Data Analysis of Titanic Dataset in a Jupyter Notebook

Didem B. Aykurt

Colorado State University Global

MIS542; Business Analytics

Dr.Emmanuel Tsukerman

June 18, 2023

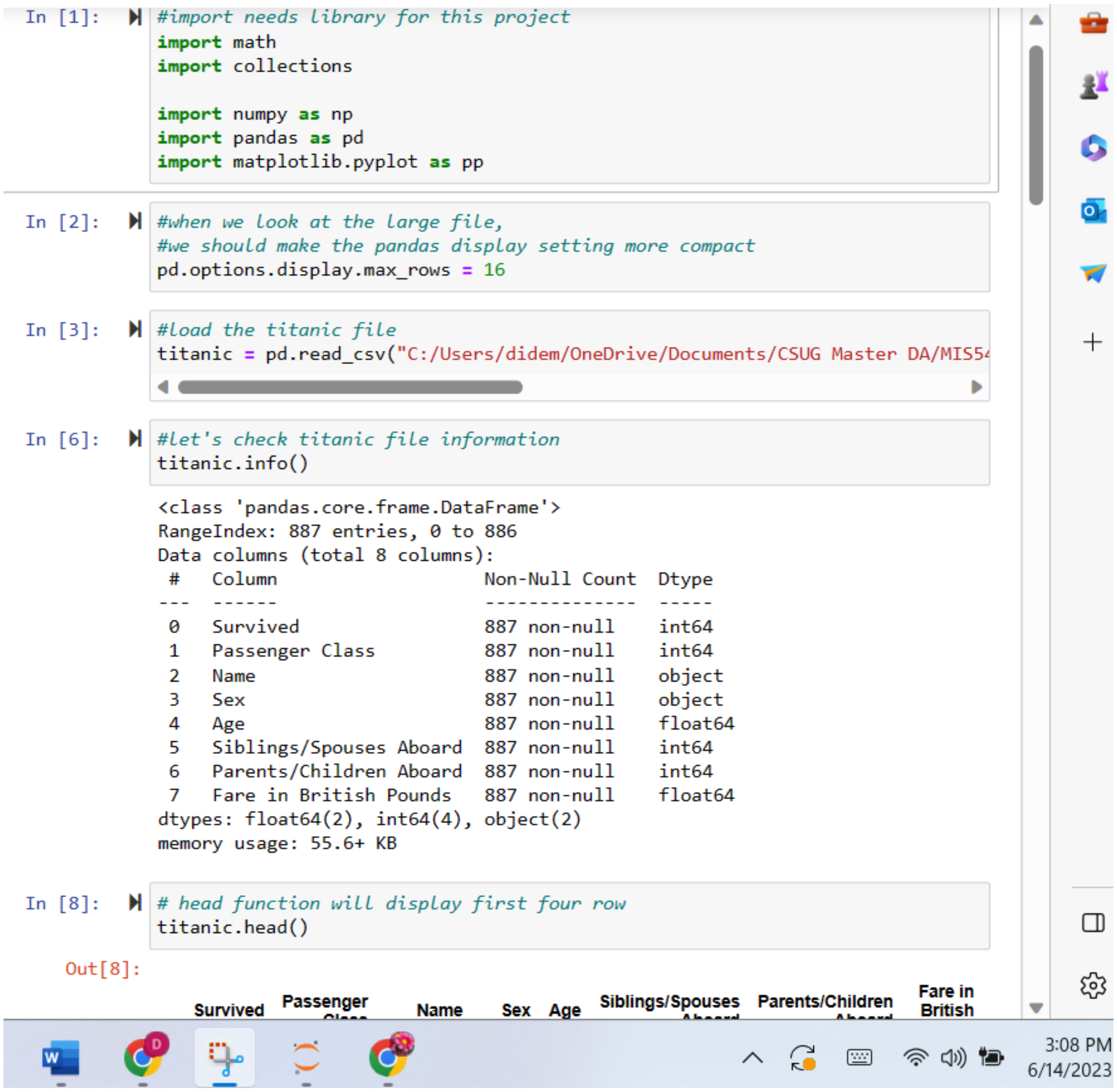
Explore Titanic Dataset with Pandas Library in Jupyter

I will use Panda's library in Python to analyze the Titanic passenger data. I want to give some minor information about Pandas as an open-source relational and labeled data library. The library has fast and high-performance properties for data structures and operations that help manipulate and analyze numerical data and time series. Easy to load different target files such as SQL database, CSV file, and Excel file from existing storage and handling of missing data, both floating point and non-floating-point data. Have access to insert and delete columns into DataFrame and set margining and joining. Capable of quickly reshaping and pivoting dataset and time-series quality. The Pandas library quickly makes groups by functionality on a dataset. Pandas have analysis functions to create graphs and charts with big and heavy data. For example, Matplotlib has a process for plotting, SciPy can statistically analyze, and sci-kit-learn can use machine learning algorithms.

The library can run any text editor efficiently, so Jupyter is a great source to execute code in a specific cell more precisely than completing the entire file. Also, Jupyter has access to visualize data frames and plots.

I work with the titanic.csv dataset that is available in CSU global sources. The Titanic dataset contains passenger detail information, and the dataset includes 887 observations with eight variables listing Survived, Passenger Class, Name, Sex, Age, Siblings/Spouses Aboard, and Parents/Children Aboard. I aim to calculate the average cost of the first class in U.S. dollars, calculate passengers over 20 with siblings onboard, and find the median age of non-survive passengers. Create a pie chart to show a group of genders, a bar chart that helps compare gender survivors, and a bar chart help to shows calculate the total number of each age group with Pandas Library.

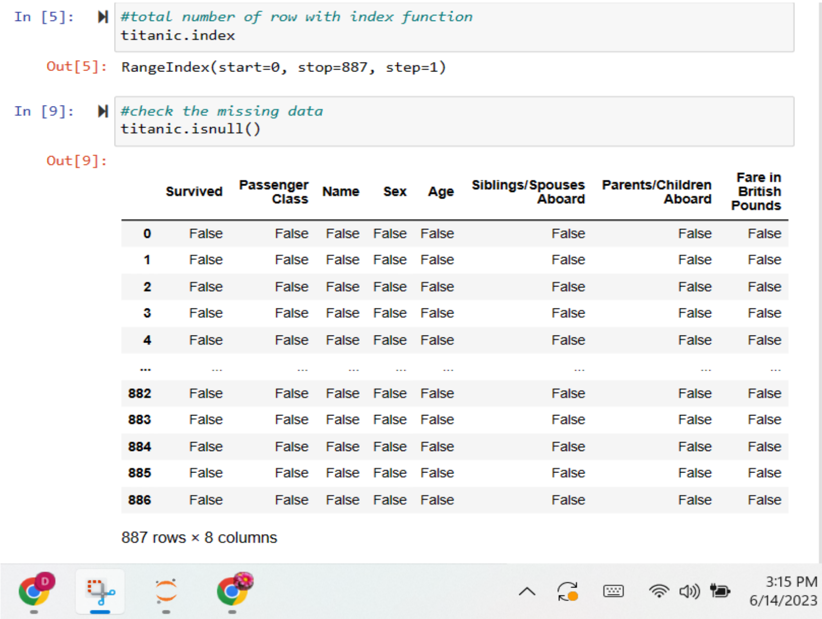
Figure 1: Import needs a library, loads the titanic.csv file, and explores the titanic dataset by getting information by info() function and seeing the dataset with the .head() process.



```
In [1]:  #import needs library for this project
         import math
         import collections

         import numpy as np
         import pandas as pd
         import matplotlib.pyplot as pp

In [2]:  #when we look at the large file,
         #we should make the pandas display setting more compact
         pd.options.display.max_rows = 16

In [3]:  #load the titanic file
         titanic = pd.read_csv("C:/Users/didem/OneDrive/Documents/CSUG Master DA/MIS5

In [6]:  #let's check titanic file information
         titanic.info()

         <class 'pandas.core.frame.DataFrame'>
         RangeIndex: 887 entries, 0 to 886
         Data columns (total 8 columns):
          #   Column                   Non-Null Count  Dtype  
         ---  ------                   --------------  -----  
          0   Survived                 887 non-null    int64  
          1   Passenger Class          887 non-null    int64  
          2   Name                     887 non-null    object 
          3   Sex                      887 non-null    object 
          4   Age                      887 non-null    float64
          5   Siblings/Spouses Aboard  887 non-null    int64  
          6   Parents/Children Aboard  887 non-null    int64  
          7   Fare in British Pounds   887 non-null    float64
         dtypes: float64(2), int64(4), object(2)
         memory usage: 55.6+ KB

In [8]:  # head function will display first four row
         titanic.head()

Out[8]:
```

Survived | Passenger Class | Name | Sex | Age | Siblings/Spouses Aboard | Parents/Children Aboard | Fare in British

Figure 2: Check index and missing data with the isnull() function.

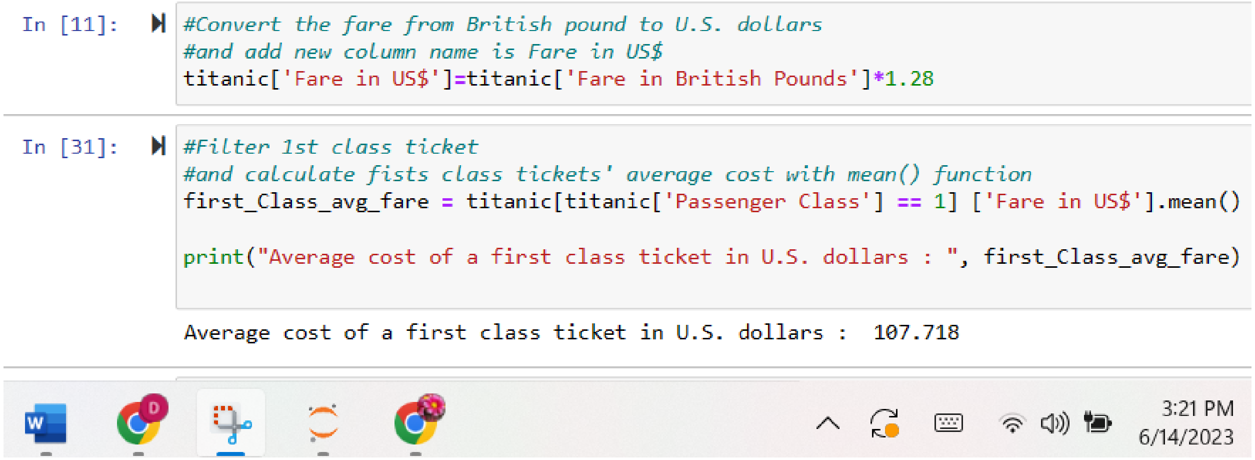


```
In [5]:   #total number of row with index function
          titanic.index

Out[5]:   RangeIndex(start=0, stop=887, step=1)

In [9]:   #check the missing data
          titanic.isnull()

Out[9]:
```

	Survived	Passenger Class	Name	Sex	Age	Siblings/Spouses Aboard	Parents/Children Aboard	Fare in British Pounds
0	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...
882	False	False	False	False	False	False	False	False
883	False	False	False	False	False	False	False	False
884	False	False	False	False	False	False	False	False
885	False	False	False	False	False	False	False	False
886	False	False	False	False	False	False	False	False

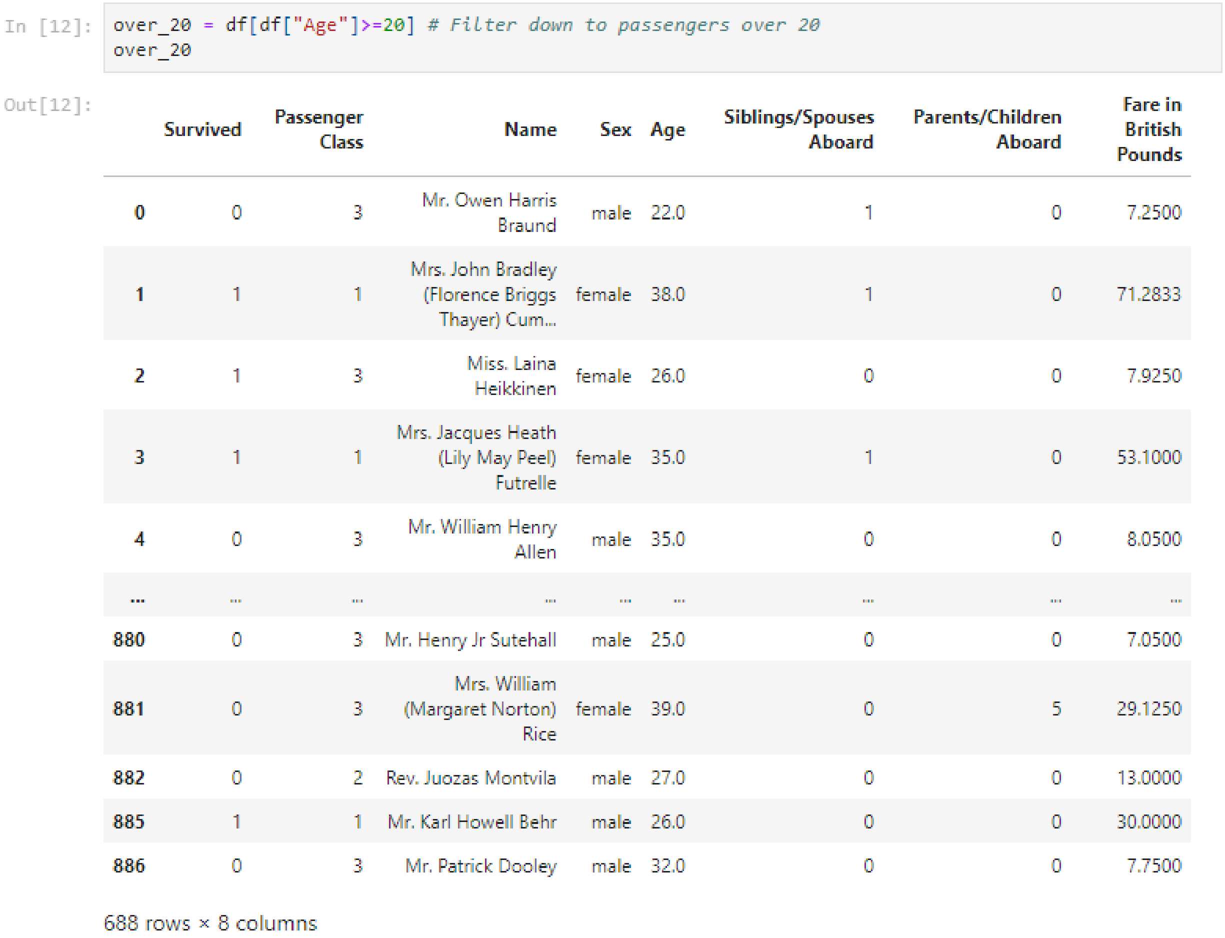
887 rows × 8 columns

Figure 3: Calculate the average cost of a first-class ticket in U.S. dollars.



```
In [11]:   #Convert the fare from British pound to U.S. dollars
           #and add new column name is Fare in US$
           titanic['Fare in US$']=titanic['Fare in British Pounds']*1.28

In [31]:   #Filter 1st class ticket
           #and calculate fists class tickets' average cost with mean() function
           first_Class_avg_fare = titanic[titanic['Passenger Class'] == 1] ['Fare in US$'].mean()

           print("Average cost of a first class ticket in U.S. dollars : ", first_Class_avg_fare)

           Average cost of a first class ticket in U.S. dollars :  107.718
```

Figure 4: Calculate the total number of passengers over 20 with siblings onboard using the if else statement to differentiate sibling and spouse by lambda function, take two arguments, and return a string insert the two parameters first and last. Then, filter it over 20 with siblings, and the shape function returns a tuple with each index having the number of checking elements.

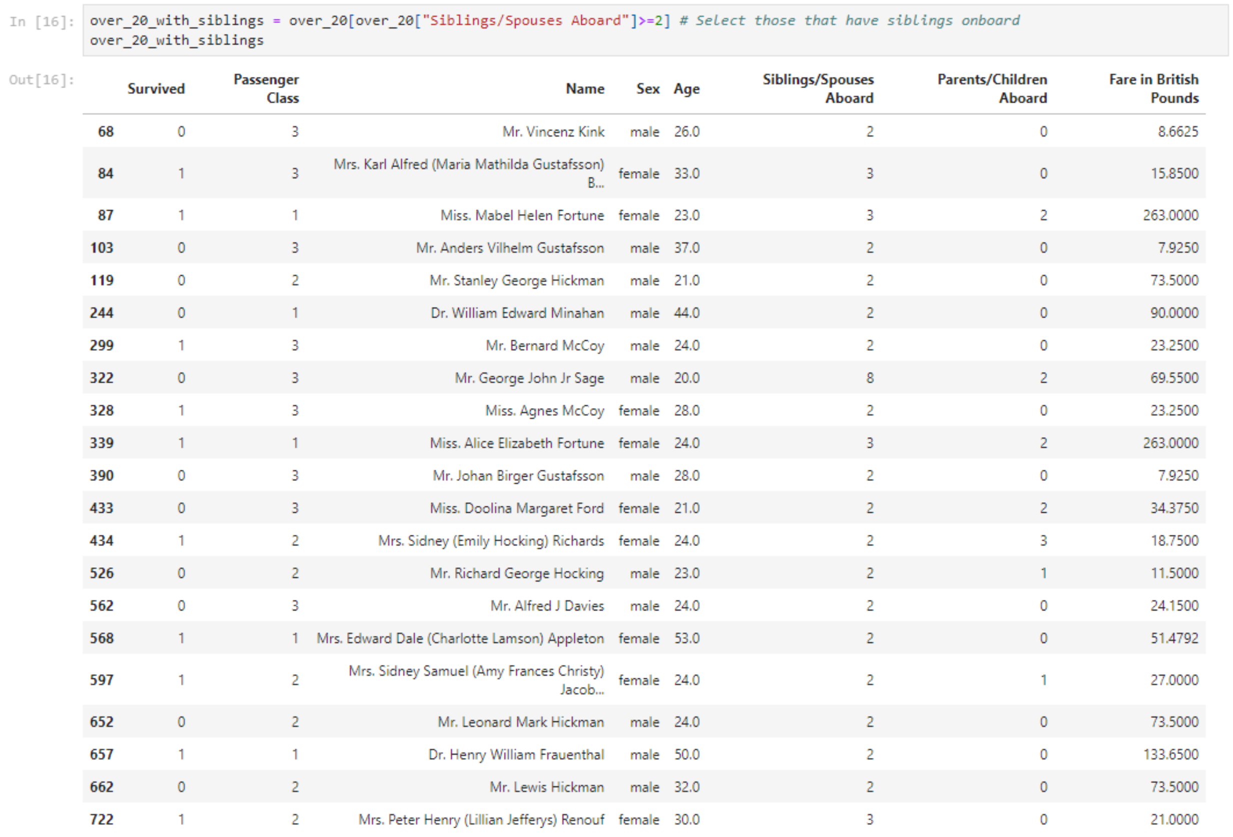


```
In [12]: over_20 = df[df["Age"]>=20] # Filter down to passengers over 20
         over_20
```

Out[12]:

	Survived	Passenger Class	Name	Sex	Age	Siblings/Spouses Aboard	Parents/Children Aboard	Fare in British Pounds
0	0	3	Mr. Owen Harris Braund	male	22.0	1	0	7.2500
1	1	1	Mrs. John Bradley (Florence Briggs Thayer) Cum...	female	38.0	1	0	71.2833
2	1	3	Miss. Laina Heikkinen	female	26.0	0	0	7.9250
3	1	1	Mrs. Jacques Heath (Lily May Peel) Futrelle	female	35.0	1	0	53.1000
4	0	3	Mr. William Henry Allen	male	35.0	0	0	8.0500
...	...	...	...	...	...	...	...	...
880	0	3	Mr. Henry Jr Sutehall	male	25.0	0	0	7.0500
881	0	3	Mrs. William (Margaret Norton) Rice	female	39.0	0	5	29.1250
882	0	2	Rev. Juozas Montvila	male	27.0	0	0	13.0000
885	1	1	Mr. Karl Howell Behr	male	26.0	0	0	30.0000
886	0	3	Mr. Patrick Dooley	male	32.0	0	0	7.7500

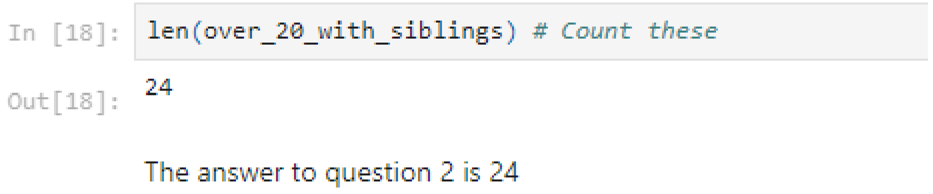
688 rows × 8 columns



```
In [16]: over_20_with_siblings = over_20[over_20["Siblings/Spouses Aboard"]>=2] # Select those that have siblings onboard
         over_20_with_siblings
```

Out[16]:

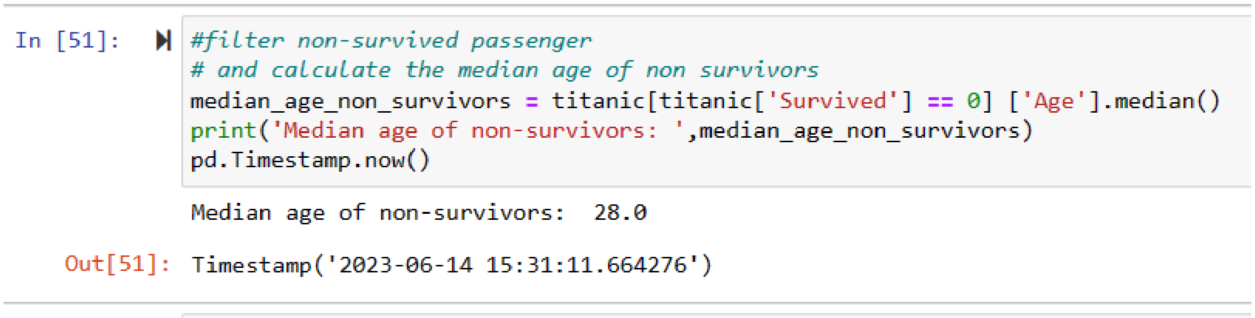
	Survived	Passenger Class	Name	Sex	Age	Siblings/Spouses Aboard	Parents/Children Aboard	Fare in British Pounds
68	0	3	Mr. Vincenz Kink	male	26.0	2	0	8.6625
84	1	3	Mrs. Karl Alfred (Maria Mathilda Gustafsson) B...	female	33.0	3	0	15.8500
87	1	1	Miss. Mabel Helen Fortune	female	23.0	3	2	263.0000
103	0	3	Mr. Anders Vilhelm Gustafsson	male	37.0	2	0	7.9250
119	0	2	Mr. Stanley George Hickman	male	21.0	2	0	73.5000
244	0	1	Dr. William Edward Minahan	male	44.0	2	0	90.0000
299	1	3	Mr. Bernard McCoy	male	24.0	2	0	23.2500
322	0	3	Mr. George John Jr Sage	male	20.0	8	2	69.5500
328	1	3	Miss. Agnes McCoy	female	28.0	2	0	23.2500
339	1	1	Miss. Alice Elizabeth Fortune	female	24.0	3	2	263.0000
390	0	3	Mr. Johan Birger Gustafsson	male	28.0	2	0	7.9250
433	0	3	Miss. Doolina Margaret Ford	female	21.0	2	2	34.3750
434	1	2	Mrs. Sidney (Emily Hocking) Richards	female	24.0	2	3	18.7500
526	0	2	Mr. Richard George Hocking	male	23.0	2	1	11.5000
562	0	3	Mr. Alfred J Davies	male	24.0	2	0	24.1500
568	1	1	Mrs. Edward Dale (Charlotte Lamson) Appleton	female	53.0	2	0	51.4792
597	1	2	Mrs. Sidney Samuel (Amy Frances Christy) Jacob...	female	24.0	2	1	27.0000
652	0	2	Mr. Leonard Mark Hickman	male	24.0	2	0	73.5000
657	1	1	Dr. Henry William Frauenthal	male	50.0	2	0	133.6500
662	0	2	Mr. Lewis Hickman	male	32.0	2	0	73.5000
722	1	2	Mrs. Peter Henry (Lillian Jefferys) Renouf	female	30.0	3	0	21.0000



```
In [18]: len(over_20_with_siblings) # Count these

Out[18]: 24
```

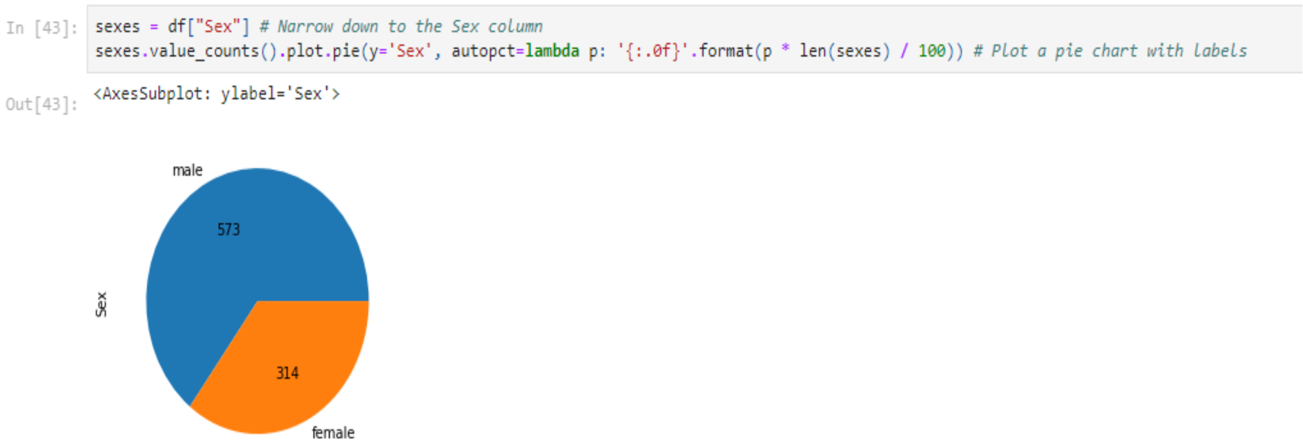
The answer to question 2 is 24

Figure 5: Calculate the median age with the median() function by specific selection of non-survivors by filter function ‘==.’



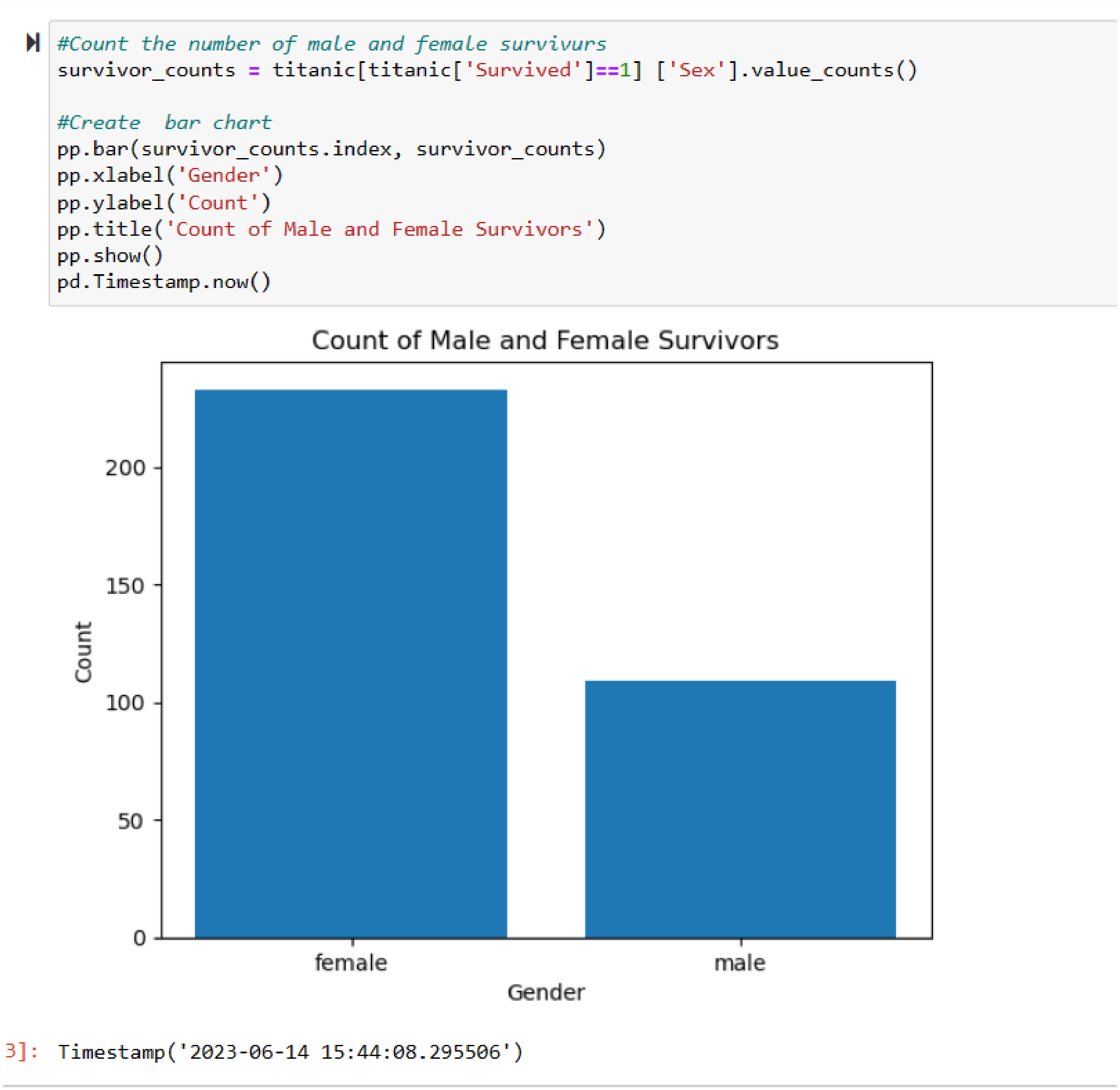
```
In [51]:  #filter non-survived passenger
          # and calculate the median age of non survivors
          median_age_non_survivors = titanic[titanic['Survived'] == 0] ['Age'].median()
          print('Median age of non-survivors: ',median_age_non_survivors)
          pd.Timestamp.now()

          Median age of non-survivors:  28.0

Out[51]:  Timestamp('2023-06-14 15:31:11.664276')
```

Figure 6: Calculate the total number of males and females with value_counts() function returns a group count of unique values then display on the pie chart by matplotlib pie() part containing gender_counts exceptional value, label it by index and size it by autopsy.



```
In [43]: sexes = df["Sex"] # Narrow down to the Sex column
         sexes.value_counts().plot.pie(y='Sex', autopct=lambda p: '{:.0f}'.format(p * len(sexes) / 100)) # Plot a pie chart with labels

Out[43]: <AxesSubplot: ylabel='Sex'>
```

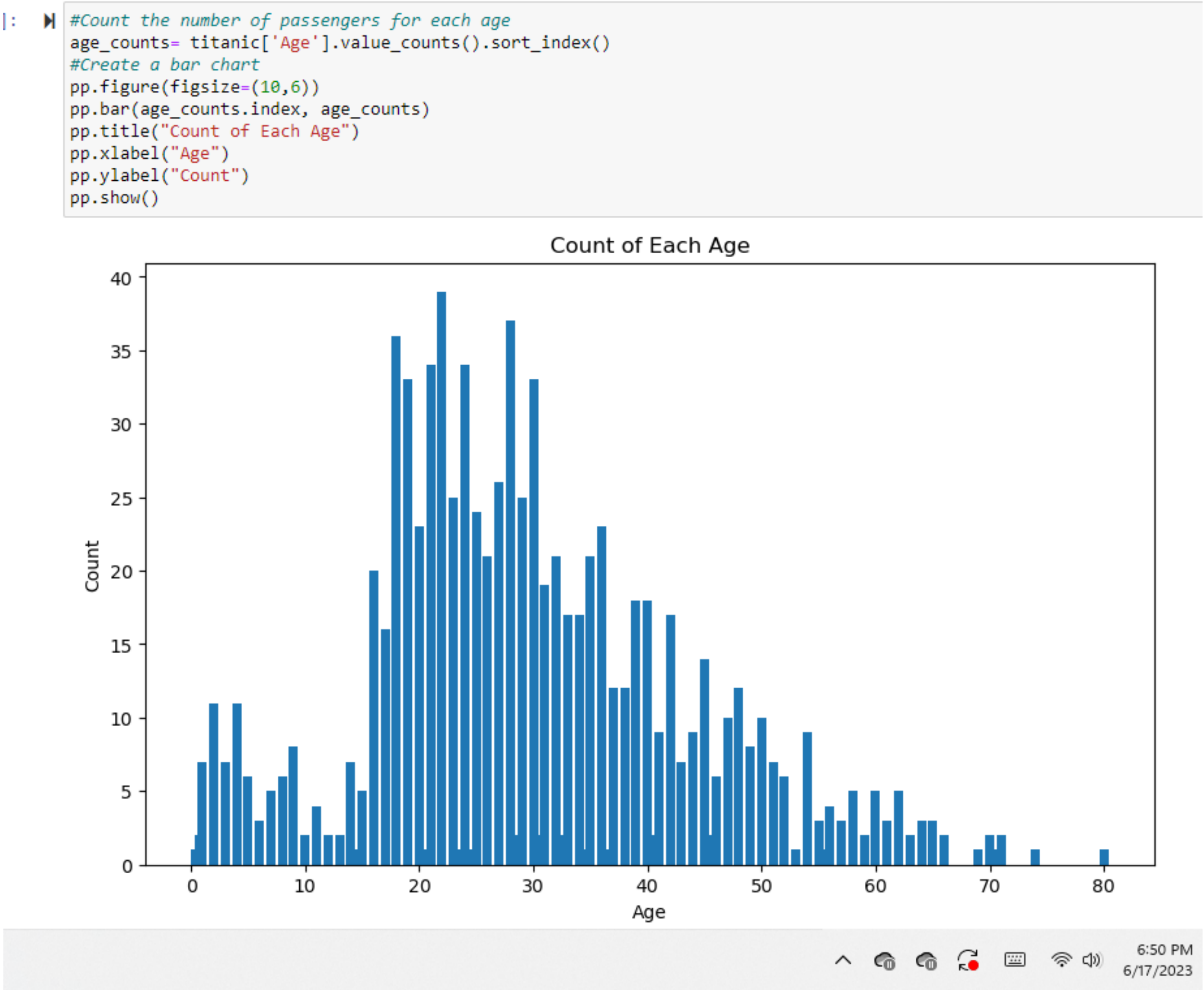
Figure 7: Create a bar chart to show the total number of male and female survivors with value_counts() count unique values, then show the bar() function containing the field name and index label and title it.



```
#Count the number of male and female survivurs
survivor_counts = titanic[titanic['Survived']==1] ['Sex'].value_counts()

#Create  bar chart
pp.bar(survivor_counts.index, survivor_counts)
pp.xlabel('Gender')
pp.ylabel('Count')
pp.title('Count of Male and Female Survivors')
pp.show()
pd.Timestamp.now()
```

```
3]:  Timestamp('2023-06-14 15:44:08.295506')
```

Figure 8: Create a histogram to show the total number of each age with hist() function containing field name age_counts include value_counts() process for calculating the unique value in series by 100 bins and x and y axis label and title it.



```
#Count the number of passengers for each age
age_counts= titanic['Age'].value_counts().sort_index()
#Create a bar chart
pp.figure(figsize=(10,6))
pp.bar(age_counts.index, age_counts)
pp.title("Count of Each Age")
pp.xlabel("Age")
pp.ylabel("Count")
pp.show()
```

Conclusion

The survivors’ passenger bar chart shows female survivors higher than males; the number of male passengers is more elevated than female passengers. Non-survivor passenger age median of 28 also surprises me because most customers are babies and kids—the average cost of the first class is $107.7, which is a valuable price. Most passengers are aged between 20 to 30, and the number of passengers over 20 with siblings on board was 23.

References

McKinney, W. (n.d.). *Python for Data Analysis Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media. ISBN- 1491957638.

Pandas.pydata.org,(2023). *Pandas.Series.value_counts*. https://pandas.pydata.org/docs/reference/api/pandas.Series.value_counts.

Burgaud, A.,(n.d.). *How to Use Python Lambda Functions*. https://realpython.com/python-lambda/

Mohanty, A. (2020). Step By Step Exploratory Data Analysis Of Titanic DataSet. https://medium.datadriveninvestor.com/step-by-step-exploratory-data-analysis-of-titanic-dataset-2d0fb09b0e86

Lindemann, A.,& Stolz, J.,(2021). *Teaching Mixed Methods: Using the Titanic Dataset to Teach Mixed Methods Data Analysis. Institute of Social Sciences of Religions, University of Lausanne, Switzerland.* 17(3),231-249, https://doi.org/10.5964/meth.4241