Module 7: Data Wrangling with Pandas

CPE311 Computational Thinking with Python

Submitted by: Paala, Anton
Performed on: 04/07/25
Submitted on: 04/07/25

Submitted to: Engr. Roman Richard

7.1 Supplementary Activity

Using the datasets provided, perform the following exercises.

Exercise 1

We want to look at data for the Facebook, Apple, Amazon, Netflix, and Google (FAANG) stocks, but we were given each as a separate CSV file. Combine them into a single file and store the dataframe of the FAANG data as faang for the rest of the exercises:

1. Read each file in.
2. Add a column to each dataframe, called ticker, indicating the ticker symbol it is for (Apple's is AAPL, for example). This is how you look up a stock. Each file's name is also the ticker symbol, so be sure to capitalize it.

3. Append them together into as single dataframe.
4. Save the result in a CSV file called faang.csv

In [71]:
```
import pandas as pd
# Reading each file
apple_df = pd.read_csv('C:/Users/paala/CPE311 - Paala/aapl.csv')
amazon_df = pd.read_csv('C:/Users/paala/CPE311 - Paala/amzn.csv')
facebook_df = pd.read_csv('C:/Users/paala/CPE311 - Paala/fb.csv')
google_df = pd.read_csv('C:/Users/paala/CPE311 - Paala/goog.csv')
netflix_df = pd.read_csv('C:/Users/paala/CPE311 - Paala/nflx.csv')

# Assigning the column 'ticker' within each dataframes
apple_df = apple_df.assign(ticker = "AAPL")
amazon_df = amazon_df.assign(ticker = "AMZN")
facebook_df = facebook_df.assign(ticker = 'FB')
google_df = google_df.assign(ticker = 'GOOG')
netflix_df = netflix_df.assign(ticker = 'NFLX')

# Create a new dataframe variable by concatentating all dataframes
# Using the dataframe variable, using the function 'to_csv' creates a csv file
# in which all dataframes were combined. It is named as faang.csv.
merged_df = pd.concat([apple_df, amazon_df, facebook_df, google_df, netflix_df])
merged_df.to_csv('C:/Users/paala/CPE311 - Paala/faang.csv', index=False)
```

Exercise 2

- With faang, use type conversion to change the date column into a datetime and volume column into integers. Then, sort by date and ticker.
- Find the seven rows with the highest value for volume.
- Right now, the data is somewhere between long and wide format. Use melt() to make it completely long format. Hint: date and ticker are our ID variables (they uniquely identify each row). We need to melt the rest so that we don't have separate columns for open, high, low, close, and volume.

In [75]:
```
df = pd.read_csv('C:/Users/paala/CPE311 - Paala/faang.csv')
df.dtypes # Identifying the datatypes of each column
```

Out[75]:
```
date       object
open      float64
high      float64
low       float64
close     float64
volume      int64
ticker     object
dtype: object
```

In [86]:
```
# Applying type conversion in 'date' and 'volume' columns.
df['date'] = df['date'].apply(pd.to_datetime) # object datatype converted into datetime datatype
df['volume'] = df['volume'].apply(pd.to_numeric) # int64 gets converted into int64 which is kind of unneccessary
df = df.sort_values(['date', 'ticker'], ascending=[False, False]) # Sort the values from highest to lowest
df.head(7) # Highest values for volume
```

Out[86]:

	date	open	high	low	close	volume	ticker
1254	2018-12-31	260.1600	270.1001	260.0000	267.6600	13508920	NFLX
1003	2018-12-31	1050.9600	1052.7000	1023.5900	1035.6100	1493722	GOOG
752	2018-12-31	134.4500	134.6400	129.9500	131.0900	24625308	FB
501	2018-12-31	1510.8000	1520.7600	1487.0000	1501.9700	6954507	AMZN
250	2018-12-31	157.8529	158.6794	155.8117	157.0663	35003466	AAPL
1253	2018-12-28	257.9400	261.9144	249.8000	256.0800	10987286	NFLX
1002	2018-12-28	1049.6200	1055.5600	1033.1000	1037.0800	1413772	GOOG

In [83]:
```
df.melt(id_vars=['date', 'ticker'])
```

Out[83]:

	date	ticker	variable	value
0	2018-12-31	NFLX	open	2.601600e+02
1	2018-12-31	GOOG	open	1.050960e+03
2	2018-12-31	FB	open	1.344500e+02
3	2018-12-31	AMZN	open	1.510800e+03
4	2018-12-31	AAPL	open	1.578529e+02
...	...	...	...	...
6270	2018-01-02	NFLX	volume	1.096689e+07
6271	2018-01-02	GOOG	volume	1.237564e+06
6272	2018-01-02	FB	volume	1.815190e+07
6273	2018-01-02	AMZN	volume	2.694494e+06
6274	2018-01-02	AAPL	volume	2.555593e+07

6275 rows × 4 columns

Exercise 3

- Using web scraping, search for the list of the hospitals, their address and contact information. Save the list in a new csv file, hospitals.csv
- Using the generated hospitals.csv, convert the csv file into pandas dataframe. Prepare the data using the necessary preprocessing techniques.

In [108…

```
import pandas as pd

hospitals_url = 'https://shop.philcare.com.ph/accredited-hospitals' # Loading the HTML into a variable

hospitals = pd.read_html(hospitals_url) # Reading the HTML from the URL variable made prior.
```

```
hospital_df = hospitals[0] # Getting the Table first, since the tables inside the html act like a list.
hospital_df.to_csv('C:/Users/paala/CPE311 - Paala/hospitals.csv', index=False) # Saving the dataframe into a CSV file
```

In [109…

```
df = pd.read_csv('C:/Users/paala/CPE311 - Paala/hospitals.csv') # Loading the CSV file
df.dtypes
# By observing datatypes,
# we can determine that all of the columns are in object.
# However, this should not be the case with the Contact Number,
# as it could be replaced as integer.
```

Out[109…

```
Provider Name       object
Complete Address    object
City                object
Province            object
Region              object
Area                object
Contact No.         object
dtype: object
```

In [116…

```
df.shape # We can observe that there are 1873 observations, and 7 attributes.
```

Out[116…

```
(1873, 7)
```

In [120…

```
df.head() # We can observe that there are NaN values in the Contact No. columns. We can drop and fill them with zeroe
```

Out[120…

	Provider Name	Complete Address	City	Province	Region	Area	Contact No.
0	CLINICA LAGUNA MULTISPECIALTY CENTER AND DIAGN...	UNIT 207 PARIAN COMMERCE CENTER PARIAN CALAMBA...	CALAMBA CITY	LAGUNA	Region IV-A (CALABARZON)	SOUTH LUZON	NaN
1	ABELLA MIDWAY HOSPITAL	125 P. VALERO ST. BRGY. POBLACION VALENCIA CIT...	VALENCIA CITY	BUKIDNON	Region X	MINDANAO	(088) 828-3533
2	ABESAMIS EYE CARE AND CONTACT LENS CENTER (MAK...	SUITE 904 MEDICAL PLAZA MAKATI, DELA ROSA CORN...	MAKATI CITY	METRO MANILA	NCR	METRO MANILA	(02) 8556-0816
3	ACCURATE MEDICAL DIAGNOSTICS (MABALACAT BRANCH)	LOT 15 BLOCK 10 MC ARTHUR HI-WAY, MABIGA BRGY....	MABALACAT	PAMPANGA	Region III	NORTH LUZON	(045) 331-8706/(045) 893-1550
4	ACCURATE MEDICAL DIAGNOSTICS (ANGELES CITY BRA...	2442 STO. ENTIERRO ST. BRGY. STO. CRISTO ANGEL...	ANGELES CITY	PAMPANGA	Region III	NORTH LUZON	(045) 626-1823

In [122…

```
df.fillna(0) # Fill all NaN values with Zeroes.
```

Out[122...

	Provider Name	Complete Address	City	Province	Region	Area	Contact No.
0	CLINICA LAGUNA MULTISPECIALTY CENTER AND DIAGN...	UNIT 207 PARIAN COMMERCE CENTER PARIAN CALAMBA...	CALAMBA CITY	LAGUNA	Region IV-A (CALABARZON)	SOUTH LUZON	0
1	ABELLA MIDWAY HOSPITAL	125 P. VALERO ST. BRGY. POBLACION VALENCIA CIT...	VALENCIA CITY	BUKIDNON	Region X	MINDANAO	(088) 828-3533
2	ABESAMIS EYE CARE AND CONTACT LENS CENTER (MAK...	SUITE 904 MEDICAL PLAZA MAKATI, DELA ROSA CORN...	MAKATI CITY	METRO MANILA	NCR	METRO MANILA	(02) 8556-0816
3	ACCURATE MEDICAL DIAGNOSTICS (MABALACAT BRANCH)	LOT 15 BLOCK 10 MC ARTHUR HI-WAY, MABIGA BRGY....	MABALACAT	PAMPANGA	Region III	NORTH LUZON	(045) 331-8706/(045) 893-1550
4	ACCURATE MEDICAL DIAGNOSTICS (ANGELES CITY BRA...	2442 STO. ENTIERRO ST. BRGY. STO. CRISTO ANGEL...	ANGELES CITY	PAMPANGA	Region III	NORTH LUZON	(045) 626-1823
...	...	...	...	...	...	...	...
1868	CARMELA MEDICAL CENTRE INC.	14A GT . UNIT 205 B 2ND FLOOR STA. RITA CORNER...	SUBIC BAY FREEPORT Z	ZAMBALES	Region III	NORTH LUZON	(047) 222-8125; (0960) 484-9588
1869	OUR LADY OF ROSARY HOSPITAL INC.	6 TALAG STREET SAN ROQUE MACABEBE PAMPANGA	MACABEBE	PAMPANGA	Region III	NORTH LUZON	(045) 300-8522; (0963) 306-0449
1870	MADRID DIAGNOSTIC CENTER	ALAS-ASIN MARIVELES BATAAN	MARIVELES	BATAAN	Region III	NORTH LUZON	(047) 638-1925; (0995) 290-0685
1871	KIRKK DIAGNOSTIC LABORATORY	MULLIGAN GOLF DRIVING RANGE BALITI , TELABASTA...	SAN FERNANDO CITY	PAMPANGA	Region III	NORTH LUZON	(045) 455-5206; (0936) 140-2582
1872	MENDEZ SPECIALISTS MEDICAL CENTER INC.	MENDEZ-TAGAYTAY ROAD GALICIA III	MENDEZ (MENDEZ-	CAVITE	Region IV-A (CALABARZON)	SOUTH LUZON	(0920) 974-6728;(046)

	Provider Name	Complete Address	City	Province	Region	Area	Contact No.
		MENDEZ (MENDE...	NUÃ‘EZ				443-9999

1873 rows × 7 columns

In [128…
```
df = df.rename(columns={"Provider Name":'Hospital Name'}) # Renaming the column Provider Name since it will cause cor
df.columns # This confirms that the column has been changed.
```

Out[128…
```
Index(['Hospital Name', 'Complete Address', 'City', 'Province', 'Region',
       'Area', 'Contact No.'],
      dtype='object')
```

7.2 Conclusion:

In conclusion, I have learned how to concatenate multiple CSVs into one CSV file. I wonder if it is possible to make a for-loop in reading the CSV files, so that it would look more organized. Nonetheless, I have also learned another function, which is melt(). Based on my understanding, it unpivots the table by using variable IDs. However, I'm not sure if the result that I got is the correct one...

Lastly, I have learned how to web scrape by finding a list of accredited hospitals by Philcare. The table consists of the Hospital name, Full address, Contact information, City, Province, Region, and Area, which are good attributes for a table. We can use this info from the web into a CSV file, which is amazing! I have also learned some preprocessing techniques, such as identifying and observing the datatypes, column names, etc.