Project 04 Answer Key

Question 1a

Import the data about the airports from the file that we used in Question 3 of Project 3, namely, `http://stat-computing.org/dataexpo/2009/airports.csv`

```
# Read in the airport data using the read.csv() function
airports = read.csv('http://stat-computing.org/dataexpo/2009/airports.csv')
```

Question 1b

Paste together the columns for the city and state, using the `paste` command.

```
# Combine the 'city' and 'state' columns into a new variable using the paste()
# function
city_and_state = paste(airports$city, airports$state)
# Use the head() function to verify that the new column was created correctly
head(city_and_state)
```

```
>>>
[1] "Bay Springs MS"      "Livingston TX"       "Colorado Springs CO"
[4] "Perry NY"            "Hilliard FL"         "Belmont MS"
```

Question 1c

Use the `table`, `sort`, and `tail` commands to find the 4 cities that have the most airports. (Ignore the missing data `NA NA`.)

```
# Since city names are not unique to each state, we use the combined city and
# state variable we made in (1b)
# We can remove any entries called 'NA NA'
unique_cities = city_and_state[city_and_state != 'NA NA']
# Use the table() function to get a frequency table of cities
# Use the sort() function to sort the table by frequency
city_freq = sort(table(unique_cities))
# Use the tail() function to see the last 4 lines of the sorted frequency table
tail(city_freq, n=4)
```

```
>>>
Indianapolis IN        Miami FL     New York NY      Houston TX
              6               6               6               8
```

Question 2a

What is the mean total fare ("total amount") for a yellow taxi cab ride in June 2019?

```
# Read in the June 2019 taxi data using the read.csv() function
taxi = read.csv('/class/datamine/data/taxi/yellow/yellow_tripdata_2019-06.csv')
# Use the mean() function on the 'total_amount' column to get the average fare
# of a cab ride in June 2019
mean(taxi$total_amount)
```

```
>>>
[1] 19.74127
```

Question 2b

What is the mean total number of passengers in a New York City yellow taxi cab ride in June 2019?

```
# Use the mean() function on the 'passenger_count' column to get the average
# number of passengers per cab ride in June 2019
mean(taxi$)
```

```
>>>
[1] 1.567322
```

Question 2c

Which pickup location ID was the most popular for yellow taxi cab rides in June 2019? Which location does that correspond to in New York?

```
# Use the table() function to get a frequency table of pickup location IDs
# Use the sort() function to sort the table by frequency
PU_location_freq = sort(table(taxi$PULocationID), decreasing=TRUE)
# Use the head() function to see the first line of the sorted frequency table
head(PU_location_freq, n=1)
```

```
>>>
   237
295057
```

```
# Use the read.csv() function to read in the pickup location data
pickup_locations = read.csv('https://s3.amazonaws.com/nyc-tlc/misc/taxi+_zone_l
ookup.csv')
# Get the pickup location information for the location with an ID of 237
pickup_locations[pickup_locations$LocationID == 237, ]
```

```
>>>
    LocationID   Borough                  Zone service_zone
237        237 Manhattan Upper East Side South  Yellow Zone
```

Question 3a

Why do you think that the data is not comma-separated? What is a reason that they chose to use | instead of a comma, as a delimiter for the data? Hint: Consider, for instance, these names of donors: `myDF$NAME[9001:9050]`

```
# Read in the election data using the read.csv() function
elections = read.csv('/class/datamine/data/election/itcont2020.txt', sep='|')
# Use the head() function to look at the subset defined in the hint
hint_subset = elections$NAME[9001:9050]
head(hint_subset)
```

```
>>>
[1] SEABROOK, MARCH EDINGS MD
[2] SPIELVOGLE, TERESA L. MRS.
[3] THOENE, MICHAEL JOS MD
[4] WILSON, MODENA ELIZABETH MD
[5] SILVA, EZEQUIEL III MD
[6] BURDICK, HOYT JEFFERY MD
669007 Levels:        MOLLY, LIG COB ... ZYWOT, CHRISTINE MS.
```

Entries in the `NAME` field contain commas (i.e. Smith, John). To avoid problems reading the data in, the chosen delimiter should not be one that occurs in the data. The '|' delimiter works well in this case.

Question 3b

Paste together the columns about the city and state where the donors live. Make a new column of the data frame called `location` which contains the city and state (together) as one new column. Hint: You can use `names(myDF)` to see the names of all of the columns of the data frame.

```
# Use the paste() function to combine the CITY and STATE columns into a new
# column called 'location'
elections$location = paste(elections$CITY, elections$STATE, sep=', ')
# Use the head() function to verify that the new column was created correctly
head(elections$location)
```

```
>>>
[1] "WASHINGTON, DC"  "WASHINGTON, DC"  "BUENA PARK, CA"
[4] "LONGWOOD, FL"    "DIAMOND BAR, CA" "BUENA PARK, CA"
```

Question 3c

Which city has the largest number of donations so far, in the 2020 election season? (Do not worry about the dollar amount. Only consider the number of donations.)

```
# Use the table() function to get a frequency table of cities
# Since city names are not unique to each state, we use the combined city and
# state column we made in (3b)
# Use the sort() function to sort the table by frequency
donation_freq = sort(table(elections$location), decreasing=TRUE)
# Use the head() function to see the first line of the sorted frequency table
head(donation_freq, n=1)
```

```
>>>
NEW YORK, NY
       77999
```