
       'Democratic Republic of Republic of the Congo', 'Denmark',
       'Djibouti', 'Dominica', 'Dominican Republic', 'EU-28', 'Ecuador',
       'Egypt', 'El Salvador', 'Equatorial Guinea', 'Eritrea', 'Estonia',
       'Ethiopia', 'Europe (other)', 'Faeroe Islands', 'Falkland Islands',
       'Fiji', 'Finland', 'France', 'French Guiana', 'French Polynesia',
       'Gabon', 'Gambia', 'Georgia', 'Germany', 'Ghana', 'Gibraltar',
       'Greece', 'Greenland', 'Grenada', 'Guadeloupe', 'Guatemala',
       'Guinea', 'Guinea-Bissau', 'Guyana', 'Haiti', 'Honduras',
       'Hong Kong', 'Hungary', 'Iceland', 'India', 'Indonesia',
       'International transport', 'Iran', 'Iraq', 'Ireland', 'Israel',
       'Italy', 'Jamaica', 'Japan', 'Jordan', 'Kazakhstan', 'Kenya',
       'Kiribati', 'Kuwait', 'Kyrgysztan', 'Kyrgyzstan', 'Laos', 'Latvia',
       'Lebanon', 'Lesotho', 'Liberia', 'Libya', 'Liechtenstein',
       'Lithuania', 'Luxembourg', 'Macao', 'Macedonia', 'Madagascar',
       'Malawi', 'Malaysia', 'Maldives', 'Mali', 'Malta',
       'Marshall Islands', 'Martinique', 'Mauritania', 'Mauritius',
       'Mexico', 'Micronesia (country)', 'Middle East', 'Moldova',
       'Mongolia', 'Montenegro', 'Montserrat', 'Morocco', 'Mozambique',
       'Myanmar', 'Namibia', 'Nauru', 'Nepal', 'Netherlands',
       'New Caledonia', 'New Zealand', 'Nicaragua', 'Niger', 'Nigeria',
       'Niue', 'North Korea', 'Norway', 'Oman', 'Pakistan', 'Palau',
       'Palestine', 'Panama', 'Papua New Guinea', 'Paraguay', 'Peru',
       'Philippines', 'Poland', 'Portugal', 'Qatar', 'Reunion', 'Romania',
       'Russia', 'Rwanda', 'Saint Helena', 'Saint Kitts and Nevis',
       'Saint Lucia', 'Saint Pierre and Miquelon',
       'Saint Vincent and the Grenadines', 'Samoa',
       'Sao Tome and Principe', 'Saudi Arabia', 'Senegal', 'Serbia',
       'Seychelles', 'Sierra Leone', 'Singapore',
       'Seychelles', 'Sierra Leone', 'Singapore',
       'Sint Maarten (Dutch part)', 'Slovakia', 'Slovenia',
       'Solomon Islands', 'Somalia', 'South Africa', 'South Korea',
       'South Sudan', 'Spain', 'Sri Lanka', 'Statistical differences',
       'Sudan', 'Suriname', 'Swaziland', 'Sweden', 'Switzerland', 'Syria',
       'Taiwan', 'Tajikistan', 'Tanzania', 'Thailand', 'Timor', 'Togo',
       'Tonga', 'Trinidad and Tobago', 'Tunisia', 'Turkey',
       'Turkmenistan', 'Turks and Caicos Islands', 'Tuvalu', 'Uganda',
       'Ukraine', 'United Arab Emirates', 'United Kingdom',
       'United States', 'Uruguay', 'Uzbekistan', 'Vanuatu', 'Venezuela',
       'Vietnam', 'Wallis and Futuna Islands', 'World', 'Yemen', 'Zambia',
       'Zimbabwe'], dtype=object)
```

In [20]:

```
eu = ["Austria","Belgium","Bulgaria","Croatia","Cyprus","Czech Republic",
          "Denmark","Estonia","Finland","France","Germany","Greece","Hungary",
          "Ireland","Italy","Latvia","Lithuania","Luxembourg","Malta","Netherlands",
          "Poland","Portugal","Romania","Slovakia","Slovenia","Spain","Sweden"]
middle_east = ["United Arab Emirates", "Turkey", "Saudi Arabia", "Iran", "Iraq", "Israel", "Yemen",
"Qatar"]
```

In [21]:

```
df_eu = df[df["Country"] == eu[0]]
for i in range(1, len(eu)):
    df_eu = df_eu.append(df[df["Country"] == eu[i]])

df_middleast = df[df["Country"] == middle_east[0]]
for i in range(1,len(middle_east)):
    df_middleast = df_middleast.append(df[df["Country"] == middle_east[i]])
```

In [23]:

```
eu_total = df_eu["CO2 Emission (tonnes)"][df_eu["Year"] >= 2010].sum()
usa_total = df["CO2 Emission (tonnes)"][(df["Country"] == "United States") & (df["Year"] > 2010)].s
um()
china_total = df["CO2 Emission (tonnes)"][(df["Country"] == "China") & (df["Year"] > 2010)].sum()
india_total = df["CO2 Emission (tonnes)"][(df["Country"] == "India") & (df["Year"] > 2010)].sum()
russia_total = df["CO2 Emission (tonnes)"][(df["Country"] == "Russia") & (df["Year"] > 2010)].sum()
japan_total = df["CO2 Emission (tonnes)"][(df["Country"] == "Japan") & (df["Year"] > 2010)].sum()
middleast_total = df_middleast["CO2 Emission (tonnes)"][df_middleast["Year"] >= 2010].sum()
```

In [27]:

```
countries_total = {"EU":[eu_total], "USA":[usa_total],
                   "China":[china_total], "India":[india_total], "Russia":[russia_total],
                   "Japan":[japan_total], "Middle East":[middleast_total]}
columns = ["EU", "USA", "China", "India", "Russia", "Japan", "Middle East"]
```

In [28]:

```
df2 = pd.DataFrame(data=countries_total)
df2 = df2.transpose()
```

In [29]:

```
df2
```

Out[29]:

	0
EU	2.563225e+10
USA	3.802700e+10
China	6.789869e+10
India	1.512696e+10
Russia	1.180438e+10

	0
Japan	8.774771e+09
Middle East	1.684205e+10

In [30]:

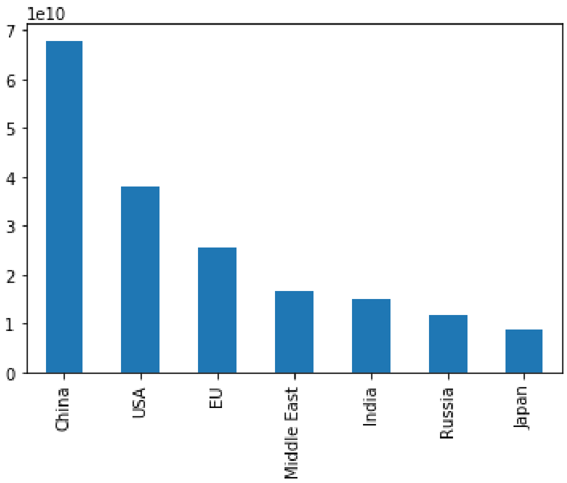
```
df2.rename(columns={0:"Total CO2 Emission"}, inplace=True)
```

In [31]:

```
df2["Total CO2 Emission"].sort_values(ascending=False).plot(kind="bar")
```

Out[31]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1be8a97bc48>
```



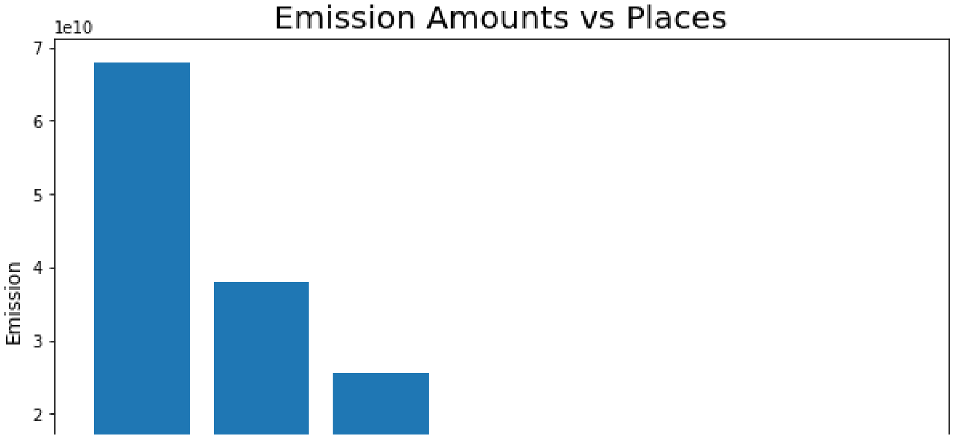
In [32]:

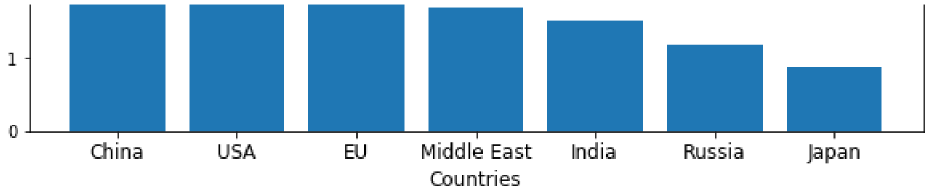
```
import plotly.express as px
```

In [33]:

```
def barchart(df):
    fig = plt.figure(figsize=(10,6))
    y = df2["Total CO2 Emission"].sort_values(ascending=False)
    x_labels =[y.index[i] for i in range(len(y))]
    x =[1, 2, 3, 4, 5, 6, 7]
    plt.bar(x,y)
    plt.xticks(x,x_labels,fontsize=12)
    plt.xlabel("Countries",fontsize=12)
    plt.ylabel("Emission",fontsize=12)
    plt.title("Emission Amounts vs Places", fontsize=20)
    plt.show()

barchart(df)
```





In [34]:

```
fig = px.bar(df2, x=["China", "USA", "EU", "Middle East", "India", "Russia","Japan",], y=df2["Total CO2 Emission"].sort_values(ascending=False), title="Emission Amount vs Regions")
fig.show()
```

In [35]:

```
df.sort_values(by="CO2 Emission (tonnes)")
```

Out[35]:

	Country	Code	Year	CO2 Emission (tonnes)
17627	Statistical differences	NaN	2000	-6.255223e+08
17620	Statistical differences	NaN	1993	-6.039042e+08
17623	Statistical differences	NaN	1996	-6.014068e+08
17617	Statistical differences	NaN	1990	-5.814990e+08
17626	Statistical differences	NaN	1999	-5.657580e+08
...	...	...	...	...
20615	World	OWID_WRL	2013	3.520789e+10
20617	World	OWID_WRL	2015	3.546275e+10
20616	World	OWID_WRL	2014	3.550583e+10
20618	World	OWID_WRL	2016	3.567510e+10
20619	World	OWID_WRL	2017	3.615326e+10

20853 rows × 4 columns

In [36]:

```
df_top = df[(df["CO2 Emission (tonnes)"] > 600000000) & (df["Year"]>1990)]
drops = df_top[df_top["Country"] == "World"].index.append(df_top[df_top["Country"] == "International transport"].index)
df_top = df_top.drop(drops)
fig = px.line(df_top, x='Year', y='CO2 Emission (tonnes)', color='Country', title="Emission Amounts After 1990 by Country")
fig.show()
```

In [ ]: