Import pandas and read dataset

In [93]:
```python
import pandas as pd
df = pd.read_excel('WorldCO2.xls', sheet_name='Data', header=None)
df.head()
```

Out[93]:

	0	1	2	3	4	5	6	7
0	Data Source	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Last Updated Date	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	Country Name	1960.000	1961.000	1962.000	1963.000	1964.000	1965.000	1966.000
4	Aruba	11092.675	11576.719	12713.489	12178.107	11840.743	10623.299	9933.903

5 rows × 62 columns

Normalize dataset

The dataset file is not normalize. We should apply some functions from pandas to normalize this file and can work with the dataset.

In [94]:
```python
df = df.drop(range(3)) # drop blank rows
df.columns = df.iloc[0] # make the first row like columns
df = df[1:] # drop the before first row
df = df.drop(columns=[2016.0, 2017.0, 2018.0, 2019.0, 2020.0]) # drop year to predi
df = df.reset_index(drop=True) # reset index
df = pd.melt(df, id_vars=['Country Name'], var_name='Year', value_name='Pollution')
df.head()
```

Out[94]:

	Country Name	Year	Pollution
0	Aruba	1960.0	11092.675
1	Afganistán	1960.0	414.371
2	Angola	1960.0	550.050
3	Albania	1960.0	2024.184
4	Andorra	1960.0	NaN

In [95]:
```python
# Select a country  "China"
df = df[df['Country Name'] == 'China'].reset_index(drop=True)
```

```
df.tail()
```

Out[95]:

	Country Name	Year	Pollution
51	China	2011.0	9.733538e+06
52	China	2012.0	1.002857e+07
53	China	2013.0	1.025801e+07
54	China	2014.0	1.029193e+07
55	China	2015.0	1.014500e+07

Definates a function to create any linear regression function

In [96]:

```
# This is a superior order function. Receives a dataframe,  column name of var x an
# and return linear regression function for the select dataset

def linear_regression_creator(df:pd.DataFrame, var_x:str, var_y:str):
    """We definate this function to create a linear regression. Justo to give it da
       and column name of variable y"""
    sum_xy = sum(df[var_x]*df[var_y])
    sum_x = sum(df[var_x])
    sum_y = sum(df[var_y])
    n = len(df[var_x])
    sum_x2 = sum(df[var_x]*df[var_x])
    sum2_x = sum(df[var_x])**2

    def linear_regression(x:float) -> float:
        """Función de regresión lineal, recibe una variable x y devuelve una variab
        beta_1 = (n*sum_xy-sum_x*sum_y)/(n*sum_x2-sum2_x)
        beta_0 = (sum_y - beta_1*sum_x)/n
        return beta_0 + beta_1*x
    return linear_regression
```

Make forecasting and show results

In [97]:

```
# We use the function defined above
linear_regression = linear_regression_creator(df, 'Year', 'Pollution')

# Select prectidion years
years = [2016, 2017, 2018, 2019, 2020]

# Apply linear regression model to each prediction year and save an list
pollution_predictions = [linear_regression(year) for year in years]

# Print predictions
predictions = {key:value for key, value in zip(years, pollution_predictions)}
predictions
```

Out[97]:

```
{2016: 7843116.383862317,
 2017: 8005665.693688929,
 2018: 8168215.003515542,
 2019: 8330764.313342154,
 2020: 8493313.623168766}
```

Append predictions to dataset and show graphics

In [98]:

```
rows = [[country_name, year, pollution] for country_name, year, pollution in zip(["
for row in rows:
    df.loc[-1] = row# adding a row
    df.index = df.index + 1
df = df.reset_index(drop=True)
df.tail()
```

Out[98]:

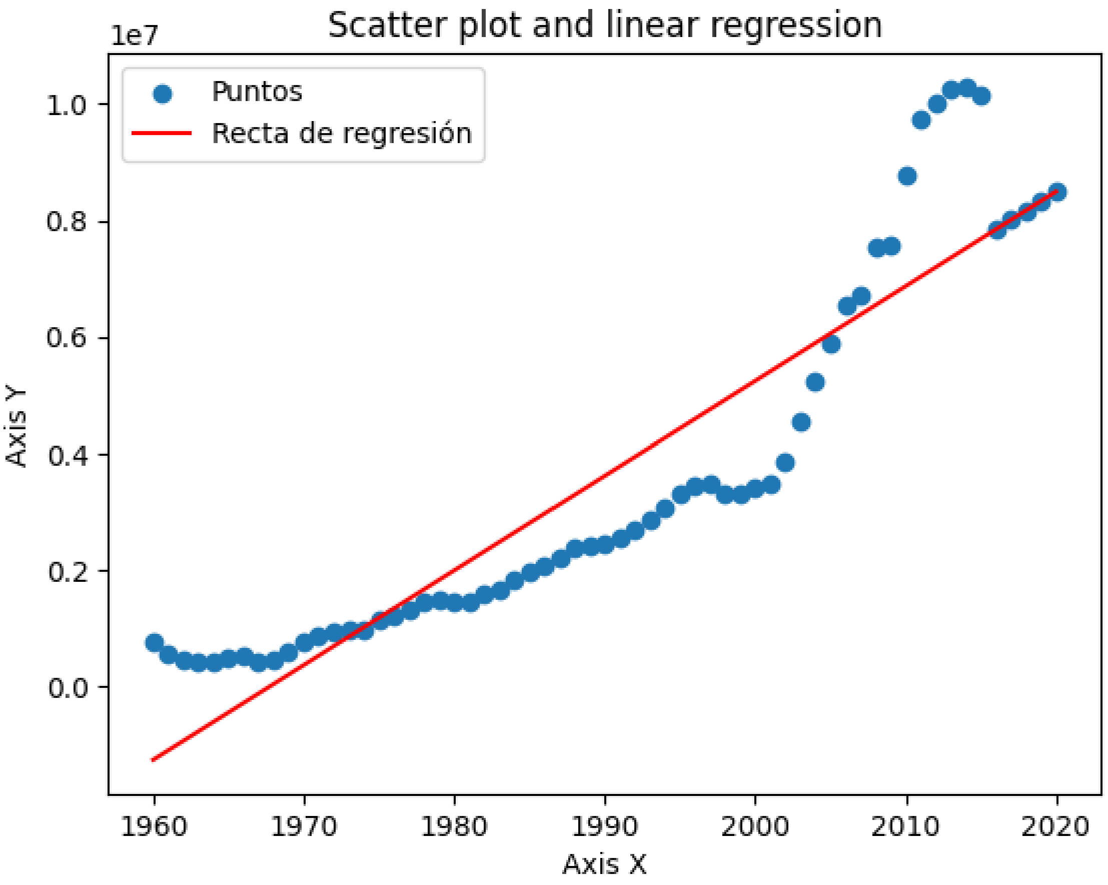
	Country Name	Year	Pollution
56	China	2016	7.843116e+06
57	China	2017	8.005666e+06
58	China	2018	8.168215e+06
59	China	2019	8.330764e+06
60	China	2020	8.493314e+06

In [99]:

```
import matplotlib.pyplot as plt
def show_linear_regression(df, var_x, var_y):
    plt.scatter(df[var_x], df[var_y], label='Puntos')
    plt.plot(df[var_x], df[var_x].apply(linear_regression), color='red', label='Rec

    plt.xlabel('Axis X')
    plt.ylabel('Axis Y')
    plt.title('Scatter plot and linear regression')
    plt.legend()
    plt.show()

show_linear_regression(df, 'Year', 'Pollution')
```


Scatter plot and linear regression
1e7
Puntos
Recta de regresión
1.0
0.8
0.6
0.4
0.2
0.0
Axis Y
1960
1970
1980
1990
2000
2010
2020
Axis X