HW11-part1 (50 points)

These instructions are applicable for all scripts that you will write in python3.

```
Instructions for Python3 Homework submission

#class header

#a
var1=124

#b
var2=3+var1
```

The HW11-part1 is due Sunday April 21st 11:59pm

If you submit it by Friday April 19th 11:59pm you will get 4 extra credit points.

Policy: Work on the HW problems on your own. It is not allowed to share code in any way.
The use of chatGPT is not allowed. Using the internet to find out how to solve problems is not allowed.

General Grading rules we will follow:

- We do not give points for instructions but take points off for not following them.

- We take all points off for a question if your code has syntax errors (comment out the code for partial credit).

- We grade the code, not the output of the code. Even if the output is correct, hard-cording, redundant logic, code that has no purpose in the program will not earn full credits.

- We will take points off if you use out-of-class material.

- We will take points off if you do not follow directions when a question specifies to use a certain command or write the code in a certain way.

HW11-part1 (50 points)

1. (50) *Climate Forcing* is a Climate Change Indicator that measures the r*adiative forcing*, or heating effect caused by greenhouse gases in the atmosphere. In this exercise you will analyze data on climate forcing available on the EPA website.

First, get the data set (climate-forcing_fig-1.csv) from this link:
https://www.epa.gov/climate-indicators/climate-change-indicators-climate-forcing

The data set **climate-forcing_fig-1.csv** contains data on radiative forcing caused by major long-lived greenhouse gases, from 1979 to 2019.
Radiative forcing is calculated in watts per square meter (W·m^{-2})., which represents the size of the energy imbalance in the atmosphere.

In a script called **ex11-radiative-forcing.py** do the following:

Q1 (25) Make the bar graph with error bars reported below, where the x axis reports the greenhouse gases, and the y axis reports the average and **standard error** of the radiative forcing values for each gas.

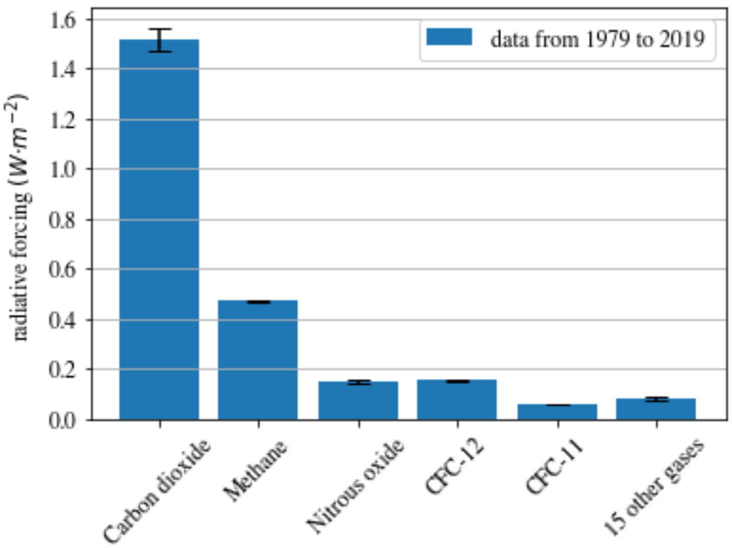
For this:

- Use pandas to read in data and calculate the average and standard deviation (σ) of the radiative forcing values for each greenhouse gas.

  To calculate the standard error use this formula (for the root mean square, use the NumPy np.sqrt())

$$SE = \frac{\sigma}{\sqrt{N}}$$

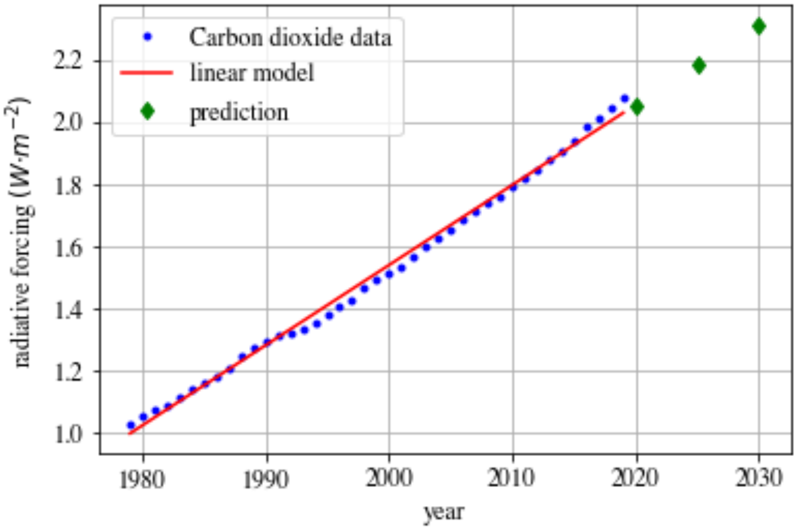
- Use the Matplotlib **bar** method to make the bar graph with error bars.
  Note: the x values are the greenhouse gases. Do not hardcode them but obtain them from the DataFrame.
  Label the y axis, make the legend, show the grid as reported in the figure. Use capsize 6 for the bars, rcParams to set the font size to 12, and latex syntax to make the m^{-2} in the y label.



Q2 (25) Fit the **Carbon Dioxide** data with a linear model, a polynomial of degree 1,

$$y = a\,x\ + b$$

Print the fitted parameters to the screen.
Calculate the R^2 and print it to the screen.
Make predictions of the radiative forcing from year 2020 to year 2030 in steps of 5.
Make this plot, which reports the data, model, and predictions.
Label the axes and include the legend.
Maintain the font size consistent with the previous plot.



Upload to Gradescope HW11-part1:
ex11-radiative-forcing.py