Assignment

You will anaylize a dataset of the global average sea level change since 1880. You will use the data to predict the sea level change through year 2050.

Use the data to complete the following tasks:

- Use Pandas to import the data from `epa-sea-level.csv`.
- Use matplotlib to create a scatter plot using the "Year" column as the x-axis and the "CSIRO Adjusted Sea Level" column as the y-axix.
- Use the `linregress` function from `scipi.stats` to get the slope and y-intercept of the line of best fit. Plot the line of best fit over the top of the scatter plot. Make the line go through the year 2050 to predict the sea level rise in 2050.
- Plot a new line of best fit just using the data from year 2000 through the most recent year in the dataset. Make the line also go through the year 2050 to predict the sea level rise in 2050 if the rate of rise continues as it has since the year 2000.
- The x label should be "Year", the y label should be "Sea Level (inches)", and the title should be "Rise in Sea Level".

Unit tests are written for you under `test_module.py`.

Development

For development, you can use `main.py` to test your functions. Click the "run" button and `main.py` will run.

Testing

We imported the tests from `test_module.py` to `main.py` for your convenience. The tests will run automatically whenever you hit the "run" button.

Submitting

Copy your project's URL and submit it to freeCodeCamp.

Data Source

Global Average Absolute Sea Level Change, 1880-2014 from the US Environmental Protection Agency using data from CSIRO, 2015; NOAA, 2015. https://datahub.io/core/sea-level-rise (https://datahub.io/core/sea-level-rise)

In [36]:
```
import pandas as pd
import os
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import linregress
```

In [6]:
```
os.getcwd()
```

Out[6]:
```
'C:\\Users\\ANAND\\FreeCodeCamp\\DA using python projects -Florin Pop'
```

In [9]:
```
os.chdir('C:\\Users\\ANAND\\FreeCodeCamp\\csv data')
```

In [21]:
```
df = pd.read_csv('epa-sea-level.csv')
```

Use matplotlib to create a scatter plot using the "Year" column as the x-axis and the "CSIRO Adjusted Sea Level" column as the y-axix.

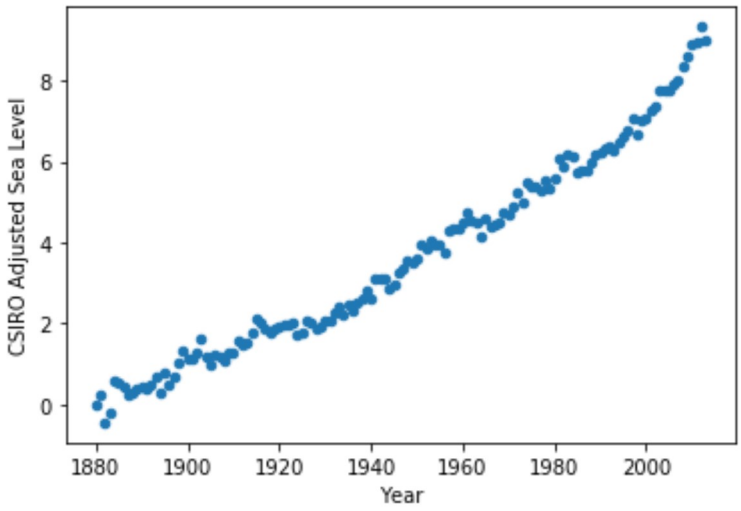
In [23]:
```
df.head()
```

Out[23]:

	Year	CSIRO Adjusted Sea Level	Lower Error Bound	Upper Error Bound	NOAA Adjusted Sea Level
0	1880	0.000000	-0.952756	0.952756	NaN
1	1881	0.220472	-0.732283	1.173228	NaN
2	1882	-0.440945	-1.346457	0.464567	NaN
3	1883	-0.232283	-1.129921	0.665354	NaN
4	1884	0.590551	-0.283465	1.464567	NaN

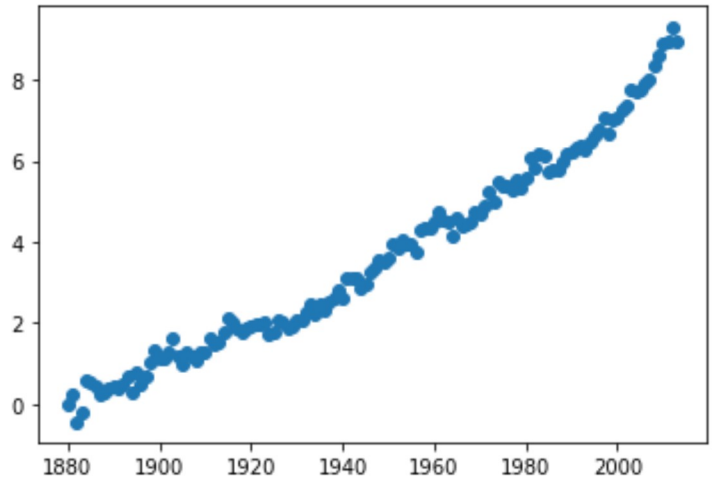
In [25]:
```
#df.plot(x = 'Year', y= 'CSIRO Adjusted Sea Level', kind = 'scatter')
```

Out[25]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a661bf0388>
```



In [82]:
```
plt.scatter(x = df['Year'], y= df['CSIRO Adjusted Sea Level'])
```

Out[82]:
```
<matplotlib.collections.PathCollection at 0x1a663c5e748>
```



Use the linregress function from scipi.stats to get the slope and y-intercept of the line of best fit. Plot the line of best fit over the top of the scatter plot. Make the line go through the year 2050 to predict the sea level rise in 2050.

In [33]:
```
slope, intercept, r_value, p_value, std_err = linregress(df['Year'],
df['CSIRO Adjusted Sea Level'])
```

In [58]:
```
m = slope
```

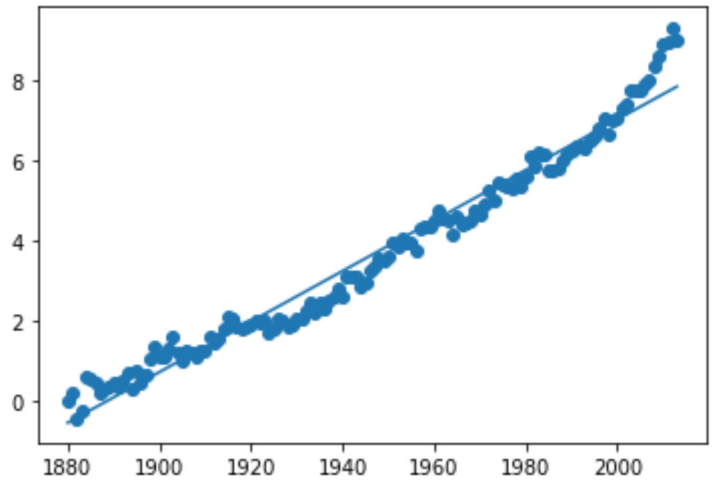
In [59]:
```
b= intercept
```

Call numpy.polyfit(x, y, deg) with x and y as arrays of data for the scatter plot and deg as 1 to calculate the slope and y-intercept of the line of best fit. Plot the linear regression line by calling matplotlib.pyplot.plot(x, eq) with x as the array of x-values and eq as the y-intercept added to the product of the slope and x ie.(x, mx +b)

In [60]:
```
#or find slope and intersect this way
m,b = np.polyfit(df['Year'],df['CSIRO Adjusted Sea Level'], deg = 1)
# deg as 1 to calculate the slope and y-intercept of the line of bes
t fit
```

In [61]:
```
plt.scatter(df['Year'],df['CSIRO Adjusted Sea Level'])
plt.plot(df['Year'], m*df['Year']+b)
```

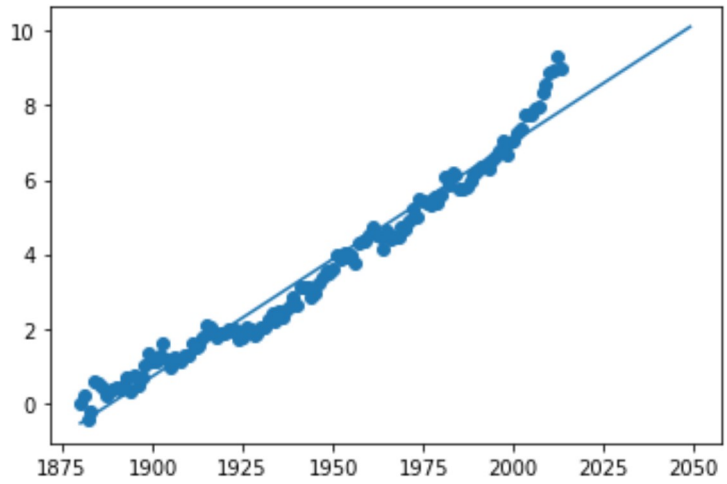
Out[61]:
```
[<matplotlib.lines.Line2D at 0x1a662882d08>]
```



In [63]:
```
#Make the line go through the year 2050 to predict the sea level ris
e in 2050
x_year = pd.Series([i for i in range(1880,2050)])
y_Adj_sea_level = pd.Series([(m*i+b) for i in x_year ])
```

In [66]:
```
plt.scatter(df['Year'],df['CSIRO Adjusted Sea Level'])
plt.plot(x_year, y_Adj_sea_level)
```

Out[66]:
```
[<matplotlib.lines.Line2D at 0x1a662993f08>]
```



Plot a new line of best fit just using the data from year 2000 through the most recent year in the dataset. Make the line also go through the year 2050 to predict the sea level rise in 2050 if the rate of rise continues as it has since the year 2000.

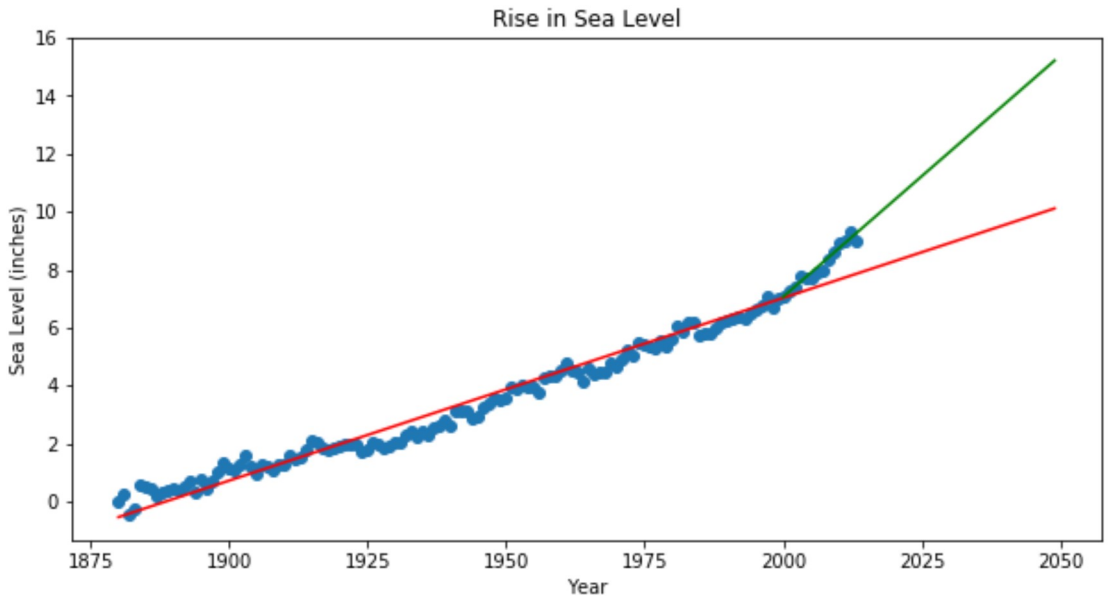
In [67]:
```
x_year1 = pd.Series([i for i in range(2000,2050)])
y_Adj_sea_level_1 = pd.Series([(m*i+b) for i in x_year1 ])
```

In [70]:
```
df_2000 = df[df['Year']>=2000]
#repeat the above process again
```

In [85]:
```
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import linregress

def draw_plot():
    # Read data from file
    df = pd.read_csv('epa-sea-level.csv')

    # Create scatter plot
    fig, ax = plt.subplots(figsize = (10,5))
    plt.scatter(x = df['Year'], y= df['CSIRO Adjusted Sea Level'])

    # Create first line of best fit
    slope, intercept, r_value, p_value, std_err = linregress(df['Yea
r'],df['CSIRO Adjusted Sea Level'])
    m = slope
    b = intercept

    x_year = pd.Series([i for i in range(1880,2050)])
    y_fit_data = pd.Series([(m*i + b) for i in x_year])

    ax2 = plt.plot(x_year, y_fit_data, 'r')

    # Create second line of best fit
    df_2000 = df[df['Year']>=2000]
    slope2, intercept2, r_value1, p_value1, std_err1 = linregress(df
_2000['Year'],df_2000['CSIRO Adjusted Sea Level'])
    m2 = slope2
    b2 = intercept2

    x_year1 = pd.Series([i for i in range(2000,2050)])
    y_fit_data1 = m2*x_year1 + b2

    #plt.scatter(df_2000['Year'],df_2000['CSIRO Adjusted Sea Level
'])
    ax3 = plt.plot(x_year1,y_fit_data1, 'g')
    # Add labels and title

    ax.set_xlabel("Year")
    ax.set_ylabel("Sea Level (inches)")
    ax.set_title("Rise in Sea Level")
    # Save plot and return data for testing (DO NOT MODIFY)
    plt.savefig('sea_level_plot.png')
    return plt.gca()
```

In [86]:
```
draw_plot()
```

Out[86]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a66271fbc8>
```



In [ ]:
```

```