Assignment #4 - Pandas Basics

👀 Click on this link: https://classroom.github.com/a/PI8nPTGv (https://classroom.github.com/a/PI8nPTGv) to accept this assignment in GitHub classroom. This will create your homework repository. Clone your new repository.

Goals

- importing data with pandas
- cleaning / preparing data with pandas
- using pandas for basic data analysis
  - displaying summary statistics
  - value counts
- consuming data from the web
- merging / joining data

Overview

This homework consists of two parts:

1. Analyzing NYC Traffic Accidents data from Janurary-August 2020
2. A data cleaning / transformation project of your choice with pandas

Part 1 - NYC Traffic Accidents from Janurary-August 2020

```
  ______
 /|_||_\`.__
(   _    _ _\
=`-(_)--(_)-'
```

ASCII art source (https://www.asciiart.eu/vehicles/cars)

Prep

- Download a csv of Traffic Accidents data from NYC from (./hw03/NYC_Accidents_2020.csv) from January-August 2020.
  - note - this 2020 data was sourced from kaggle (https://www.kaggle.com/code/pabsantos/nyc-2020-accidents-eda/data), which in turn was a snapshot of NYC Open Data's Motor Vehicle Collisions dataset (https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95)
  - Each record represents an individual collision, including the date, time and location of the accident (borough, zip code, street name, latitude/longitude), vehicles and victims involved, and contributing factors.
  - save the `csv` file into your `data/raw` directory in your repository

Starting a Notebook and General Requirements

- open up the empty notebook, `src/traffic_accidents.ipynb`, in jupyterlab / jupyter notebook
- go through the instructions below… and make sure that…
- ⚠️ for each numbered instruction, insert a markdown cell before your code that has the first line of the instruction ⚠️
  - (the number and the accompanying line of text)
  - (no need to include the details / bulleted list underneath single instruction)

Instructions

1. import `NYC_Accidents_2020.csv` as a `DataFrame`

    - bring in the csv file by using `read_csv` from pandas; don't use any keyword arguments initially
    - use a relative path as if your notebook were opened from the root of the repository if possible (`../data/raw/NYC_Accidents_2020.csv`)
    - compare the resulting `DataFrame` against opening the spreadsheet in LibreOffice, Google Sheets, Excel, Numbers, etc.
    - you *should* immediately see an issue with the import
    - use a keyword argument from the docs (https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_csv.html) to fix the issue
    - **in a markdown cell after the import, describe what fix had to be made to make the initial import usable**

2. display columns and row samples
    - show only the names of the columns
    - show the first 5 rows
    - show a random sampling of 5 rows
    - show the last 5 rows
3. describe the rows and data types
    - use any method to show:
        - each column
        - the type of each column
        - the number of non-missing values in each column
    - **in a markdown cell after displaying the column info**:
        - list out the columns that look like they have the "wrong" (or *too wide*) type
        - and next to the column name, specify what type the column should probably be
        - lastly, preview the remainder of the instructions and write out any data transformations or cleaning that you think will be necessary to complete this part of the homework
4. initial column (or row) clean-up
    - remove at least two columns
        - in a markdown cell describe why the columns should be removed
        - show evidence (with code) of why each column should be removed
    - rename or transform at least one column
        - in a markdown cell describe why the column(s) should be renamed
    - (optional) do any other clean up you deem necessary to make the following work easier
5. determine the top three streets(Use the `ON STREET NAME` column) that had the most accidents
    - it's ok to show more than 3 streets
    - show the street name and the number of accidents occurred on each street
    - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed
6. shows the number of accidents that occurred at each borough
    - ⚠️ show a visualization that allows comparison of the number of the accidents.
        - BRONX
        - BROOKLYN
        - QUEENS
        - MANHATTAN
        - everything else can fall under "other" (including missing values)
    - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed
7. calculate summary statistics for the number of persons injured in all NYC and for a couple of selected boroughs (you can choose the two boroughs)
    - use any method to calculate mean, median, percentiles (25 and 75), max, and min
    - again, pick two boroughs
        - calculate summary statistics for each borough: use any method to calculate mean, median, percentiles (25 and 75), max, and min
        - ⚠️ in a markdown cell below the calculations, compare the results
    - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed
8. what are the distributions of accidents based on the geo location (latitude & longitude)?
    - ⚠️ show a visualization that shows the accidents that occurred at each borough.

   - ⚠️ that is plot the accidents based on the geo location, where x-axis is the latitude & y-axis is the longitude. And then differentiate the points by borough(by point color).
   - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed

9. shows the covariance between each pair of the columns
   - ⚠️ choose the columns that you think are necessary & explain your choices
   - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed
10. which month did the most number of accidents occur?
    - ⚠️ document every step that you use to do this, including how the data was cleaned and/or transformed
    - the `calendar` module and `month_abbr` (https://docs.python.org/3/library/calendar.html#calendar.month_abbr) may be useful for labels
    - it's ok to show more than one month
    - optionally, visualize this data instead of simply listing the counts
    - in a markdown cell, what can you conclude about when accidents reach a lull?

Part 2 - Freeform Pandas Project

```
                              _,add8ba,
                            ,d888888888b,
                           d8888888888888b                        _,ad8ba,_
                          d888888888888888)                     ,d888888888b,
                          I8888888888888888 _________          ,8888888888888b
                __________`Y88888888888888P"""""""""""baaa,__ ,888888888888888,
            ,adP"""""""""""9888888888P""^                 ^""Y8888888888888888I
         ,a8"^           ,d888P"888P^                           ^"Y8888888888P'
       ,a8^            ,d8888'                                     ^Y8888888P'
      a88'           ,d8888P'                                        I88P"^
    ,d88'           d88888P'                                          "b,
   ,d88'           d888888'                                            `b,
  ,d88'           d888888I                                              `b,
  d88I           ,8888888'            ___                                `b,
 ,888'           d8888888          ,d88888b,              ____            `b,
 d888           ,8888888I         d88888888b,           ,d8888b,           `b
,8888           I8888888I        d8888888888I          ,88888888b           8,
I8888           88888888b       d88888888888'          8888888888b          8I
d8886           888888888       Y888888888P'           Y8888888888,        ,8b
88888b          I88888888b      `Y8888888^             `Y888888888I        d88,
Y88888b         `888888888b,      `""""^                `Y8888888P'       d888I
`888888b         88888888888b,                           `Y8888P^        d88888
 Y888888b       ,8888888888888ba,_          _______        `""^        ,d888888
 I8888888b,    ,888888888888888888ba,_     d88888888b               ,ad8888888I
 `888888888b,  I8888888888888888888888b,    ^"Y888P"^      ____.,ad88888888888I
  88888888888b,`888888888888888888888888b,     ""      ad888888888888888888888'
  8888888888888698888888888888888888888888b_,ad88ba,_,d888888888888888888888888
  88888888888888888888888888888888888888888b,`"""^ d8888888888888888888888888I
  8888888888888888888888888888888888888888888baaad888888888888888888888888888'
  Y8888888888888888888888888888888888888888888888888888888888888888888888888P
  I888888888888888888888888888888888888888888888P^  ^Y8888888888888888888888'
  `Y88888888888888888P88888888888888888888888888'     ^88888888888888888888I
   `Y8888888888888888 `8888888888888888888888888       8888888888888888888P'
    `Y888888888888888  `888888888888888888888888,     ,888888888888888888P'
     `Y88888888888888b  `88888888888888888888888I     I888888888888888888'
       "Y8888888888888b  `8888888888888888888888I     I88888888888888888'
         "Y88888888888P   `888888888888888888888b     d8888888888888888'
            ^""""""""^     `Y88888888888888888888,    888888888888888P'
                             "8888888888888888888b,   Y888888888888P^
                              `Y888888888888888888b   `Y8888888P"^
                                "Y8888888888888888P     `""""^
```

Using your data set from the previous homework, practice using 🐼 .

1. don't use the dog bites data set referenced in the course materials
2. try to write code that's different from the programs that we've done in class (it's not adequate to simply use class sample code with a different data set 🙅)

1. Reference previous assignment or write some documentation

1. Open up the empty notebook, `project.ipynb`, in jupyterlab / jupyter notebook
2. In a markdown cell, either:
    1. write a note mentioning that your data set was documented in the previous assignment
    2. if you decide to use a different data set, describe the data that you've selected using the template markdown below:

```
## About the Data

1. Name / Title: (TODO name of data set)
2. Link to Data: (TODO link to any documentation about the data that you've found )
3. Source / Origin:
        * Author or Creator: TODO
        * Publication Date: TODO
        * Publisher: TODO
        * Version or Data Accessed: TODO
4. License: (TODO name of license)
5. Can You Use this Data Set for Your Intended Use Case? (TODO answer this question)

## Format and Samples

### Overview

Format: (TODO add what file format the data is in)
Size: (TODO how large is the file in KB, MB, GB, etc. ... use finder, windows explorer for this)
Number of Records: (TODO how many rows)

### Sample of Data

TODO show a few lines of data from the actual file.  ⚠ Use "regular" Python to do this in this code
block.  Assuming that jupyter-lab was started in your root directory: with open('../data/raw/example
-data.csv', 'r')

### Fields or Column Headers

* Field/Column 1: (TODO add field name and potential type using Python types)
* Field/Column 2: (TODO same as above)
* Field/Column N: (TODO same as above)
```

2. Retrieve the data, create a DataFrame

Include the data for the graders by either:

1. downloading it and placing it into your `data/raw` directory
2. linking to it in a markdown cell

Create a data frame from the data; use any method to do this (for example `read_csv`)

3. Using the Data

In a markdown cell, describe what you'd like to use the data for:

- repeat the same analysis that you did previously in plain python, but with `pandas` instead
- perhaps you would simply like to clean it up for further analysis later

Write code to achieve what you've written out above. The code should contain at least 4 (repetition is allowed) of the following (these can overlap with your analysis above):

- 1 filling in missing values
- 1 type conversion
- 1 transform a column
- 1 create a new calculated column
- 1 visualization
- 1 calculate summary statistics
- 1 calculate value counts

In a markdown cell above your code, write out which of the above requirements you're implementing. As you write your code, document your process in an accompanying markdown cell.