Name	Date	Class	Assignment
Josiah Davis	9/23/2016	STAT243	PS02

Problem 1

My general approach for this problem was to download the data using the curl command, unzip data to a named file using the -p command, and process/filter the data usign the grep commands.

1a Download the data for apricots (i.e. the zip file)

```
curl "http://data.un.org/Handlers/DownloadHandler.ashx?DataFilter=itemCode:526&DataMartId=FAO&Format=csv&s=countryName:asc,elementCode:asc,year:desc&%20c=2,3,4,5,6,7&" -o data.zip
```

Notes:

- Hat tip to Lev Golod for figuring this out as curl -o "http://data (http://data) ...." option was not working for me

1b. Unzip the file. This will be the raw data file.

```
unzip -p data.zip > data.csv
```

1c. Extract the data for regions of the world into one file regions.csv.

```
grep "+" data.csv | tail -n +2 > regions.csv
```

1d. Extract the data for individual countries into one file countries.csv. Notes:

- Records with + sign are countries
- tail -r does a reversal which is used to eliminate the last 7 rows

  ```
  grep -v "+" data.csv | tail -n +2 | tail -r | tail -n +8 | tail -r > countries.csv
  sed 's/, / /g' countries.csv > countries2.csv
  sed 's/\"//g' countries2.csv > countries3.csv
  ```

1e. Find out how many countries are in countries.csv.

```
uniq countries3.csv | wc -l
```

1f. Subset the country-level data to the year 2005 into one file countries2005.csv.

```
grep "2005" countries3.csv > countries2005.csv
```

1g. Based on the "Area Harvested" determine the five countries in 2005 using the most land to produce apricots.

```
# Filter by area harvested | sort by the area harvested amount | take top 5 | select country column
grep 'Area Harvested' countries2005.csv | sort -t$"," -k6 -nr | head -n 5 | cut -f 1 -d ","
```

1h. Now use a bash for-loop to automate your analysis and examine the top five countries for 1965, 1975, 1985, 1995, and 2005. Have the rankings changed?

Yes, a lot has changed in the production of apricots. Two examples: American was in the top five in 1965, but not since. USSR was the top producer in 1965 and 1975 but not in the top 5 in 1995 or 2005.

```
for year in 1965 1975 1985 1995 2005
do
  echo
  echo 'Top Apricots Producers in' $year
  echo --------------------------------
  grep $year countries3.csv | grep 'Area Harvested' | sort -t$"," -k6 -nr | head -n 5 | cut -f 1 -d ","
done
```

1i. Write a bash function download_item() that takes as input a single item code

```
function download_item(){
# This function returns historical for a particular item code (e.g., 526)
curl "http://data.un.org/Handlers/DownloadHandler.ashx?DataFilter=itemCode:"$1"&DataMartId=FAO&Format=csv&s=countryName:asc,elementCode:asc,year:desc&%20c=2,3,4,5,6,7&" -o data$1.zip

  # Unzip the file and rename it
  unzip -p data$1.zip > data$1.csv

  # Print out the contents of the file
  cat data.csv
}
```

Problem 2

My approach for this problem was to I download the html file using the curl command, grep out the .txt file links, store this in a file. Next I loop through these values to download the data.

```
# Download the files
mkdir food_files
curl http://textfiles.com/food/ | grep -Eo '[-_0-9A-Za-z]+\.txt' | uniq > file_names.txt
```

```
# Store file names in their own variable
tfs=$(cat file_names.txt)
for tf in $tfs
do
  # Print out a status message
  echo
  echo ---- Downloading $tf ----
  curl -s http://textfiles.com/food/$tf > food_files/$tf
done

# How many .txt files were downloaded?
wc -l file_names.txt

# Which are the top-5 largest files (sorted by size)?
wc -c food_files/*.txt | sort -n | head -n 5

# What files contain numbers in their names (e.g. 1st_aid.txt)?
grep '[0-9]' file_names.txt

# How many files do not contain numbers in their names?
grep -v '[0-9]' file_names.txt | wc -l
```