Project 7 -- SEYI OGUNMODEDE

Instructor: Dr. Ward

- Help with figuring out how to write a function.

Question 1

In [58]:
```
import pandas as pd
```

In [59]:
```
# The airline data has lot of columns
# You will not see all of them unless you ask it to show all the columns
# i will set my pandas to show me all of them
# from 1987-2008,this takes a very long time in pandas

pd.set_option('display.max_columns', None)
```

In [ ]:
```
# to make it faster, we can use things from the bass shell
# we could make a call to shell underneath the hood
# we could talk to the operating system directly
```

In [13]:
```
# This indicates that is not python code
# This is show code. What does it do? I am doing two things

# I am taking my first line out of my 1987 flight data
# and storing it in a new file in my home directory, Tilda (~)

# I take the first sign from 1987 dataset
# and put it in a file called INDflights.csv

# That will just only get the header from the 1987 file
# and store it in this new file i am building. You could do for any year it doesnot re

# We just want to get the header first then afterwards we will know what the column re
# Then, from all of the other files that has (*.csv) extension on them, we are going t
# Then we will end up finding the Origin and Destination columns from them.
# None of the columns is going to say IND
```

In [60]:
```
%%bash
head -n1 /anvil/projects/tdm/data/flights/subset/1987.csv >~/INDflights.csv
grep -h ",IND," /anvil/projects/tdm/data/flights/subset/*.csv >>~/INDflights.csv
```

In [61]:
```
# Then, I can load in from this new files that i built
# Also, I put a tilda (~) in the front to say this is my home directory

myDf=pd.read_csv('~/INDflights.csv')
```

```
/tmp/ipykernel_16/2542556726.py:4: DtypeWarning: Columns (10,22) have mixed types. Sp
ecify dtype option on import or set low_memory=False.
  myDf=pd.read_csv('~/INDflights.csv')
```

In [62]:
```
# There are 1589899 flights in our file altogether

myDf.shape
```

Out[62]:
```
(1589899, 29)
```

In [63]:
```
# To know how many flight deoarted from indianapolis

myDf[myDf['Origin']=='IND'].shape
```

Out[63]:
```
(796496, 29)
```

In [64]:
```
myDf[myDf['Dest']=='IND'].shape
```

Out[64]:
```
(793403, 29)
```

a. How many flights are there altogether in myDF? You can check this using myDF.shape. There are 1589899 flights in our file altogether

b. How many of the flights are departing from IND? (I.e., the Origin airport is IND.) There are 796496 flights departing from indianapolis.

c. How many of the flights are arriving to IND? (I.e., the Dest airport is IND.) There are 793403 flights ariving to indianapolis

Question 2

In [65]:
```
# for all flights departing from indiaapolis,
# we want to study the destination airport.
# and see how many times each destination airports occurs
# and display the most popular 20 of them.

myDf[myDf['Origin']=='IND']['Dest'].value_counts().head(20)
```

Out[65]:
```
ORD    77720
DTW    55974
STL    54186
ATL    48975
DFW    36523
MSP    34648
CLT    34199
DEN    29056
PIT    28033
EWR    26795
MDW    26120
MCO    25755
PHL    24492
IAH    24187
CLE    23221
CVG    22698
MEM    21645
DCA    20505
PHX    19628
LGA    18300
Name: Dest, dtype: int64
```

In [66]:
```
# for all flights departing from indiaapolis,
# we want to study the popular airlines.
# and see how many times each airline occurs
# and display the most popular 5 UniqueCarrier`s of them.

myDf[myDf['Origin']=='IND']['UniqueCarrier'].value_counts().head(5)
```

Out[66]:
```
US    192109
NW    119455
WN     94232
DL     68089
UA     62763
Name: UniqueCarrier, dtype: int64
```

In [67]:
```
myDf.head()
```

Out[67]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	UniqueCar
0	1987	10	1	4	700.0	700	804.0	755	
1	1987	10	2	5	700.0	700	805.0	755	
2	1987	10	3	6	659.0	700	757.0	755	
3	1987	10	4	7	700.0	700	756.0	755	
4	1987	10	6	2	702.0	700	806.0	755	

In [68]:
```
myDf.tail()
```

Out[68]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	Un
1589894	2008	4	16	3	1458.0	1500	1635.0	1637	
1589895	2008	4	16	3	1334.0	1220	1505.0	1354	
1589896	2008	4	16	3	1255.0	1248	1426.0	1419	
1589897	2008	4	17	4	2016.0	2015	2148.0	2148	
1589898	2008	4	17	4	958.0	1005	1137.0	1140	

In [69]:
```
myDf['UniqueCarrier'].value_counts()
```

Out[69]:
```
US         384725
NW         238138
WN         188054
DL         136138
UA         125200
AA         114151
CO         102587
TW          69317
XE          41932
MQ          34307
TZ          29336
FL          25824
HP          25407
9E          18939
DH          11429
EA          10456
OH           9044
F9           7160
EV           5276
OO           4555
PI           3217
PA (1)       2673
YV           2034
Name: UniqueCarrier, dtype: int64
```

For flights departing from 'IND' (i.e., with IND as the Origin), what are the 20 most popular destination airports (i.e., the 20 most popular Dest airports)?

For flights departing from 'IND' (i.e., with IND as the Origin), what are the 5 most popular airlines (i.e., the 5 most popular `UniqueCarrier`s)?

Question 3

In [70]:
```
def myrecords(myDf: pd.DataFrame, myairport: str)-> pd.Series:
    """
    myrecords is a function that accept myDf and myairport as arguments,
    and from all flights departing from myairport
    it returns a Series of the 20 most popular destination airports

    Arg:
        myDf(pd.DataFrame): The Data Frame that has the flight data
                            corresponding to flights departing from myairport.
```

```
        myairport(str): The Origin airport that we are studying.

    Returns:
        pd.Series: A Panda Series that contains 20 most popular destination airports.
    """
    return myDf[myDf['Origin']== myairport]['Dest'].value_counts().head(20)
    # do not forget to change the airport from 'IND' to my airport
```

In [78]:
```
myrecords(myDf,'IND' )
```

Out[78]:
```
ORD    77720
DTW    55974
STL    54186
ATL    48975
DFW    36523
MSP    34648
CLT    34199
DEN    29056
PIT    28033
EWR    26795
MDW    26120
MCO    25755
PHL    24492
IAH    24187
CLE    23221
CVG    22698
MEM    21645
DCA    20505
PHX    19628
LGA    18300
Name: Dest, dtype: int64
```

In [72]:
```
def takeoff(myDf: pd.DataFrame, myairlines: str)-> pd.Series:
    """
    takeoff is a function that accept myDf and myairlines as arguments,
    and for all flights departing myairlines
    it returns a Series of the 5 most popular departing airlines

    Arg:
        myDf(pd.DataFrame): The Data Frame that has the flight data
                            corresponding to flights departing myairlines .
        myairlines(str): The Origin airlines that we are studying.

    Returns:
        pd.Series: A Panda Series that contains 5 most popular destination airlines.
    """
    return myDf[myDf['Origin']==myairlines]['UniqueCarrier'].value_counts().head(5)
    # do not forget to change the airport from 'IND' to myairport
```

In [73]:
```
takeoff(myDf,'IND')
```

Out[73]:
```
US    192109
NW    119455
WN     94232
DL     68089
UA     62763
Name: UniqueCarrier, dtype: int64
```

Wrap your work for question 2a into a function that takes 1 data frame as an argument and the corresponding 3-letter code as an argument, and finds the 20 most popular destination airports in that data frame.

Wrap your work for question 2b into a function that takes 1 data frame as an argument and the corresponding 3-letter code as an argument, and finds the 5 most popular airlines in that data frame.

Question 4

In [ ]:
```
# First import the data from Buffallo airport into a file
```

In [88]:
```
%%bash
head -n1 /anvil/projects/tdm/data/flights/subset/1987.csv >~/BUFflights.csv
grep -h ",BUF," /anvil/projects/tdm/data/flights/subset/*.csv >>~/BUFflights.csv
```

In [89]:
```
# Then read the data into a Pandas Data Frame

myDf=pd.read_csv('~/BUFflights.csv')
```

```
/tmp/ipykernel_16/312471128.py:3: DtypeWarning: Columns (10,22) have mixed types. Spe
cify dtype option on import or set low_memory=False.
  myDf=pd.read_csv('~/BUFflights.csv')
```

In [93]:
```
def myrecords(myDf: pd.DataFrame, myairport: str)-> pd.Series:
    """
    myrecords is a function that accept myDf and myairport as arguments,
    and from all flights departing from myairport
    it returns a Series of the 20 most popular destination airports

    Arg:
        myDf(pd.DataFrame): The Data Frame that has the flight data
                            corresponding to flights departing from myairport.
        myairport(str): The Origin airport that we are studying.

    Returns:
        pd.Series: A Panda Series that contains 20 most popular destination airports.
    """
    return myDf[myDf['Origin']== myairport]['Dest'].value_counts().head(20)
    # do not forget to change the airport from 'IND' to my airport
```

In [94]:
```
myrecords(myDf,'BUF' )
```

Out[94]:
```
ORD    71736
EWR    51712
ATL    37953
LGA    36544
DTW    34581
PHL    29254
BWI    26571
PIT    24493
JFK    18745
CLT    18099
DCA    17086
BOS    15660
CLE    14424
CVG    12850
IAD    10932
MCO     7155
RDU     6669
MSP     6457
ROC     5088
ALB     5012
Name: Dest, dtype: int64
```

In [95]:
```
def takeoff(myDf: pd.DataFrame, myairlines: str)-> pd.Series:
    """
    takeoff is a function that accept myDf and myairlines as arguments,
    and for all flights departing myairlines
    it returns a Series of the 5 most popular departing airlines

    Arg:
        myDf(pd.DataFrame): The Data Frame that has the flight data
                            corresponding to flights departing myairlines .
        myairlines(str): The Origin airlines that we are studying.

    Returns:
        pd.Series: A Panda Series that contains 5 most popular destination airlines.
    """
    return myDf[myDf['Origin']==myairlines]['UniqueCarrier'].value_counts().head(5)
    # do not forget to change the airport from 'IND' to myairport
```

In [92]:
```
# Then apply the function from question 3b

takeoff(myDf,'BUF')
```

Out[92]:
```
US    173474
NW     40627
UA     39720
AA     36695
CO     34902
Name: UniqueCarrier, dtype: int64
```

In [ ]:
```
# First import the data from Jacksonville (JAX) airport into a file
```

In [96]:
```
%%bash
head -n1 /anvil/projects/tdm/data/flights/subset/1987.csv >~/JAXflights.csv
grep -h ",JAX," /anvil/projects/tdm/data/flights/subset/*.csv >>~/JAXflights.csv
```

In [97]:
```
# Then read the data into a Pandas Data Frame

myDf=pd.read_csv('~/JAXflights.csv')
```

```
/tmp/ipykernel_16/1229860190.py:3: DtypeWarning: Columns (10,22) have mixed types. Sp
ecify dtype option on import or set low_memory=False.
  myDf=pd.read_csv('~/JAXflights.csv')
```

In [98]:
```
myrecords(myDf,'JAX' )
```

Out[98]:
```
ATL    85085
CLT    44495
DFW    34485
FLL    32245
EWR    26351
BWI    22522
IAH    22073
PHL    21410
TPA    17383
CVG    16685
BNA    15295
ORD    15123
STL    14318
LGA    14264
DTW    14236
MIA    12374
IAD    10251
RDU     8726
DCA     8209
ORF     7838
Name: Dest, dtype: int64
```

In [99]:
```
takeoff(myDf,'JAX')
```

Out[99]:
```
US    99543
DL    92018
WN    76669
CO    52194
AA    36526
Name: UniqueCarrier, dtype: int64
```

Test your functions from questions 3a and 3b on Jacksonville (JAX) and Buffalo (BUF).

Question 5

In [ ]:
```

```

Markdown notes and sentences and analysis written here.

Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.