1. Download the file `annual.csv` from this page:

   http://data.okfn.org/data/core/global-temp

   Save the file in the same directory as a new Python file `a2.py`. Paste the following code into your Python file:

   ```
   from matplotlib import pyplot as plt
   import csv

   with open('annual.csv', 'r') as f:
       reader = csv.reader(f, delimiter=',')
       next(reader)
       rows = [r for r in reader if r[0] == 'GCAG']
   years = [row[1] for row in rows]
   temps = [row[2] for row in rows]
   plt.plot(years, temps)
   plt.title('Global Mean Temperatures')
   plt.xlabel('Year')
   plt.ylabel('Difference from 20th century average ('
       + u'\N{DEGREE SIGN}' + 'C)')
   plt.show()
   ```

   Running your file should display a graph.

2. Download the file `global.csv` from this page to the same directory:

   http://data.okfn.org/data/core/co2-fossil-global

   Using similar code, plot the total carbon emissions as a function of the year.

3. Get the data spreadsheet from this page:

   http://www.theguardian.com/news/datablog/2012/jul/22/gun-homicides-ownership-world-list#data

   In the File menu within the Google spreadsheet, choose Download as → Comma-separated values. Rename the file `guns.csv` and put it in the same directory as the others.

The following code reads the data into a list of dictionaries. The keys in the dictionaries are the column headers from the files:

```
with open('guns.csv', 'r') as f:
    reader = csv.DictReader(f, delimiter=',')
    rows = [r for r in reader
            if r['Average firearms per 100 people'] != '']
```

Here's a little function to look up a particular value in a row:

```
def rate(row):
    return float(row['Average firearms per 100 people'])
```

The code below now sorts `rows` based on that column, but in decreasing order, and then discards all but the first 20 rows.

```
decreasing = sorted(rows, key=rate, reverse=True)
decreasing = decreasing[:20]
```

Using the code on p. 39 of Grus as an example, plot a bar chart of firearm ownership per 100 people for the 20 countries with the highest ownership rates. Use the column "ISO code" to label the bars.

4. Get the file `iris.data` from here:

   http://archive.ics.uci.edu/ml/machine-learning-databases/iris/

   Rename the file `iris.csv` and put it in the same directory as the others.

   Using the code on p. 45 of Grus as an example, make a scatterplot with sepal length on the x axis and sepal width on the y axis. (You will need to look in iris.names to see what the columns in the csv file mean.)

What to hand in: a Python file `a2.py` containing code to plot all four of these graphs. You don't have to hand in the data files, since we already have them, but make sure your code expects the files to have precisely the names specified above.