Grammys Project

Are you excited to dive into data work for an exciting project at The Recording Academy? You know, the non-profit organization behind the Grammy Awards!

In this project, you'll work on real data from both websites owned by The Recording Academy, the non-profit organization behind the famous Grammy Awards. As you just learned, Ray Starck, the VP of Digital Strategy, decided to split the websites into grammy.com and recordingacademy.com to better serve the Recording Academy's various audience needs.

Now, you are tasked with examining the impact of splitting up the two websites, and analyzing the data for a better understanding of trends and audience behavior on both sites.

Are you ready?!?!

Let's do this!

Data Dictionary

To start, you will be working with two files, `grammys_live_web_analytics.csv` and `ra_live_web_analytics.csv`.

These files will contain the following information:

- **date** - The date the data was confirmed. It is in `yyyy-mm-dd` format.
- **visitors** - The number of users who went on the website on that day.
- **pageviews** - The number of pages that all users viewed on the website.
- **sessions** - The total number of sessions on the website. A session is a group of user interactions with your website that take place within a given time frame. For example a single session can contain multiple page views, events, social interactions.
- **bounced_sessions** - The total number of bounced sessions on the website. A bounced session is when a visitor comes to the website and does not interact with any pages / links and leaves.
- **avg_session_duration_secs** - The average length for all session durations for all users that came to the website that day.
- **awards_week** - A binary flag if the dates align with marketing campaigns before and after the Grammys award ceremony was held. This is the big marketing push to get as many eyeballs watching the event.
- **awards_night** - The actual night that Grammy Awards event was held.

Part I - Exploratory Data Analysis

Task 1

Import the `pandas`,`numpy`, and `plotly.express` libraries.

```
# Import libraries
import pandas as pd
import numpy as np
import plotly.express as px
```

```
# RUN THIS CELL - DO NOT MODIFY
# this formats numbers to two decimal places when shown in pandas
pd.set_option('display.float_format', lambda x: '%.2f' % x)
```

Task 2

Load in the first two files for your analysis. They are the `grammy_live_web_analytics.csv` and `ra_live_web_analytics.csv`.

A. For the `grammy_live_web_analytics.csv` file store that into a dataframe called `full_df`

B. For the `ra_live_web_analytics.csv` file store that into a dataframe called `rec_academy`

C. Preview the dataframes to familiarize yourself with the data.

All files needed can be found in the `datasets` folder.

```
# Read in dataframes
full_df = pd.read_csv('datasets/grammy_live_web_analytics.csv')
rec_academy = pd.read_csv('datasets/ra_live_web_analytics.csv')



# preview full_df dataframe
full_df.head()
```

	date	visitors	pageviews	sessions	bounced_sessions	avg_session_duration_secs	awards_week	awards_night
0	2017-01-01	9611	21407	10196	6490	86	0	0
1	2017-01-02	10752	25658	11350	7055	100	0	0
2	2017-01-03	11425	27062	12215	7569	92	0	0
3	2017-01-04	13098	29189	13852	8929	90	0	0
4	2017-01-05	12234	28288	12990	8105	95	0	0

```
# preview rec_academy dataframe
rec_academy.head()
```

	date	visitors	pageviews	sessions	bounced_sessions
0	2022-02-01	928	2856	1092	591
1	2022-02-02	1329	3233	1490	923
2	2022-02-03	1138	3340	1322	754
3	2022-02-04	811	2552	963	534
4	2022-02-05	541	1530	602	326

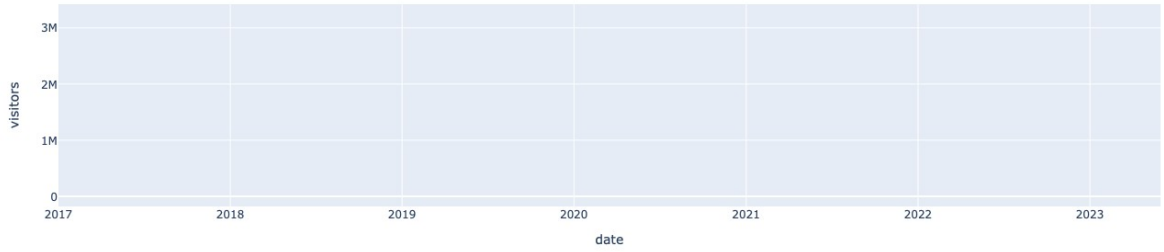
```
   avg_session_duration_secs  awards_week  awards_night
0                        148            0             0
1                         90            0             0
2                        127            0             0
3                        142            0             0
4                        111            0             0
```

Task 3

We all know The Grammy Awards is *the* biggest music event in the music industry, but how many visitors does that bring to the website?

A. Create a line chart of the number of users on the site for every day in the `full_df`. See if you can spot the days the Grammys awards are hosted.

```
# Plot a line chart of the visitors on the site.
line_chart = px.line(full_df, x='date', y='visitors', labels={'Date':
'Number of Users'})

line_chart.show()
```



Remark: The smaller spikes, typically around November/December of each year, are when the nominees are announced.

B. What can you say about the visitors to the website by looking at the graph?

It seems a majority of the viewers are typically going on the website for the Grammys, whether it be around the dates of the actual ceremony or it be for when the nomiees are ititially announced.

Task 4

Let's investigate what an "average" day looks like when the awards show is being hosted versus the other 364 days out of the year.

A. Use the pandas `.groupby()` to calculate the number of visitors on the site based on the values in the column `awards_night`.

```
avg_visitors = full_df.groupby('awards_night')['visitors'].mean()

print(avg_visitors)
```

```
awards_night
0     32388.28
1   1389590.23
Name: visitors, dtype: float64
```

B. What can you say about these results? Roughly how many more visitors are on the website for the awards ceremony versus a regular day?

(There are more than 1 million more visitors on the awards ceremony day compared to a regular day.)

Remark: This is The Recording Academy's biggest challenge! How do you transform a business that relies on the success of **one** event per year into one that continues to bring users back on the site year round?

Task 5

When The Recording Academy decided to split their website into two domains, grammy.com and recordingacademy.com, the data capture for grammy.com was not affected. So the `full_df` variable needs to be split separately into two dataframes. The day the domains were switched is on `2022-02-01`.

Create two new dataframes:

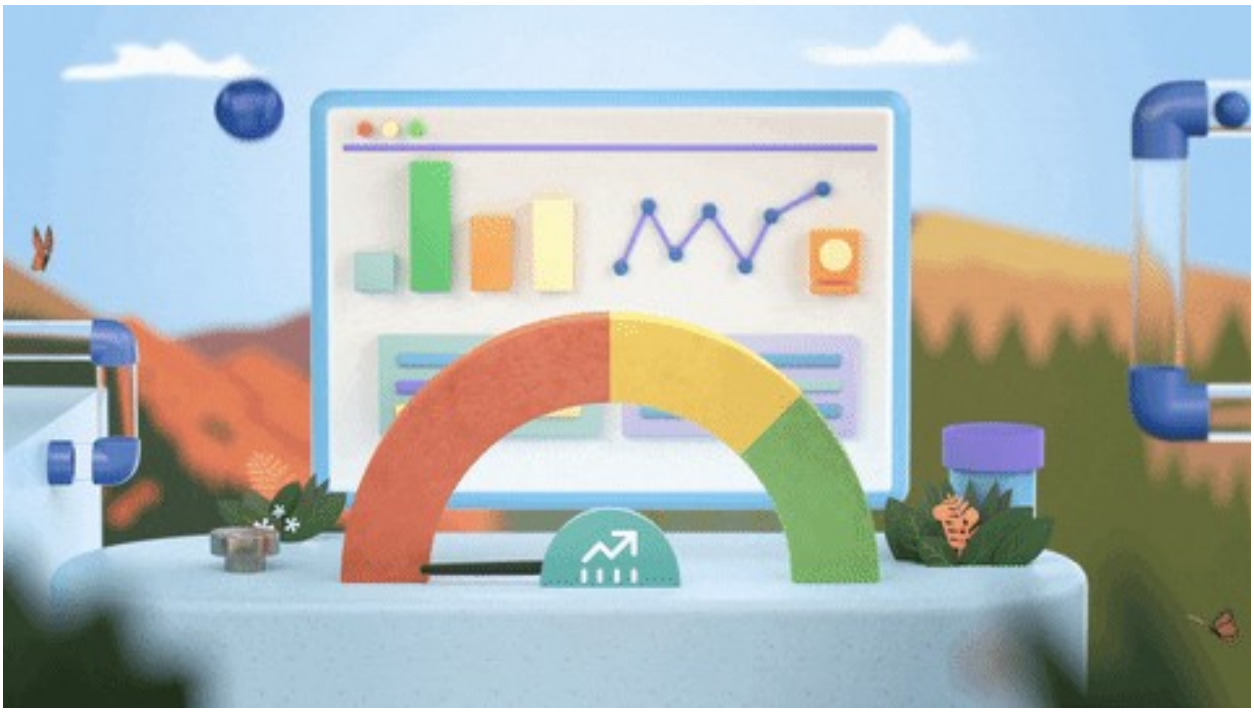
1. `combined_site` for all dates before `2022-02-01`
2. `grammys` for all dates after (and including) `2022-02-01`

```
# Split the data to separate the full_df into two new dataframes.
# One for before the switch of the websites and one for after

full_df['date'] = pd.to_datetime(full_df['date'])
combined_site = full_df[full_df['date'] < '2022-02-01']
grammys = full_df[full_df['date'] >= '2022-02-01']

# Run the following cell - DO NOT MODIFY
# .copy() prevents pandas from printing a scary-looking warning
message
combined_site = combined_site.copy()
grammys = grammys.copy()

# print the shape of the combined_site dataframe
combined_site.shape
```

```
(1857, 8)
```

If done correctly, the `combined_site` dataframe should have a total of `1857` rows and `8` columns

Part II - It's all about KPIs



There are certain key performance indicators (KPIs) of interest for The Recording Academy. Let's investigate those a little more.

Task 6

A. Create a new list called `frames` that has the `combined_site`, `rec_academy`, and `grammys` dataframes as entries. e.g. If the 3 dataframes were `df1`, `df2`, and `df3`, then the code would look like:

```
frames = [df1, df2, df3]

# create the list of dataframes
frames = [combined_site, rec_academy, grammys]
```

B. For each frame in the frames list, create a new column `pages_per_session`. This new column is the average number of pageviews per session on a given day. The higher this number the more "stickiness" your website has with your visitors.

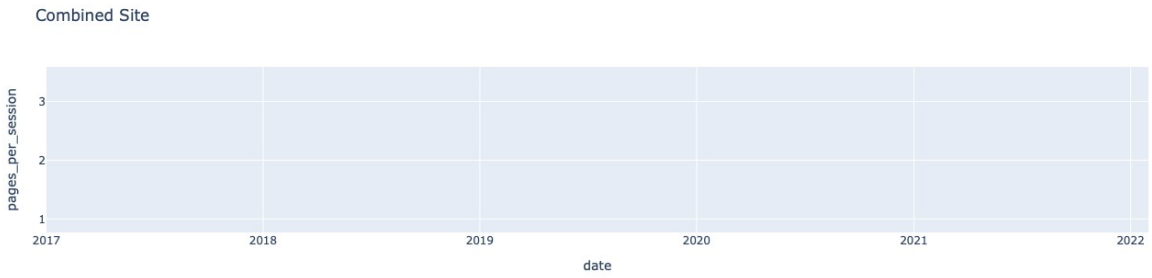
Hint: Divide the `pageviews` by `sessions`

This can be achieved by using the following template:

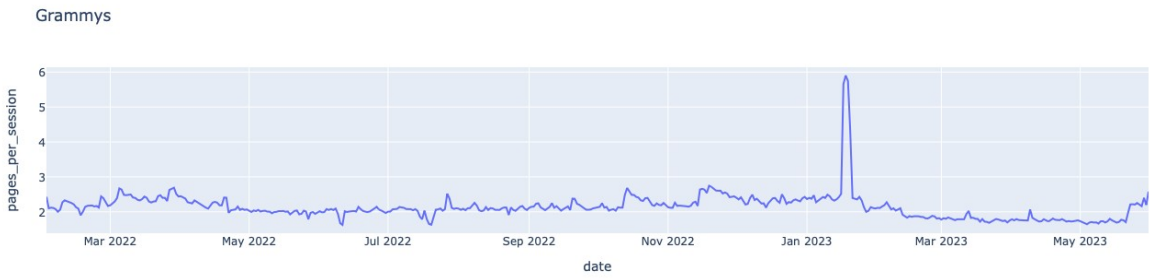
```
for frame in frames:
    frame['new_col'] = frame['col_A'] / frame['col_B']

# create the `pages_per_session` column for all 3 dataframes.
for frame in frames:
    frame['pages_per_session'] = frame['pageviews'] /
frame['sessions']
```

C. Visualize this new metric using a line chart for each site. (You will have 3 separate graphs)

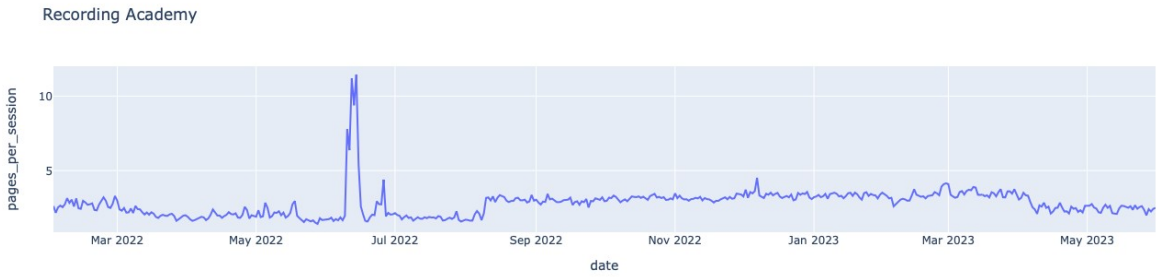
```
# combined_site graph
Combined_Site_Graph = px.line(combined_site, x='date',
y='pages_per_session', title='Combined Site')
Combined_Site_Graph.show()
```




```
# grammys graph
Grammys_Graph = px.line(grammys, x='date', y='pages_per_session',
title='Grammys')
Grammys_Graph.show()
```




```
# rec_academy graph
Rec_Academy_Graph = px.line(rec_academy, x='date',
y='pages_per_session', title='Recording Academy')
Rec_Academy_Graph.show()
```



D. Looking at the 3 charts above, what can you say about the `pages_per_session` when the websites were combined versus after they were split?

Note: Any large spikes in the data that do not correspond with the Grammy Awards Ceremony can be attributed to abnormalities in the data collection process and ignored in your analysis.

Separately, the data seems a lot more stable, but when combined, we can see where there are some dips and plateaus that aren't visible when looking at them one on one.

Task 7

Bounce rate is another important metric for The Recording Academy. Bounce Rate is a measure of the percentage of visitors who come to the site and *never interact with the website and leave*. In this task, you will define a function that takes in a dataframe as input and outputs the bounce rate.

A. Create a function called `bounce_rate` that:

1. Takes in a `dataframe` as input
2. adds up all of the values in the `bounced_sessions` column and stores in a variable called `sum_bounced`
3. adds up all of the values in the `sessions` column and stores it in a variable called `sum_sessions`
4. returns `100 * sum_bounced / sum_sessions`

Hint: You will need use the `.sum()` function both in the `sum_bounced` and `sum_sessions` calculations. Don't forget to multiply by `100` so that the answer appears as a percentage instead of a decimal.

```
def bounce_rate(dataframe):
    '''
    Calculates the bounce rate for visitors on the website.
    input: dataframe with bounced_sessions and sessions columns
    output: numeric value from bounce rate
    '''
    # WRITE YOUR CODE BELOW
    # Remember, the input for the function is called `dataframe`
    # So all calculations should reference that variable.
    sum_bounced = dataframe['bounced_sessions'].sum()
```

```
    sum_sessions = dataframe['sessions'].sum()
    bounce_rate_percentage = 100 * sum_bounced / sum_sessions

    return bounce_rate_percentage
```

B. Use the `frames` variable from Task 6 to loop over each website (represented by a dataframe) to calculate the bounce rate. Print the bounce rate for each site.

A template for getting the function to work will look like code below. Remember that this is NOT the print statement, you will still need to add that part.

Hint: To get the bounce rate use `bounce_rate(frame)`

```
for frame in frames:
    my_value = my_function(frame)
```

Tip: If you want to reduce the number of decimals shown in an f-string, you can add `:0.2f` just before the end of the curly brackets but after your variable. Example: `print(f'my value is: {my_value:0.2f}')`

```
# Calculate the Bounce Rate for each site. Use the frames list you
created in Task 6.
for frame in frames:
    bounce_rate_value = bounce_rate(frame)
    print(f'Bounce rate for {frames}: {bounce_rate_value:.2f}%')

Bounce rate for [           date  visitors  pageviews  sessions
bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
...         ...       ...        ...       ...               ...
1852 2022-01-27         2          2         2                 2
1853 2022-01-28     32986      79160     36571             20268
1854 2022-01-29     37899      79095     41920             25316
1855 2022-01-30     39931      81186     43743             26636
1856 2022-01-31     38221      92863     42291             21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
```

```
2                              92            0             0
2.22
3                              90            0             0
2.11
4                              95            0             0
2.18
...                           ...          ...           ...
...
1852                            0            0             0
1.00
1853                           83            0             0
2.16
1854                           63            0             0
1.89
1855                           61            0             0
1.86
1856                           67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928       2856      1092               591
1    2022-02-02      1329       3233      1490               923
2    2022-02-03      1138       3340      1322               754
3    2022-02-04       811       2552       963               534
4    2022-02-05       541       1530       602               326
..          ...       ...        ...       ...               ...
480  2023-05-27       845       2110      1058               702
481  2023-05-28       702       2100       872               537
482  2023-05-29      1027       2693      1197               777
483  2023-05-30      1320       4032      1658               992
484  2023-05-31      1618       5163      2072              1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
1                           90            0             0
2.17
2                          127            0             0
2.53
3                          142            0             0
2.65
4                          111            0             0
2.54
..                         ...          ...           ...
...
480                         96            0             0
1.99
```

```
481                          137             0              0
2.41
482                          125             0              0
2.25
483                          166             0              0
2.43
484                          172             0              0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
1858 2022-02-02     30511      43642     20761             11814
1859 2022-02-03     31502      44147     20830             12015
1860 2022-02-04     26863      39483     18700             10731
1861 2022-02-05     18014      35046     16860              9604
...         ...       ...        ...       ...               ...
2337 2023-05-27     14332      34178     15430              5424
2338 2023-05-28     13798      31708     14662              5509
2339 2023-05-29     20563      53396     22244              7005
2340 2023-05-30     16105      37950     17264              6452
2341 2023-05-31     31253      85686     33237              8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                          69            0             0
2.43
1858                          85            0             0
2.10
1859                          90            0             0
2.12
1860                          85            0             0
2.11
1861                          75            0             0
2.08
...                          ...          ...           ...
...
2337                          75            0             0
2.22
2338                          73            0             0
2.16
2339                          92            0             0
2.40
2340                          87            0             0
2.20
2341                         106            0             0
2.58

[485 rows x 9 columns]]: 41.58%
Bounce rate for [           date  visitors  pageviews  sessions
```

```
bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
...         ...       ...        ...       ...               ...
1852 2022-01-27         2          2         2                 2
1853 2022-01-28     32986      79160     36571             20268
1854 2022-01-29     37899      79095     41920             25316
1855 2022-01-30     39931      81186     43743             26636
1856 2022-01-31     38221      92863     42291             21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
2                            92            0             0
2.22
3                            90            0             0
2.11
4                            95            0             0
2.18
...                         ...          ...           ...
...
1852                          0            0             0
1.00
1853                         83            0             0
2.16
1854                         63            0             0
1.89
1855                         61            0             0
1.86
1856                         67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928       2856      1092               591
1    2022-02-02      1329       3233      1490               923
2    2022-02-03      1138       3340      1322               754
3    2022-02-04       811       2552       963               534
4    2022-02-05       541       1530       602               326
..          ...       ...        ...       ...               ...
480  2023-05-27       845       2110      1058               702
481  2023-05-28       702       2100       872               537
482  2023-05-29      1027       2693      1197               777
```

```
483  2023-05-30      1320       4032      1658              992
484  2023-05-31      1618       5163      2072             1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
1                           90            0             0
2.17
2                          127            0             0
2.53
3                          142            0             0
2.65
4                          111            0             0
2.54
..                         ...          ...           ...
...
480                         96            0             0
1.99
481                        137            0             0
2.41
482                        125            0             0
2.25
483                        166            0             0
2.43
484                        172            0             0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
1858 2022-02-02     30511      43642     20761             11814
1859 2022-02-03     31502      44147     20830             12015
1860 2022-02-04     26863      39483     18700             10731
1861 2022-02-05     18014      35046     16860              9604
...         ...       ...        ...       ...               ...
2337 2023-05-27     14332      34178     15430              5424
2338 2023-05-28     13798      31708     14662              5509
2339 2023-05-29     20563      53396     22244              7005
2340 2023-05-30     16105      37950     17264              6452
2341 2023-05-31     31253      85686     33237              8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                         69            0             0
2.43
1858                         85            0             0
2.10
1859                         90            0             0
2.12
```

```
1860                           85            0             0
2.11
1861                           75            0             0
2.08
...                           ...          ...           ...
...
2337                           75            0             0
2.22
2338                           73            0             0
2.16
2339                           92            0             0
2.40
2340                           87            0             0
2.20
2341                          106            0             0
2.58

[485 rows x 9 columns]]: 33.67%
Bounce rate for [           date  visitors  pageviews  sessions
bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
...         ...       ...        ...       ...               ...
1852 2022-01-27         2          2         2                 2
1853 2022-01-28     32986      79160     36571             20268
1854 2022-01-29     37899      79095     41920             25316
1855 2022-01-30     39931      81186     43743             26636
1856 2022-01-31     38221      92863     42291             21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
2                            92            0             0
2.22
3                            90            0             0
2.11
4                            95            0             0
2.18
...                         ...          ...           ...
...
1852                          0            0             0
1.00
1853                         83            0             0
```

```
2.16
1854                          63            0             0
1.89
1855                          61            0             0
1.86
1856                          67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928       2856      1092               591
1    2022-02-02      1329       3233      1490               923
2    2022-02-03      1138       3340      1322               754
3    2022-02-04       811       2552       963               534
4    2022-02-05       541       1530       602               326
..          ...       ...        ...       ...               ...
480  2023-05-27       845       2110      1058               702
481  2023-05-28       702       2100       872               537
482  2023-05-29      1027       2693      1197               777
483  2023-05-30      1320       4032      1658               992
484  2023-05-31      1618       5163      2072              1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
1                           90            0             0
2.17
2                          127            0             0
2.53
3                          142            0             0
2.65
4                          111            0             0
2.54
..                         ...          ...           ...
...
480                         96            0             0
1.99
481                        137            0             0
2.41
482                        125            0             0
2.25
483                        166            0             0
2.43
484                        172            0             0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
```

```
1858 2022-02-02     30511      43642     20761              11814
1859 2022-02-03     31502      44147     20830              12015
1860 2022-02-04     26863      39483     18700              10731
1861 2022-02-05     18014      35046     16860               9604
...         ...       ...        ...       ...                ...
2337 2023-05-27     14332      34178     15430               5424
2338 2023-05-28     13798      31708     14662               5509
2339 2023-05-29     20563      53396     22244               7005
2340 2023-05-30     16105      37950     17264               6452
2341 2023-05-31     31253      85686     33237               8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                         69            0             0
2.43
1858                         85            0             0
2.10
1859                         90            0             0
2.12
1860                         85            0             0
2.11
1861                         75            0             0
2.08
...                         ...          ...           ...
...
2337                         75            0             0
2.22
2338                         73            0             0
2.16
2339                         92            0             0
2.40
2340                         87            0             0
2.20
2341                        106            0             0
2.58

[485 rows x 9 columns]]: 40.16%
```

If done correctly, the `combined_site` and `grammys` site will each have bounce rates in the low 40s. The `rec_academy` will have a bounce rate in the low 30s

C. Another useful metric is how long on average visitors are staying on the website.

Calculate the `mean` of the `avg_session_duration_secs` for each of the sites. Print each one using an f-string.

```
# Calculate the average of the avg_session_duration_secs. Use the
frames list you created in Task 6.

for i, frame in enumerate(frames):
```

```
    mean_session_duration = frame['avg_session_duration_secs'].mean()
    print(f'Mean session duration for {frames}:
{mean_session_duration:.2f} seconds')


Mean session duration for [           date  visitors  pageviews
sessions  bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
...         ...       ...        ...       ...               ...
1852 2022-01-27         2          2         2                 2
1853 2022-01-28     32986      79160     36571             20268
1854 2022-01-29     37899      79095     41920             25316
1855 2022-01-30     39931      81186     43743             26636
1856 2022-01-31     38221      92863     42291             21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
2                            92            0             0
2.22
3                            90            0             0
2.11
4                            95            0             0
2.18
...                         ...          ...           ...
...
1852                          0            0             0
1.00
1853                         83            0             0
2.16
1854                         63            0             0
1.89
1855                         61            0             0
1.86
1856                         67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928       2856      1092               591
1    2022-02-02      1329       3233      1490               923
2    2022-02-03      1138       3340      1322               754
3    2022-02-04       811       2552       963               534
```

```
4    2022-02-05       541       1530       602              326
..          ...       ...        ...       ...              ...
480  2023-05-27       845       2110      1058              702
481  2023-05-28       702       2100       872              537
482  2023-05-29      1027       2693      1197              777
483  2023-05-30      1320       4032      1658              992
484  2023-05-31      1618       5163      2072             1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
1                           90            0             0
2.17
2                          127            0             0
2.53
3                          142            0             0
2.65
4                          111            0             0
2.54
..                         ...          ...           ...
...
480                         96            0             0
1.99
481                        137            0             0
2.41
482                        125            0             0
2.25
483                        166            0             0
2.43
484                        172            0             0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
1858 2022-02-02     30511      43642     20761             11814
1859 2022-02-03     31502      44147     20830             12015
1860 2022-02-04     26863      39483     18700             10731
1861 2022-02-05     18014      35046     16860              9604
...         ...       ...        ...       ...               ...
2337 2023-05-27     14332      34178     15430              5424
2338 2023-05-28     13798      31708     14662              5509
2339 2023-05-29     20563      53396     22244              7005
2340 2023-05-30     16105      37950     17264              6452
2341 2023-05-31     31253      85686     33237              8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                         69            0             0
```

```
2.43
1858                          85            0             0
2.10
1859                          90            0             0
2.12
1860                          85            0             0
2.11
1861                          75            0             0
2.08
...                          ...          ...           ...
...
2337                          75            0             0
2.22
2338                          73            0             0
2.16
2339                          92            0             0
2.40
2340                          87            0             0
2.20
2341                         106            0             0
2.58

[485 rows x 9 columns]]: 102.85 seconds
Mean session duration for [           date  visitors  pageviews
sessions  bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
...         ...       ...        ...       ...               ...
1852 2022-01-27         2          2         2                 2
1853 2022-01-28     32986      79160     36571             20268
1854 2022-01-29     37899      79095     41920             25316
1855 2022-01-30     39931      81186     43743             26636
1856 2022-01-31     38221      92863     42291             21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
2                            92            0             0
2.22
3                            90            0             0
2.11
4                            95            0             0
2.18
```

```
...                          ...          ...           ...
...
1852                           0            0             0
1.00
1853                          83            0             0
2.16
1854                          63            0             0
1.89
1855                          61            0             0
1.86
1856                          67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928        2856      1092               591
1    2022-02-02      1329        3233      1490               923
2    2022-02-03      1138        3340      1322               754
3    2022-02-04       811        2552       963               534
4    2022-02-05       541        1530       602               326
..          ...       ...         ...       ...               ...
480  2023-05-27       845        2110      1058               702
481  2023-05-28       702        2100       872               537
482  2023-05-29      1027        2693      1197               777
483  2023-05-30      1320        4032      1658               992
484  2023-05-31      1618        5163      2072              1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
1                           90            0             0
2.17
2                          127            0             0
2.53
3                          142            0             0
2.65
4                          111            0             0
2.54
..                         ...          ...           ...
...
480                         96            0             0
1.99
481                        137            0             0
2.41
482                        125            0             0
2.25
483                        166            0             0
2.43
```

```
484                         172            0             0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
1858 2022-02-02     30511      43642     20761             11814
1859 2022-02-03     31502      44147     20830             12015
1860 2022-02-04     26863      39483     18700             10731
1861 2022-02-05     18014      35046     16860              9604
...         ...       ...        ...       ...               ...
2337 2023-05-27     14332      34178     15430              5424
2338 2023-05-28     13798      31708     14662              5509
2339 2023-05-29     20563      53396     22244              7005
2340 2023-05-30     16105      37950     17264              6452
2341 2023-05-31     31253      85686     33237              8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                         69            0             0
2.43
1858                         85            0             0
2.10
1859                         90            0             0
2.12
1860                         85            0             0
2.11
1861                         75            0             0
2.08
...                         ...          ...           ...
...
2337                         75            0             0
2.22
2338                         73            0             0
2.16
2339                         92            0             0
2.40
2340                         87            0             0
2.20
2341                        106            0             0
2.58

[485 rows x 9 columns]]: 128.50 seconds
Mean session duration for [           date  visitors  pageviews
sessions  bounced_sessions  \
0    2017-01-01      9611      21407     10196              6490
1    2017-01-02     10752      25658     11350              7055
2    2017-01-03     11425      27062     12215              7569
3    2017-01-04     13098      29189     13852              8929
4    2017-01-05     12234      28288     12990              8105
```

```
...         ...       ...         ...       ...                ...
1852 2022-01-27         2           2         2                  2
1853 2022-01-28     32986       79160     36571              20268
1854 2022-01-29     37899       79095     41920              25316
1855 2022-01-30     39931       81186     43743              26636
1856 2022-01-31     38221       92863     42291              21747

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                            86            0             0
2.10
1                           100            0             0
2.26
2                            92            0             0
2.22
3                            90            0             0
2.11
4                            95            0             0
2.18
...                         ...          ...           ...
...
1852                          0            0             0
1.00
1853                         83            0             0
2.16
1854                         63            0             0
1.89
1855                         61            0             0
1.86
1856                         67            0             0
2.20

[1857 rows x 9 columns],            date  visitors  pageviews
sessions  bounced_sessions  \
0    2022-02-01       928       2856      1092               591
1    2022-02-02      1329       3233      1490               923
2    2022-02-03      1138       3340      1322               754
3    2022-02-04       811       2552       963               534
4    2022-02-05       541       1530       602               326
..          ...       ...        ...       ...               ...
480  2023-05-27       845       2110      1058               702
481  2023-05-28       702       2100       872               537
482  2023-05-29      1027       2693      1197               777
483  2023-05-30      1320       4032      1658               992
484  2023-05-31      1618       5163      2072              1195

     avg_session_duration_secs  awards_week  awards_night
pages_per_session
0                          148            0             0
2.62
```

```
1                            90            0             0
2.17
2                           127            0             0
2.53
3                           142            0             0
2.65
4                           111            0             0
2.54
..                          ...          ...           ...
...
480                          96            0             0
1.99
481                         137            0             0
2.41
482                         125            0             0
2.25
483                         166            0             0
2.43
484                         172            0             0
2.49

[485 rows x 9 columns],            date  visitors  pageviews  sessions
bounced_sessions  \
1857 2022-02-01     33209      74033     30472             13070
1858 2022-02-02     30511      43642     20761             11814
1859 2022-02-03     31502      44147     20830             12015
1860 2022-02-04     26863      39483     18700             10731
1861 2022-02-05     18014      35046     16860              9604
...         ...       ...        ...       ...               ...
2337 2023-05-27     14332      34178     15430              5424
2338 2023-05-28     13798      31708     14662              5509
2339 2023-05-29     20563      53396     22244              7005
2340 2023-05-30     16105      37950     17264              6452
2341 2023-05-31     31253      85686     33237              8200

      avg_session_duration_secs  awards_week  awards_night
pages_per_session
1857                         69            0             0
2.43
1858                         85            0             0
2.10
1859                         90            0             0
2.12
1860                         85            0             0
2.11
1861                         75            0             0
2.08
...                         ...          ...           ...
...
```

```
2337                          75             0             0
2.22
2338                          73             0             0
2.16
2339                          92             0             0
2.40
2340                          87             0             0
2.20
2341                         106             0             0
2.58

[485 rows x 9 columns]]: 82.99 seconds
```

D. What can you say about these two metrics as it relates to each of the websites?

Users seem to be staying a significant number of seconds longer on two of the websites versus the other one.

Part III - Demographics

Age demographics are a way to see which audience(s) your content is resonating with the most. This can inform marketing campaigns, ads, and much more.

Let's investigate the demographics for the two websites. This will require reading in two new files and joining them in python.

Task 8

The `grammys_age_demographics.csv` and `tra_age_demographics.csv` each contain the following information:

- **age_group** - The age group range. e.g. `18-24` are all visitors between the ages of 18 to 24 who come to the site.
- **pct_visitors** - The percentage of all of the websites visitors that come from that specific age group.

A. Read in the `grammys_age_demographics.csv` and `tra_age_demograhics.csv` files and store them into dataframes named `age_grammys` and `age_tra`, respectively.

```
# read in the files
age_grammys = pd.read_csv('datasets/grammys_age_demographics.csv')
age_tra = pd.read_csv('datasets/tra_age_demographics.csv')

# preview the age_grammys file. the age_tra will look very similar.
age_grammys.head()
```

```
  age_group  pct_visitors
0     18-24         27.37
1     25-34         24.13
2     35-44         18.72
3     45-54         13.57
4     55-64          9.82
```

B. For each dataframe, create a new column called `website` whose value is the name of the website. e.g. the `age_grammys` values for `website` should all be `Grammys` and for the `age_tra` they should be `Recording Academy`.

```
# create the website column
age_grammys['website'] = 'Grammys'
age_tra['website'] = 'Recording Academy'
```

C. use the `pd.concat()` method to join these two datasets together. Store the result into a new variable called `age_df`

Hint: Remember that you need to put your dataframe variables inside of a **list** first then pass that as your input of `pd.concat()`

```
# use pd.concat to join the two datasets
age_df = pd.concat([age_grammys, age_tra])
age_df.reset_index(drop=True, inplace=True)

age_df.shape
```

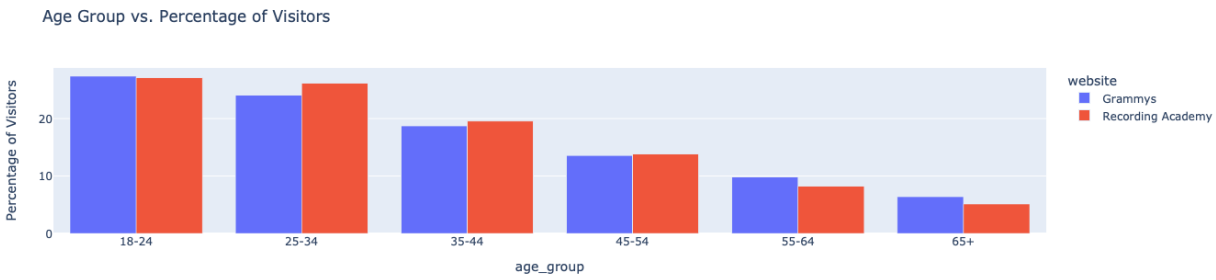
```
(12, 3)
```

If done correctly your new dataframe will have `12` rows and `3` columns.

D. Create a bar chart of the `age_group` and `pct_visitors`. This chart should have, for each age group, one color for the Recording Academy and a different color for the Grammys.

Hint: You will need to use the `barmode='group'` option in `px.bar()`. See the code snippet below to guide you.

```
# template for visualization
px.bar(dataframe, x='variable1', y='variable2', color='variable3',
barmode='group')

# Create bar chart
visitorages = px.bar(age_df, x='age_group', y='pct_visitors',
color='website', barmode='group',
             title='Age Group vs. Percentage of Visitors',
labels={'pct_visitors': 'Percentage of Visitors'})

visitorages.show()
```



E. Looking at the chart above, what can you say about how the age demographics differ between the two websites?

It seems the age demographics are pretty similar, with there being a small shift in whihc middle aged individuals tend to look at the recording academy website omre often thand the grammys website, but then a small shift occurs again once we hit 55-64 where they tend to view the grammys more than the recording academy.

Part IV - Recommendation

Task 9

Using the work you did in this project, would you recommend that the websites stay separate? Please give a 2-3 paragraph answer using details from the analysis work above explaining why or why not they should stay separate.

I believe that The Recording Academy and The Grammy's should keep their websites separate. There are plenty of indicators here that point towards that being the best option. To start, I think that having separate data to analyze for the two sectors is important. By keeping The Recording Academy and The Grammys separate, we are able to tell the specific interests of the website viewer. If we were to combine these two sites, there would be heavier traffick of course, but we would not be able to as easily pinpoint what the user in using the website for.

Looking at the amount of time users typically spend on the sites when they are browsing, we can see that The Recording Academy website users are staying on the site for a longer duration of seconds, probably looking for somethign specific. If the websites were combined, it might cause navigational issues for the users, resulting in them taking more time to find the content they are looking for. Since The Grammys website is much more specific, it makes sense to keep that away from the Recording Academy, so those users have ease when finding the information they want

instead of being overwhelmed with the other information that is provided on The Recording Academy's webpage.

LevelUp

Ray and Harvey are both interested to see how the Grammys.com website compares to that of their main music award competitor, The American Music Awards (AMA). The dashboard below is aggregated information about the performace of The AMA website for the months of April, May, and June of 2023.

Your task is to determine how the Grammys website is performing relative to The AMA website. In particular, you will be looking at the device distribution and total visits over the same time span and leveraging information about Visit Duration, Bounce Rate, and Pages / Visit from your work in the core of this project.

Let's review some of the content from above.

The **Total Visits** column is the total number of visitors on the website during the timespan given. The **Device Distribution** is the percentage share of visitors coming from Desktop users (PCs, Macs, etc.) and Mobile Users (iPhone, Android, etc.).

Visitors on the AMA website are spending on average, 5 mins and 53 seconds on the site and viewing 2.74 pages per visit (aka session). They have a bounce rate of 54.31%

A. Load in the two files. The `desktop_users.csv` and `mobile_users.csv` files contain the users coming from desktop users and mobile users respectively.

Store them in variables named `desktop_users` and `mobile_users`

```
# Load in the data
desktop_users = pd.read_csv('datasets/desktop_users.csv')
mobile_users = pd.read_csv('datasets/mobile_users.csv')

# preview the desktop_users file
desktop_users.head()
```

```
         date          segment  visitors
0  2022-02-01  Desktop Traffic     10195
1  2022-02-02  Desktop Traffic     10560
2  2022-02-03  Desktop Traffic      9935
3  2022-02-04  Desktop Traffic      8501
4  2022-02-05  Desktop Traffic      5424
```

```
# preview mobile_users file
mobile_users.head()
```

```
         date         segment  visitors
0  2022-02-01  Mobile Traffic     23494
1  2022-02-02  Mobile Traffic     20234
2  2022-02-03  Mobile Traffic     22816
3  2022-02-04  Mobile Traffic     18592
4  2022-02-05  Mobile Traffic     13298
```

As you can imagine, you will be joining the two datasets together! But before you do that, you will modify the column names so that it's easier to use.

B. For each dataframe, change the name of the `visitors` column so that it says which category they come from. For example, the `desktop_users` dataframe should have a column named `desktop_visitors` instead of `visitors`.

Additionally, drop the `segment` column since it is no longer needed.

```
# change name of the visitors column to indicate which category it
comes from

desktop_users = desktop_users.rename(columns={'visitors':
'desktop_visitors'})
mobile_users = mobile_users.rename(columns={'visitors':
'mobile_visitors'})

# drop the segment column from each dataframe

desktop_users = desktop_users.drop(columns=['segment'])
mobile_users = mobile_users.drop(columns=['segment'])
```

```
---------------------------------------------------------------------
-----
KeyError                                  Traceback (most recent call
```

```
last)
Cell In [57], line 3
      1 # drop the segment column from each dataframe
----> 3 desktop_users = desktop_users.drop(columns=['segment'])
      4 mobile_users = mobile_users.drop(columns=['segment'])

File
/opt/conda/lib/python3.10/site-packages/pandas/util/_decorators.py:311
, in
deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*arg
s, **kwargs)
    305 if len(args) > num_allow_args:
    306     warnings.warn(
    307         msg.format(arguments=arguments),
    308         FutureWarning,
    309         stacklevel=stacklevel,
    310     )
--> 311 return func(*args, **kwargs)

File
/opt/conda/lib/python3.10/site-packages/pandas/core/frame.py:4906, in
DataFrame.drop(self, labels, axis, index, columns, level, inplace,
errors)
   4774 @deprecate_nonkeyword_arguments(version=None,
allowed_args=["self", "labels"])
   4775 def drop(
   4776     self,
   (...)
   4783     errors: str = "raise",
   4784 ):
   4785     """
   4786     Drop specified labels from rows or columns.
   4787
   (...)
   4904             weight  1.0     0.8
   4905     """
-> 4906     return super().drop(
   4907         labels=labels,
   4908         axis=axis,
   4909         index=index,
   4910         columns=columns,
   4911         level=level,
   4912         inplace=inplace,
   4913         errors=errors,
   4914     )

File
/opt/conda/lib/python3.10/site-packages/pandas/core/generic.py:4150,
in NDFrame.drop(self, labels, axis, index, columns, level, inplace,
errors)
```

```
   4148 for axis, labels in axes.items():
   4149     if labels is not None:
-> 4150         obj = obj._drop_axis(labels, axis, level=level,
errors=errors)
   4152 if inplace:
   4153     self._update_inplace(obj)

File
/opt/conda/lib/python3.10/site-packages/pandas/core/generic.py:4185,
in NDFrame._drop_axis(self, labels, axis, level, errors)
   4183         new_axis = axis.drop(labels, level=level,
errors=errors)
   4184     else:
-> 4185         new_axis = axis.drop(labels, errors=errors)
   4186     result = self.reindex(**{axis_name: new_axis})
   4188 # Case for non-unique axis
   4189 else:

File
/opt/conda/lib/python3.10/site-packages/pandas/core/indexes/base.py:60
17, in Index.drop(self, labels, errors)
   6015 if mask.any():
   6016     if errors != "ignore":
-> 6017         raise KeyError(f"{labels[mask]} not found in axis")
   6018     indexer = indexer[~mask]
   6019 return self.delete(indexer)

KeyError: "['segment'] not found in axis"
```

C. Join the two dataframes together in a new variable called `segment_df`.

```
# join the two dataframes and preview the dataframe
segment_df = pd.concat([desktop_users, mobile_users], axis=1)

segment_df.head()
```

	date	desktop_visitors	date	mobile_visitors
0	2022-02-01	10195	2022-02-01	23494
1	2022-02-02	10560	2022-02-02	20234
2	2022-02-03	9935	2022-02-03	22816
3	2022-02-04	8501	2022-02-04	18592
4	2022-02-05	5424	2022-02-05	13298

D. In the next few steps, you will calculate the percentage share of users coming from desktop and mobile on the Grammys website.

Calculate a new column, `total_visitors` that is the addition of `desktop_visitors` and `mobile_visitors`.

```
# create total_visitors column
segment_df['total_visitors'] = segment_df['desktop_visitors'] +
segment_df['mobile_visitors']
```

To calculate the percentage share you will first need to filter the data to dates after (and including) `2023-04-01`. Then calculate the `sum` of desktop visitors and total visitors and divide those values. The percentage share of mobile visitors will be the value needed to get to 100%.

```
# filter and calculate the percentage share

# Check for duplicate values in the 'date' column
duplicate_dates = segment_df['date'].duplicated(keep=False)

# Print the duplicate dates if any
print(segment_df[duplicate_dates])
```

```
---------------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
Cell In [66], line 4
      1 # filter and calculate the percentage share
      2
      3 # Convert the date column to datetime using errors='coerce' to
handle any potential issues
----> 4 segment_df['date'] = pd.to_datetime(segment_df['date'],
errors='coerce')
      6 # Drop rows with missing dates
      7 segment_df = segment_df.dropna(subset=['date'])

File
/opt/conda/lib/python3.10/site-packages/pandas/core/tools/datetimes.py
:890, in to_datetime(arg, errors, dayfirst, yearfirst, utc, format,
exact, unit, infer_datetime_format, origin, cache)
    888         result = arg._constructor(values, index=arg.index,
name=arg.name)
    889 elif isinstance(arg, (ABCDataFrame, abc.MutableMapping)):
--> 890     result = _assemble_from_unit_mappings(arg, errors, tz)
    891 elif isinstance(arg, Index):
    892     cache_array = _maybe_cache(arg, format, cache,
convert_listlike)

File
/opt/conda/lib/python3.10/site-packages/pandas/core/tools/datetimes.py
:975, in _assemble_from_unit_mappings(arg, errors, tz)
    973 arg = DataFrame(arg)
```

```
    974 if not arg.columns.is_unique:
--> 975     raise ValueError("cannot assemble with duplicate keys")
    977 # replace passed unit with _unit_map
    978 def f(value):

ValueError: cannot assemble with duplicate keys
```

What is the percentage share of desktop and mobile visitors on the Grammys website in the timeframe in question?

What is the total number of visitors on the site during this timeframe?

(Double-click this cell to write your answer)

E. How is the Grammys website performing relative to its competitor? What is the Grammys doing well and what KPIs does it need to improve?

(Double-click this cell to write your answer)