# Homework 07

## ⚠ Before you start ⚠

*Duplicate this Jupyter Notebook in your `week-08` folder (right-click -> Duplicate) and then add your last name to the beginning of it (ie. `blevins-hw-07.ipynb` - otherwise you risk having all your work overwritten when you try to sync your GitHub repository with your instructor's repository.*

---

Name: Rayce Loveland

We're going to be practing using the Pandas library to explore another dataset: a famouse collection of information about some passengers on board the *Titanic*. To find out more information about this dataset look at the data dictionary on this page: https://www.kaggle.com/c/titanic/data#:~:text=should%20look%20like.-,data%20dictionary,-Variable

**Import the pandas library.**

In [7]:

```
#Your Code Here
import pandas as pd
```

**Read in the CSV file.**

In [9]:

```
#Your Code Here
titanic_df = pd.read_csv('titanic.csv')
```

**Display the first 12 rows of your dataset.**

In [11]:

```
#Your Code Here
titanic_df.head(12)
```

Out[11]:

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Far |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 0 | 3 | Braund, Mr. Owen Harris | male | 22.0 | 1 | 0 | A/5 21171 | 7.250 |
| **1** | 2 | 1 | 1 | Cumings, Mrs. John Bradley (Florence Briggs Th... | female | 38.0 | 1 | 0 | PC 17599 | 71.283 |
| **2** | 3 | 1 | 3 | Heikkinen, Miss. Laina | female | 26.0 | 0 | 0 | STON/O2. 3101282 | 7.925 |
| **3** | 4 | 1 | 1 | Futrelle, Mrs. Jacques Heath (Lily May Peel) | female | 35.0 | 1 | 0 | 113803 | 53.100 |
| **4** | 5 | 0 | 3 | Allen, Mr. William Henry | male | 35.0 | 0 | 0 | 373450 | 8.050 |
| **5** | 6 | 0 | 3 | Moran, Mr. James | male | NaN | 0 | 0 | 330877 | 8.458 |
| **6** | 7 | 0 | 1 | McCarthy, Mr. Timothy J | male | 54.0 | 0 | 0 | 17463 | 51.862 |
| **7** | 8 | 0 | 3 | Palsson, Master. Gosta Leonard | male | 2.0 | 3 | 1 | 349909 | 21.075 |
| **8** | 9 | 1 | 3 | Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) | female | 27.0 | 0 | 2 | 347742 | 11.133 |
| **9** | 10 | 1 | 2 | Nasser, Mrs. Nicholas (Adele Achem) | female | 14.0 | 1 | 0 | 237736 | 30.070 |
| **10** | 11 | 1 | 3 | Sandstrom, Miss. Marguerite Rut | female | 4.0 | 1 | 1 | PP 9549 | 16.700 |

| | PassengerId | Survived | Pclass | Name | Sex | Age | SibSp | Parch | Ticket | Far |
|---|---|---|---|---|---|---|---|---|---|---|
| **11** | 12 | 1 | 1 | Bonnell, Miss. Elizabeth | female | 58.0 | 0 | 0 | 113783 | 26.550 |

**What are the different data types contained in each column?**

In [13]:

```python
#Your Code Here
titanic_df.columns
```

Out[13]:

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

**In your own words, what is the difference in the data types for `Survived` vs. `Age` columns?**

The difference between the two columns is that survived indicates whether the person survived the sinking or not. Age tells their age while on the Titanic.

**Use the `.isna()` or `.notna()` methods in conjunction with a filter to only select rows from your dataframe consisting of passengers for which we have information about the cabin they were in.**

In [16]:

```python
#Your Code Here
cabin_filter = titanic_df['Cabin'].notna()
titanic_cabin_df = titanic_df[cabin_filter]
print(titanic_cabin_df)
```

```
     PassengerId  Survived  Pclass  \
1              2         1       1   
3              4         1       1   
6              7         0       1   
10            11         1       3   
11            12         1       1   
..           ...       ...     ...   
871          872         1       1   
872          873         0       1   
879          880         1       1   
887          888         1       1   
889          890         1       1   

                                                  Name     Sex   Age  SibSp  \
1    Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1   
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1   
6                              McCarthy, Mr. Timothy J    male  54.0      0   
10                     Sandstrom, Miss. Marguerite Rut  female   4.0      1   
11                            Bonnell, Miss. Elizabeth  female  58.0      0   
..                                                 ...     ...   ...    ...   
871   Beckwith, Mrs. Richard Leonard (Sallie Monypeny)  female  47.0      1   
872                           Carlsson, Mr. Frans Olof    male  33.0      0   
879      Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)  female  56.0      0   
887                       Graham, Miss. Margaret Edith  female  19.0      0   
889                              Behr, Mr. Karl Howell    male  26.0      0   

     Parch    Ticket     Fare        Cabin Embarked  
1        0  PC 17599  71.2833          C85        C  
3        0    113803  53.1000         C123        S  
6        0     17463  51.8625          E46        S  
10       1   PP 9549  16.7000           G6        S  
11       0    113783  26.5500         C103        S  
..     ...       ...      ...          ...      ...  
871      1     11751  52.5542          D35        S  
872      0       695   5.0000  B51 B53 B55        S  
879      1     11767  83.1583          C50        C  
887      0    112053  30.0000          B42        S  
889      0    111369  30.0000         C148        C  

[204 rows x 12 columns]
```

**What percentage of rows (passengers) in the dataset have information about their cabin number?**

```
In [18]: #Your Code Here
         titanic_df['Cabin'].notna().value_counts(normalize=True)
```

```
Out[18]: Cabin
         False    0.771044
         True     0.228956
         Name: proportion, dtype: float64
```

23%

Some of our columns are hard to read. **Rename the following columns:**

- The `SibSp` column contains information about whether the passenger had family on board (siblings or spouses). **Rename the column `siblings_spouses` .**
- The `Pclass` column stands for the ticket class (1st, 2nd, or 3rd). **Rename the column `ticket_class` .**

*Hint: remember to change it permanently rather than temporarily*.

In [21]:
```
#Your Code Here
titanic_df.rename(columns={'SibSp': 'siblings_spouses'}, inplace=True)
titanic_df.rename(columns={'Pclass': 'ticket_class'}, inplace=True)
```

**Which passengers bought the nine most expensive tickets?**

In [23]:
```
#Your Code Here
titanic_df.sort_values('Fare', ascending=False)[['Name','Fare']].head(12)
```

Out[23]:

| | Name | Fare |
|---|---|---|
| **258** | Ward, Miss. Anna | 512.3292 |
| **737** | Lesurer, Mr. Gustave J | 512.3292 |
| **679** | Cardeza, Mr. Thomas Drake Martinez | 512.3292 |
| **88** | Fortune, Miss. Mabel Helen | 263.0000 |
| **27** | Fortune, Mr. Charles Alexander | 263.0000 |
| **341** | Fortune, Miss. Alice Elizabeth | 263.0000 |
| **438** | Fortune, Mr. Mark | 263.0000 |
| **311** | Ryerson, Miss. Emily Borie | 262.3750 |
| **742** | Ryerson, Miss. Susan Parker "Suzette" | 262.3750 |
| **118** | Baxter, Mr. Quigg Edmond | 247.5208 |
| **299** | Baxter, Mrs. James (Helene DeLaudeniere Chaput) | 247.5208 |
| **557** | Robbins, Mr. Victor | 227.5250 |

**What was the median age of passengers on the Titanic?**

In [25]:
```
#Your Code Here
titanic_df['Age'].median()
```

Out[25]:
```
28.0
```

**Who was the oldest passenger on the Titanic in our dataset?**

In [27]:
```
#Your Code Here
titanic_df.sort_values('Age', ascending=False)[['Name', 'Age']].head(1)
```

Out[27]:

|  | Name | Age |
|---|---|---|
| **630** | Barkworth, Mr. Algernon Henry Wilson | 80.0 |

**Use the groupby function to count how many passengers bought each class of ticket.**

In [29]:

```
#Your Code Here
titanic_df['ticket_class'].value_counts()
```

Out[29]:

```
ticket_class
3    491
1    216
2    184
Name: count, dtype: int64
```

**Use the groupby function to group passengers into different classes of ticket and then calculate the median age of passengers within each ticket class.**

In [31]:

```
#Your Code Here
titanic_df.groupby('ticket_class')['Age'].median()
```

Out[31]:

```
ticket_class
1    37.0
2    29.0
3    24.0
Name: Age, dtype: float64
```

**Use the groupby function to group passengers into different classes of ticket and then calculate the median ticket fare within each ticket class.**

In [34]:

```
#Your Code Here
titanic_df.groupby('ticket_class')['Fare'].median()
```

Out[34]:

```
ticket_class
1    60.2875
2    14.2500
3     8.0500
Name: Fare, dtype: float64
```

# Bonus Questions

**Bonus: Make the Survived column more legible. Write a function and apply it to the dataframe that changes the 0 and 1 values to "Died" and "Lived." Then display the first 10 rows to see if it worked.**

Note: when changing the values in columns, you might make mistakes. That's okay! You can always reload the dataframe from the original file to start over. When trying to answer this questions, each time you run it I'm going to have you start with the "original" dataframe so that you don't have to go back to the beginning of the notebook and run all the cells again.

In [37]:

```
titanic_df=pd.read_csv('titanic.csv')

# Your Code Here
```

**Bonus: What percentage of people survived the Titanic?**

In [39]:

```
#Your Code Here
```

**Bonus: Make a pie chart visualizing the proportion of people who survived the Titanic.** Hint: use the total number of rows in the dataframe to calculate the percentage.

In [41]:

```
#Your Code Here
```