## **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!



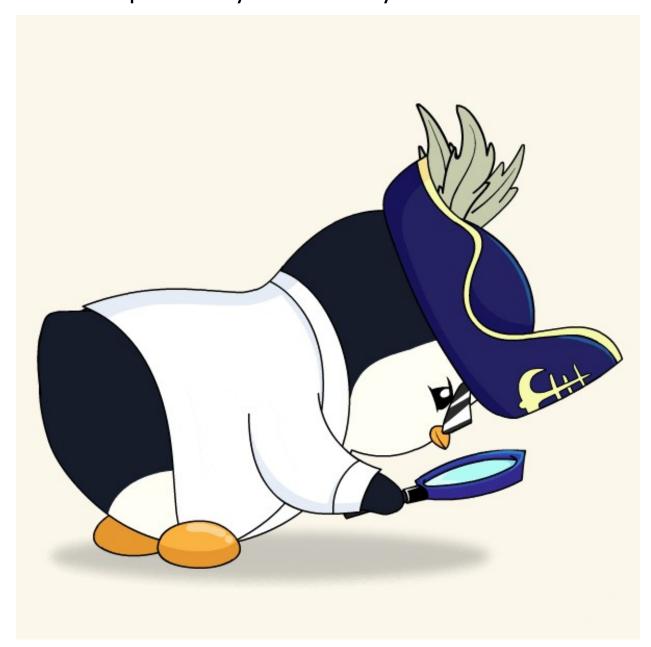
## **Data Dictionary**

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()
                                               bounced sessions \
         date
               visitors
                         pageviews
                                     sessions
  2017-01-01
                   9611
                              21407
                                        10196
                                                            6490
  2017-01-02
                              25658
                                        11350
                                                            7055
1
                  10752
2 2017-01-03
                  11425
                              27062
                                        12215
                                                            7569
  2017-01-04
                  13098
                              29189
                                        13852
                                                            8929
4 2017-01-05
                  12234
                              28288
                                        12990
                                                            8105
   avg session duration secs awards week
                                            awards night
0
                           86
                                         0
1
                          100
                                         0
                                                        0
2
                                         0
                                                        0
                           92
3
                           90
                                         0
                                                        0
4
                           95
# preview rec_academy dataframe
rec academy.head()
                                               bounced sessions \
         date
               visitors
                         pageviews
                                     sessions
  2022-02-01
                    928
                               2856
                                         1092
                                                             591
  2022-02-02
1
                   1329
                               3233
                                         1490
                                                             923
  2022-02-03
                   1138
                               3340
                                         1322
                                                             754
   2022-02-04
                    811
                               2552
                                          963
                                                             534
4 2022-02-05
                    541
                               1530
                                          602
                                                             326
```

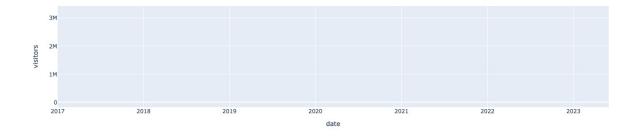
	<pre>avg_session_duration_secs</pre>	awards_week	awards_night
0	148	0	0
1	90	0	0
2	127	0	0
3	142	0	0
4	111	0	0

#### 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 <code>.groupby()</code> to calculate the number of visitors on the site based on the values in the column <code>awards\_night</code>.

```
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']

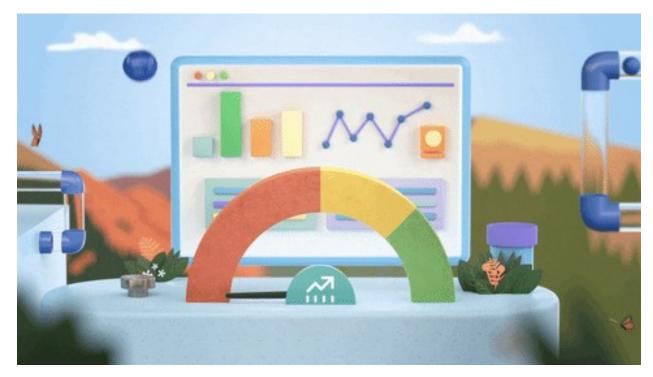
# Run the following cell - DO NOT MODIFY
# .copy() prevents pandas from printing a scary-looking warning
message
combined_site = combined_site.copy()
grammys = grammys.copy()

# 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

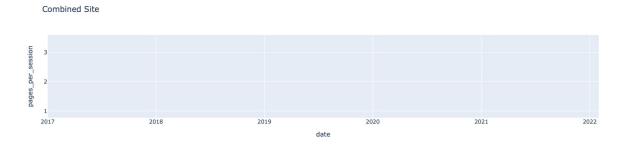
This can be achieved by using the following template:

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

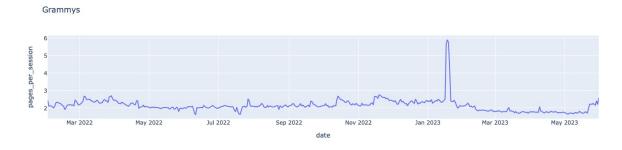
# create the `pages_per_session` column for all 3 dataframes.
for frame in frames:
    frame['pages_per_session'] = frame['pageviews'] /
frame['sessions']
```

**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
- 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 \
     2017-01-01
                     9611
                                21407
                                          10196
                                                              6490
1
     2017-01-02
                    10752
                                25658
                                          11350
                                                              7055
2
     2017-01-03
                    11425
                                27062
                                          12215
                                                              7569
     2017-01-04
3
                    13098
                                29189
                                          13852
                                                              8929
4
     2017-01-05
                    12234
                                28288
                                          12990
                                                              8105
1852 2022-01-27
                                    2
                                                                 2
                         2
                                              2
                                79160
                                          36571
1853 2022-01-28
                    32986
                                                             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
2.10
                             100
                                                           0
1
2.26
```

2		92	2	0	0	
2.22		9(	9	0	0	
2.11 4		9!	5	0	Θ	
2.18		9.	J	U	0	
			•			
1852		(	9	0	0	
1.00 1853		83	3	0	Θ	
2.16						
1854 1.89		63	3	0	0	
1855		6	1	0	0	
1.86 1856		6	7	0	0	
2.20		-		•		
[185]	7 rows x 9 colur	mns],	date	visitors	pageviews	
	ions bounced_se 2022-02-01	essions \ 928	2856	1092	591	
0 1	2022-02-01	1329	3233	1490	923	
2 3 4	2022-02-03 2022-02-04	1138 811	3340 2552	1322 963	754 534	
4	2022-02-04	541	1530	602	326	
 480	2023-05-27	 845	2110	 1058	 702	
481	2023-05-28	702	2100	872	537	
482 483	2023-05-29 2023-05-30	1027 1320	2693 4032	1197 1658	777 992	
484	2023-05-31	1618	5163	2072	1195	
	avg_session_du	ration secs	awards we	ek awards	niaht	
page	s_per_session	_			_	
0 2.62		148		0	0	
1		90		0	0	
2.17		127		0	0	
2.53		142		0	Θ	
3 2.65		142		0		
4 2.54		111		0	0	
2.54						
480		96		0	0	
1.99		30		U	V	

481	137	Θ	0	
2.41	105	•	•	
482 2.25	125	0	0	
483	166	Θ	0	
2.43	100	Ü	· ·	
484	172	0	0	
2.49				
[485 rows x 9 columns]	d	ate visitors	pageviews session	c
bounced sessions \	, u	ate visitors	pageviews session	3
	209 74033	30472	13070	
	511 43642	20761	11814	
	502 44147		12015	
	863 39483		10731	
1861 2022-02-05 18	014 35046	16860	9604	
	 ววว ว <i>и</i> 170	15/20	 5424	
	332 34178 798 31708		5424 5509	
	563 53396		7005	
	105 37950		6452	
	253 85686		8200	
		33237	0200	
avg_session_dura	tion_secs awa	rds_week awar	ds_night	
pages_per_session				
1857	69	0	0	
2.43 1858	85	Θ	0	
2.10	0.3	U	U	
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	07	0	0	
2340 2.20	87	0	0	
2341	106	0	0	
2.58	100	O .	Ü	
[485 rows x 9 columns]	_			
Bounce rate for [	date vi	sitors pagevi	ews sessions	

0										
1		<del>_</del>	0611	2	1407	10106	•		6400	
2 2017-01-03 11425 27062 12215 7569 3 2017-01-04 13098 29189 13852 8929 8105										
3 2017-01-04 13098 29189 13852 8929 4 2017-01-05 12234 28288 12990 8105										
4	2									
1852 2022-01-27       2       0       0       0       2       2       2       2       0       0       0       2       2       2       2       2       2       0       0       0       0       2       2       2       2       2       2       2       2       2       2       2       2       2       2       2       2		2017-01-03								
1853 2022-01-28 32986 79160 36571 20268 1854 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		2022-01-27								
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				7						
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										
1856 2022-01-31       38221       92863       42291       21747         avg_session_duration_secs       awards_week       awards_night         pages_per_session       0       0       0         2.10       1       100       0       0         2.26       92       0       0       0         2.22       92       0       0       0         2.11       4       95       0       0       0         2.18              1852       0       0       0       0       0       0         1853       83       0 <td></td>										
avg_session_duration_secs awards_week awards_night pages_per_session 0										
pages_per_session	1030	2022 01 31	30221	,	2003	72231	_		21/4/	
pages_per_session		avg session	duration se	ecs	awards w	eek	award	s nigh	t	
86	pages		_		_					
2.10 1	0	_, _		86		0			0	
2.26 2	2.10									
2 92 0 0 2.22 3 90 0 0 2.11 4 95 0 0 2.18 1852 0 0 0 1853 83 0 0 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	1		-	L00		0			0	
2 92 0 0 2.22 3 90 0 0 2.11 4 95 0 0 2.18 1852 0 0 0 1853 83 0 0 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	2.26									
90 0 0 0 2.11 4 95 0 0 2.18	2			92		0			0	
2.11 4	2.22									
4 95 0 0 2.18 1852 0 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	3			90		0			0	
2.18	2.11									
	4			95		0			0	
1852	2.18									
1852										
1.00 1853										
1853 83 0 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				0		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										
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				83		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						_			_	
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				63		0			Θ	
1.86 1856 67 0 0 2.20  [1857 rows x 9 columns], date visitors pageviews sessions bounced_sessions \ 0				<b>6</b> 1		•			•	
1856 67 0 0 2.20  [1857 rows x 9 columns], date visitors pageviews sessions bounced_sessions \ 0				ρŢ		Θ			U	
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				67		0			0	
[1857 rows x 9 columns], date visitors pageviews sessions bounced_sessions \ 0    2022-02-01    928    2856    1092    591				0/		U			U	
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	2.20									
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	[185]	7 rows x 0 col	umns 1		date	Vici	tors	nagev	i ews	
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	_		- 1		uate	A T 2 I		pagev	TCW3	
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			<b>-</b>	-	2856	1007	)		501	
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										
4       2022-02-05       541       1530       602       326                 480       2023-05-27       845       2110       1058       702         481       2023-05-28       702       2100       872       537										
4       2022-02-05       541       1530       602       326                 480       2023-05-27       845       2110       1058       702         481       2023-05-28       702       2100       872       537	3									
480       2023-05-27       845       2110       1058       702         481       2023-05-28       702       2100       872       537		2022 02 03								
481 2023-05-28 702 2100 872 537		2023-05-27								
		_325 03 25	202,		_555				,,,	

483 484	2023-05-30 2023-05-31	1320 1618		4032 5163		658 072		992 1195
	avg_session_	_	secs	awards_	_week	awards	_night	
	s_per_session						_	
0			148		0		0	
2.62					_			
1			90		0		0	
2.17			107		•		•	
2			127		0		0	
2.53			1.40		•		0	
3			142		0		0	
2.65			111		0		0	
4			111		0		0	
2.54								
480			96		0		0	
1.99			90		U		U	
481			137		0	1	0	
2.41			137		U		J	
482			125		0		0	
2.25			123		Ū		J	
483			166		0		0	
2.43								
484			172		0		0	
2.49								
_	rows x 9 col	umns],		date	e vi	sitors	pageviews	sessions
	· · · <b>_</b> · · · · · · · · · ·	\					_	
	2022-02-01	33209		74033		472		13070
	2022-02-02	30511		43642		761		1814
	2022-02-03	31502		44147		830		2015
	2022-02-04	26863		39483		700	_	10731
1801	2022-02-05	18014		35046	10	860		9604
2227	2023-05-27	14332		34178	1 5	430		5424
	2023-05-27	13798		31708		662		5509
	2023-05-28	20563		53396		244		7005
	2023-05-29	16105		37950		264		6452
	2023-05-30	31253		85686		237		8200
2341	2023-03-31	31233		03000	))	237		0200
	avg session	duration	secs	awards	s wee	k award	s night	
pages	s per session		_				_ 5	
1857			69			0	Θ	
2.43								
1858			85			0	0	
2.10								
1859			90			0	0	
2.12								

1860							
1861 75 0 0 0 2.08			85		0	0	
2.08							
			75		0	0	
2337	2.08						
2337							
2.22 2338							
2318	2337		75		0	0	
2.16 2339 2.40 2.40 2340 2.70 2.341 2.58  [485 rows x 9 columns]]: 33.67% Bounce rate for [	2.22						
2349	2338		73		0	0	
2.40 2340 87 0 0 2.20 2.20 2341 106 0 0 2.58  [485 rows x 9 columns]]: 33.67%  Bounce rate for [							
2340	2339		92		0	0	
2.20 2341							
2341			87		0	0	
2.58  [485 rows x 9 columns]]: 33.67%  Bounce rate for [							
[485 rows x 9 columns]]: 33.67% Bounce rate for [			106		0	0	
Bounce rate for [	2.58						
Bounce rate for [	_	_					
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         2022-01-30         39931         81186         43743         26636           1856         2022-01-31         38221         92863         42291         21747    avg_session_duration_secs awards_week awards_night  pages_per_session  0							
0		<del>-</del>	dat	e visitor	s pagevi	iews sessions	
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 0 0 0 2.11 4 95 0 0 0 0 2.18 990 0 0 0 2.11 4 95 0 0 0 0 2.18 990 0 0 0 2.11 4 95 0 0 0 0 2.18							
1852 2022-01-27							
1852 2022-01-27	1						
1852 2022-01-27	2						
1852 2022-01-27	3						
1852 2022-01-27	4	2017-01-05	12234	28288	12990	8105	
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 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 0 0 18571 20268 1920 0 0 1931 1920 1931 1931 1931 1931 1931 1931 1931 193							
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 0 0 1855 2022-01-31 38221 92863 42291							
1855 2022-01-30							
1856 2022-01-31 38221 92863 42291 21747  avg_session_duration_secs awards_week awards_night pages_per_session 0							
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 0							
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	1000	2022-01-31	30221	92803	42291	21/4/	
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		avg session du	uration secs	awards v	veek awa i	rds night	
0       86       0       0         2.10       100       0       0         2.26       92       0       0         2.22       90       0       0         3       90       0       0         2.11       95       0       0         2.18            1852       0       0       0         1.00       0       0       0	pages		_	_			
2.10         1       100       0       0         2.26       92       0       0         2.22       90       0       0         3       90       0       0         2.11       95       0       0         2.18            1852       0       0       0         1.00       0       0       0		_, _	86		0	0	
1       100       0       0         2.26       92       0       0         2.22       90       0       0         2.11       95       0       0         2.18            1852       0       0       0         1.00       0       0       0	2.10						
2.26 2 92 0 0 2.22 3 90 0 0 2.11 4 95 0 0 2.18 1852 0 0 0 0 1.00			100		0	0	
2       92       0       0         2.22       90       0       0         2.11       95       0       0         4       95       0       0         2.18            1852       0       0       0         1.00       0       0       0	2.26						
2.22         3       90       0       0         2.11       95       0       0         2.18            1852       0       0       0         1.00       0       0       0			92		0	0	
3 90 0 0 2.11 4 95 0 0 2.18 1852 0 0 0 1.00							
2.11 4 95 0 0 2.18  1852 0 0 0 0 1.00			90		0	0	
4 95 0 0 2.18  1852 0 0 0 0						•	
2.18   1852 0 0 0 1.00			95		0	0	
 1852 0 0 0 1.00			33		-		
1852 0 0 0 1.00							
1852 0 0 0 1.00							
1.00			Θ		0	0	
					-	·	
			83		0	0	
					-		

2.16						
1854		6.	3	0	Θ	
1.89 1855		6	1	0	Θ	
1.86		U	L	U	U	
1856		6	7	0	0	
2.20						
[185]	7 rows x 9 colu	ımns 1	date	visitors	pageviews	
	ions bounced_s		date		pageviens	
0	2022-02-01	928	2856	1092		91
1	2022-02-02	1329	3233	1490		23
2	2022-02-03 2022-02-04	1138 811	3340 2552	1322 963		54 34
4	2022-02-04	541	1530	602		26
480	2023-05-27	845	2110	1058		92
481 482	2023-05-28 2023-05-29	702 1027	2100 2693	872 1197		37 77
483	2023-05-30	1320	4032	1658		92
484	2023-05-31	1618	5163	2072	119	
			d		a sadada	
nage	<pre>avg_session_du s_per_session</pre>	ration_secs	awards_w	eek award	s_nignt	
0	3_pc1_3c331011	148		0	0	
2.62						
1		90		0	0	
2.17		127		0	0	
2.53		12,		Ū	Ü	
3		142		Θ	Θ	
2.65		111		0	0	
4 2.54		111		0	0	
480 1.99		96		0	Θ	
481		137		0	0	
2.41		23.		· ·	J	
482		125		0	0	
2.25 483		166		0	0	
2.43		100		U	U	
484		172		0	0	
2.49						
[/125	rows x 9 colum	nns 1	date	visitors	pageviews	sessions
_	ced sessions \		uate	ATSTERIS	pageviews	3C33T0112
	2022-02-01	33209	74033	30472	130	70

1858 2022-02-02 1859 2022-02-03 1860 2022-02-04 1861 2022-02-05	30511 31502 26863 18014	43642 44147 39483 35046	20761 20830 18700 16860		11814 12015 10731 9604
2337 2023-05-27 2338 2023-05-28 2339 2023-05-29 2340 2023-05-30 2341 2023-05-31	14332 13798 20563 16105 31253	34178 31708 53396 37950 85686	15430 14662 22244 17264 33237		5424 5509 7005 6452 8200
avg session				wards night	0_00
pages per session	daracion_5ee	.s awaras_	week a	war as_nigne	
1857	6	9	0	0	
2.43	_	_	_	_	
1858	8	5	0	0	
2.10 1859	C	0	0	0	
2.12	3	.0	U	O .	
1860	8	5	0	0	
2.11					
1861	7	5	0	0	
2.08					
		•			
2337	7	5	0	0	
2.22					
2338	7	3	0	0	
2.16	0		Θ	0	
2339 2.40	9	2	U	0	
2340	8	37	0	0	
2.20	_		-	_	
2341	10	6	0	0	
2.58					
[485 rows x 9 colu	mns]]: 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 \
                                 21407
     2017-01-01
                      9611
                                            10196
                                                                6490
1
     2017-01-02
                     10752
                                 25658
                                            11350
                                                                7055
2
     2017-01-03
                     11425
                                 27062
                                            12215
                                                                7569
3
     2017-01-04
                                 29189
                                            13852
                     13098
                                                                8929
     2017-01-05
4
                     12234
                                 28288
                                            12990
                                                                8105
1852 2022-01-27
                         2
                                    2
                                                2
                                                                   2
1853 2022-01-28
                     32986
                                 79160
                                            36571
                                                               20268
                                            41920
1854 2022-01-29
                     37899
                                 79095
                                                               25316
1855 2022-01-30
                     39931
                                 81186
                                            43743
                                                               26636
1856 2022-01-31
                                 92863
                                            42291
                     38221
                                                               21747
      avg session duration secs awards week awards night
pages per session
                               86
                                              0
                                                             0
2.10
                              100
1
                                                             0
2.26
2
                               92
                                                             0
2.22
3
                               90
                                              0
                                                             0
2.11
4
                               95
                                                             0
2.18
. . .
. . .
                                0
                                                             0
1852
1.00
1853
                               83
                                                             0
                                              0
2.16
                               63
                                                             0
1854
1.89
1855
                               61
                                                             0
1.86
                                                             0
1856
                               67
                                              0
2.20
[1857 rows x \ 9 \ columns],
                                            visitors pageviews
                                      date
sessions bounced sessions \
     2022-02-01
                                  2856
                                             1092
                                                                 591
0
                       928
1
     2022-02-02
                      1329
                                  3233
                                             1490
                                                                 923
2
     2022-02-03
                      1138
                                  3340
                                             1322
                                                                 754
3
     2022-02-04
                                  2552
                                              963
                                                                 534
                       811
```

4	2022-02-05	541	1530	602	326							
480	2023-05-27	 845	2110	 1058	 702							
481	2023-05-28	702	2100	872	537							
482 483	2023-05-29 2023-05-30	1027 1320	2693 4032	1197 1658	777 992							
484	2023-05-31	1618	5163	2072	1195							
2000	<pre>avg_session_duration_secs awards_week awards_night</pre>											
0	s_per_session	148		0	0							
2.62		90		0	0							
2.17												
2 2.53		127		0	0							
3		142		0	0							
2.65 4		111		0	0							
2.54												
480 1.99		96		0	0							
481		137		0	0							
2.41 482		125		0	0							
2.25												
483 2.43		166		0	0							
484		172		0	0							
2.49												
_	rows x 9 colu		date	visitors	pageviews sessions							
	ced_sessions 2022-02-01	33209	74033	30472	13070							
1858	2022-02-02	30511	43642	20761	11814							
	2022-02-03 2022-02-04	31502 26863	44147 39483	20830 18700	12015 10731							
	2022-02-04	18014	35046	16860	9604							
2337	 2023-05-27	14332	 34178	 15430	 5424							
	2023-05-27	13798	31708	14662	5509							
	2023-05-29	20563	53396	22244	7005							
	2023-05-30 2023-05-31	16105 31253	37950 85686	17264 33237	6452 8200							
	avg_session_duration_secs awards_week awards_night											
1857	s_per_session	6	9	0	Θ							

2.43					_
1858			85	0	0
2.10 1859			90	Θ	0
2.12			90	U	U
1860			85	0	0
2.11				-	-
1861			75	0	0
2.08					
2227			75	0	0
2337 2.22			75	0	0
2338			73	0	0
2.16			, 3	Ū	O .
2339			92	0	0
2.40					
2340			87	0	0
2.20					
2341		-	L06	0	0
2.58					
Mean sess 0 1 2	rows x 9 colu session durat ions bounced_ 2017-01-01 2017-01-03	ion for [ sessions \ 9611 10752 11425	21407 25658 27062	10196 11350 12215	6490 7055 7569
3 1	2017-01-04 2017-01-05	13098 12234	29189 28288	13852 12990	8929 8105
4	2017-01-03	12234	20200	12990	8103
	2022-01-27	2	2	2	2
	2022-01-28	32986	79160	36571	20268
	2022-01-29	37899	79095	41920	25316
	2022-01-30	39931	81186	43743	26636
1856	2022-01-31	38221	92863	42291	21747
page	avg_session_ s_per_session	duration_se	ecs award	ds_week award	ls_night
0			86	0	0
2.10					
1			L00	0	0
2.26				_	
2			92	0	Θ
2.22					
			00	0	Ω
3			90	0	Θ
3 2.11					
3			90 95	0	0 0

1852			0		0			0	
1.00 1853			83		0			0	
2.16 1854			63		0			0	
1.89									
1855 1.86			61		0			0	
1856 2.20			67		0			0	
				al = ±					
sess	7 rows x 9 colurions bounced_se	essions	\	date		sitors	page	views	
0 1	2022-02-01 2022-02-02	928 1329		2856 3233		192 .90		59 92	
2 3	2022-02-03 2022-02-04	1138 811		3340 2552		622 63		75 53	
4	2022-02-04	541		1530		02		32	
480	 2023-05-27	845		2110		58		70	
481 482	2023-05-28 2023-05-29	702 1027		2100 2693		.97		53 77	
483 484	2023-05-30 2023-05-31	1320 1618		4032 5163	16	58 172		99 119	2
404									,5
page	avg_session_du s_per_session	ration_se	CS	awards_v	week	awards	_nigh	it	
0 2.62		1	48		0			0	
1			90		0			0	
2.17		1	27		0			0	
2.53		1	42		0			0	
2.65 4		1	11		0			0	
2.54		_	11		U			U	
			• •					•	
480 1.99			96		0			0	
481		1	37		0			0	
2.41 482		1	25		0			0	
2.25 483		1	66		0			0	
2.43									

484 2.49	172	2	Θ	0	
[485 rows x 9 columbounced sessions \		date	visitors	pageviews	sessions
1857 2022-02-01 1858 2022-02-02 1859 2022-02-03 1860 2022-02-04 1861 2022-02-05	33209 30511 31502 26863 18014	74033 43642 44147 39483 35046	30472 20761 20830 18700 16860	11 12 10	070 814 015 731 604
2337 2023-05-27 2338 2023-05-28 2339 2023-05-29 2340 2023-05-30 2341 2023-05-31	14332 13798 20563 16105 31253	34178 31708 53396 37950 85686	15430 14662 22244 17264 33237	5 7 6	424 509 005 452 200
avg_session_c pages_per_session	uration_sed	s awards_	week awar	ds_night	
1857 2.43	6	59	0	0	
1858	8	35	0	0	
2.10 1859 2.12	Ğ	90	0	0	
1860 2.11	3	35	0	0	
1861 2.08	7	75	0	0	
		•			
2337 2.22	7	75	0	0	
2338	7	<b>7</b> 3	0	0	
2.16 2339	Ć	92	0	0	
2.40 2340	8	37	0	0	
2.20 2341 2.58	16	06	0	Θ	
[485 rows x 9 colum Mean session durati sessions bounced_s 0 2017-01-01	on for [		te visito 10196		ws 490
1 2017-01-02 2 2017-01-03 3 2017-01-04 4 2017-01-05	10752 11425 13098 12234	25658 27062 29189 28288	11350 12215 13852 12990	7 7 8	055 569 929 105

1853 1854 1855	2022-01-27 2022-01-28 2022-01-29 2022-01-30 2022-01-31	32986 37899 39931 38221	79160 79095 81186 92863	2 36571 41920 43743 42291	20268 20268 25316 26636 21747		
page	avg_session_o s per session	duration_se	cs awards	_week award	ls_night		
0		}	36	0	0		
2.10 1		10	90	0	0		
2.26 2			92	0	0		
2.22		9	90	0	0		
2.11 4		(	95	0	Θ		
2.18							
		•					
1852 1.00			0	0	Θ		
1853		{	83	0	0		
2.16 1854		(	63	0	0		
1.89 1855			61	0	0		
1.86			67	0	0		
1856 2.20			67	0	Θ		
	7 rows x 9 col		dat	e visitors	pageviews		
0 1 2 3 4	ions bounced_: 2022-02-01 2022-02-02 2022-02-03 2022-02-04 2022-02-05	sessions \     928     1329     1138     811     541	2856 3233 3340 2552 1530	1092 1490 1322 963 602	591 923 754 534 326		
480 481 482 483 484	2023-05-27 2023-05-28 2023-05-29 2023-05-30 2023-05-31	845 702 1027 1320 1618	2110 2100 2693 4032 5163	1058 872 1197 1658 2072	702 537 777 992 1195		
<pre>avg_session_duration_secs awards_week awards_night pages_per_session</pre>							
0		148	8	0	0		
2.62							

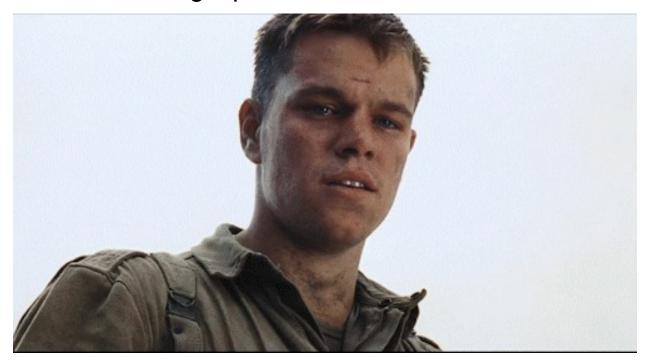
1	90		0	0	
2.17	90		U	O	
2	127		0	Θ	
2.53					
3	142		0	0	
2.65			_	_	
4	111		Θ	0	
2.54					
• •			• • •		
480	96		0	0	
1.99	30		U	O	
481	137		Θ	0	
2.41			•	•	
482	125		0	0	
2.25					
483	166		0	Θ	
2.43					
484	172		0	0	
2.49					
[485 rows x 9 col	ump c 1	date	visitors	pageviews	coccions
bounced_sessions	\ \	uate	ATSTUUS	pageviews	262210112
1857 2022-02-01	33209	74033	30472	13	070
1858 2022-02-02	30511	43642	20761		814
1859 2022-02-03	31502	44147	20830		015
1860 2022-02-04	26863	39483	18700	10	731
1861 2022-02-05	18014	35046	16860	9	604
2337 2023-05-27	14332	34178	15430		424
2338 2023-05-28	13798	31708	14662		509
2339 2023-05-29	20563	53396	22244		005
2340 2023-05-30	16105	37950	17264		452
2341 2023-05-31	31253	85686	33237	8	200
ava coccion	_duration_sec	s awarde	wook awar	de night	
pages_per_session		s awarus_	_week awai	us_nignt	
1857	69	9	0	0	
2.43	0.		ŭ	· ·	
1858	8!	5	0	Θ	
2.10					
1859	90	9	0	0	
2.12					
1860	8!	5	0	0	
2.11					
1861	7!	5	0	0	
2.08					

2337	75	0	0
2.22 2338	73	0	0
2.16	02	0	0
2339 2.40	92	0	0
2340 2.20	87	0	0
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. <u>18 24</u> 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
                    24.13
      25-34
1
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** 

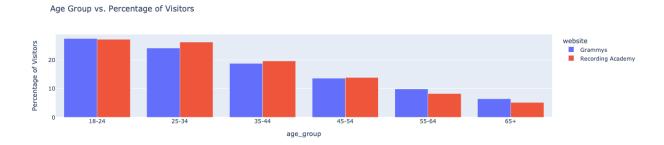
**Hint:** Remember that you need to put your dataframe variables inside of a **list** first then pass that as your input of pd.concat()

```
# 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.



**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()
                      segment visitors
        date
 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()
                     segment visitors
        date
 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.

```
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
deprecate nonkeyword arguments.<locals>.decorate.<locals>.wrapper(*arg
s, **kwarqs)
    305 if len(args) > num allow args:
    306
            warnings.warn(
                msq.format(arguments=arguments),
    307
    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,
   (\ldots)
            errors: str = "raise",
   4783
   4784 ):
            11 11 11
   4785
   4786
            Drop specified labels from rows or columns.
   4787
   (\ldots)
   4904
                    weight 1.0
                                     0.8
   4905
-> 4906
            return super().drop(
   4907
                labels=labels,
   4908
                axis=axis.
   4909
                index=index.
                columns=columns,
   4910
                level=level,
   4911
   4912
                inplace=inplace,
   4913
                errors=errors,
   4914
            )
/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:
            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)
                new axis = axis.drop(labels, level=level,
   4183
errors=errors)
   4184
          else:
                new axis = axis.drop(labels, errors=errors)
-> 4185
   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():
            if errors != "ignore":
   6016
-> 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()
                                            mobile visitors
        date desktop visitors
                                      date
  2022-02-01
                         10195
                                2022-02-01
                                                      23494
                         10560 2022-02-02
1 2022-02-02
                                                      20234
  2022-02-03
                                2022-02-03
                          9935
                                                      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
      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)):
            result = assemble from unit mappings(arg, errors, tz)
    891 elif isinstance(arg, Index):
            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 \text{ arg} = DataFrame(arg)
```

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)