In [2]:
```
import pandas as pd
import numpy as np
```

Content data exploration

In [3]:
```
content = pd.read_csv("/Users/thiernodicko/Desktop/Panda_files/Accenture data/
```

In [4]:
```
content.head(5)
```

Out[4]:

	Unnamed: 0	Content ID	User ID	Type	Category	
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	8d3cd87d-8a31-4935-9a4f-b319bfe05f31	photo	Studying	https://socialbuzz.cdn.com/content/
1	1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	beb1f34e-7870-46d6-9fc7-2e12eb83ce43	photo	healthy eating	https://socialbuzz.cdn.com/content
2	2	230c4e4d-70c3-461d-b42c-ec09396efb3f	a5c65404-5894-4b87-82f2-d787cbee86b4	photo	healthy eating	https://socialbuzz.cdn.com/content/
3	3	356fff80-da4d-4785-9f43-bc1261031dc6	9fb4ce88-fac1-406c-8544-1a899cee7aaf	photo	technology	https://socialbuzz.cdn.com/content/
4	4	01ab84dd-6364-4236-abbb-3f237db77180	e206e31b-5f85-4964-b6ea-d7ee5324def1	video	food	https://socialbuzz.cdn.com/content

Let remove this duplicate column called unnamed from the dataset

In [5]:
```
content = content.loc[:, ~content.columns.str.contains('^Unnamed')]
```

In [6]:
```
content.head(5)
```

Out[6]:

	Content ID	User ID	Type	Category	UR
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	8d3cd87d-8a31-4935-9a4f-b319bfe05f31	photo	Studying	https://socialbuzz.cdn.com/content/storage/975.
1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	beb1f34e-7870-46d6-9fc7-2e12eb83ce43	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/9f7.
2	230c4e4d-70c3-461d-b42c-ec09396efb3f	a5c65404-5894-4b87-82f2-d787cbee86b4	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/230.
3	356fff80-da4d-4785-9f43-bc1261031dc6	9fb4ce88-fac1-406c-8544-1a899cee7aaf	photo	technology	https://socialbuzz.cdn.com/content/storage/356.
4	01ab84dd-6364-4236-abbb-3f237db77180	e206e31b-5f85-4964-b6ea-d7ee5324def1	video	food	https://socialbuzz.cdn.com/content/storage/01a.

Showing a general information of the data

In [7]:

```
content.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Content ID  1000 non-null   object
 1   User ID     1000 non-null   object
 2   Type        1000 non-null   object
 3   Category    1000 non-null   object
 4   URL         801 non-null    object
dtypes: object(5)
memory usage: 39.2+ KB
```

Checking for the existance of null values

In [8]:

```
content.isna().sum()
```

Out[8]:

```
Content ID      0
User ID         0
Type            0
Category        0
URL           199
dtype: int64
```

Rename the column Type to Content Type

In [9]:

```
content.rename(columns={'Type':'Content Type'}, inplace=True)
```

Since both User ID and URL columns seems irrelevant to us, lets remove them.

In [10]:
```
content.drop(['User ID', 'URL'], axis = 1, inplace=True)
```

In [11]:
```
content.head(5)
```

Out[11]:

	Content ID	Content Type	Category
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	Studying
1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	photo	healthy eating
2	230c4e4d-70c3-461d-b42c-ec09396efb3f	photo	healthy eating
3	356fff80-da4d-4785-9f43-bc1261031dc6	photo	technology
4	01ab84dd-6364-4236-abbb-3f237db77180	video	food

Displaying all rows that contains null values

In [12]:
```
content.isna().sum()
```

Out[12]:
```
Content ID      0
Content Type    0
Category        0
dtype: int64
```

In [13]:
```
content.shape
```

Out[13]:
```
(1000, 3)
```

The content table seems now to be clean and ready for further analysis

Reaction Data exploration

In [14]:
```
reaction = pd.read_csv("/Users/thiernodicko/Desktop/Panda_files/Accenture data/
```

In [15]:
```
reaction.head(5)
```

Out[15]:

	Unnamed: 0	Content ID	User ID	Type	Datetime
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	NaN	2021-04-22 15:17:15
1	1	97522e57-d9ab-4bd6-97bf-c24d952602d2	5d454588-283d-459d-915d-c48a2cb4c27f	disgust	2020-11-07 09:43:50
2	2	97522e57-d9ab-4bd6-97bf-c24d952602d2	92b87fa5-f271-43e0-af66-84fac21052e6	dislike	2021-06-17 12:22:51
3	3	97522e57-d9ab-4bd6-97bf-c24d952602d2	163daa38-8b77-48c9-9af6-37a6c1447ac2	scared	2021-04-18 05:13:58
4	4	97522e57-d9ab-4bd6-97bf-c24d952602d2	34e8add9-0206-47fd-a501-037b994650a2	disgust	2021-01-06 19:13:01

Showing a general information of the data

In [16]:

```
reaction.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25553 entries, 0 to 25552
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  25553 non-null  int64 
 1   Content ID  25553 non-null  object
 2   User ID     22534 non-null  object
 3   Type        24573 non-null  object
 4   Datetime    25553 non-null  object
dtypes: int64(1), object(4)
memory usage: 998.3+ KB
```

Remove unnamed column from the dataset

In [17]:

```
reaction = reaction.loc[:, ~reaction.columns.str.contains('^Unnamed')]
```

In [18]:

```
reaction.head(5)
```

Out[18]:

	Content ID	User ID	Type	Datetime
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	NaN	2021-04-22 15:17:15
1	97522e57-d9ab-4bd6-97bf-c24d952602d2	5d454588-283d-459d-915d-c48a2cb4c27f	disgust	2020-11-07 09:43:50
2	97522e57-d9ab-4bd6-97bf-c24d952602d2	92b87fa5-f271-43e0-af66-84fac21052e6	dislike	2021-06-17 12:22:51
3	97522e57-d9ab-4bd6-97bf-c24d952602d2	163daa38-8b77-48c9-9af6-37a6c1447ac2	scared	2021-04-18 05:13:58
4	97522e57-d9ab-4bd6-97bf-c24d952602d2	34e8add9-0206-47fd-a501-037b994650a2	disgust	2021-01-06 19:13:01

Changing the data type of the datetime column to a date format

In [19]:

```
reaction['Datetime'] = pd.to_datetime(reaction['Datetime'])
```

In [20]:

```
# Checking if update was successfull
reaction.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25553 entries, 0 to 25552
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype         
---  ------      --------------  -----         
 0   Content ID  25553 non-null  object        
 1   User ID     22534 non-null  object        
 2   Type        24573 non-null  object        
 3   Datetime    25553 non-null  datetime64[ns]
dtypes: datetime64[ns](1), object(3)
memory usage: 798.7+ KB
```

Renaming the Type column to reaction type

In [21]:

```
reaction.rename(columns={'Type': 'Reaction Type'}, inplace=True)
```

Let also remove the User ID columns from the reaction table

In [22]:

```
reaction.drop('User ID', axis=1, inplace=True)
```

In [23]:

```
# Update verification
reaction.head(5)
```

Out[23]:

	Content ID	Reaction Type	Datetime
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	2021-04-22 15:17:15
1	97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2020-11-07 09:43:50
2	97522e57-d9ab-4bd6-97bf-c24d952602d2	dislike	2021-06-17 12:22:51
3	97522e57-d9ab-4bd6-97bf-c24d952602d2	scared	2021-04-18 05:13:58
4	97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2021-01-06 19:13:01

Checking for the existance of null values

In [24]:

```
reaction.isna().sum()
```

Out[24]:

```
Content ID         0
Reaction Type    980
Datetime           0
dtype: int64
```

Displaying all rows that contains missing values

In [25]:

```
reaction[reaction['Reaction Type'].isnull()]
```

Out[25]:

	Content ID	Reaction Type	Datetime
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	2021-04-22 15:17:15
46	9f737e0a-3cdd-4d29-9d24-753f4e3be810	NaN	2020-12-04 20:00:31
62	230c4e4d-70c3-461d-b42c-ec09396efb3f	NaN	2021-03-19 08:19:38
94	356fff80-da4d-4785-9f43-bc1261031dc6	NaN	2020-08-28 23:43:55
102	01ab84dd-6364-4236-abbb-3f237db77180	NaN	2021-02-08 21:55:56
...	...	...	...
25445	b4cef9ef-627b-41d7-a051-5961b0204ebb	NaN	2020-11-30 15:26:32
25449	7a79f4e4-3b7d-44dc-bdef-bc990740252c	NaN	2021-04-04 19:39:36
25454	435007a5-6261-4d8b-b0a4-55fdc189754b	NaN	2021-01-04 20:28:29
25499	4e4c9690-c013-4ee7-9e66-943d8cbd27b7	NaN	2021-05-25 18:05:31
25540	75d6b589-7fae-4a6d-b0d0-752845150e56	NaN	2021-04-25 05:09:20

980 rows × 3 columns

In [26]:

```
# Dropping rows with missing values
reaction.dropna(inplace=True)
```

In [27]:

```
reaction.shape
```

Out[27]:

```
(24573, 3)
```

The Reaction table seems now to be clean and ready for further analysis

Reaction Type data Exploration

In [28]:

```
reaction_type = pd.read_csv("/Users/thiernodicko/Desktop/Panda_files/Accenture
```

In [29]:

```
reaction_type.head(5)
```

Out[29]:

	Unnamed: 0	Type	Sentiment	Score
0	0	heart	positive	60
1	1	want	positive	70
2	2	disgust	negative	0
3	3	hate	negative	5
4	4	interested	positive	30

Displaying a general information of the data

In [30]:

```
reaction_type.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16 entries, 0 to 15
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  16 non-null     int64 
 1   Type        16 non-null     object
 2   Sentiment   16 non-null     object
 3   Score       16 non-null     int64 
dtypes: int64(2), object(2)
memory usage: 644.0+ bytes
```

Removing unnamed column from the dataset

In [31]:

```
reaction_type = reaction_type.loc[:, ~reaction_type.columns.str.contains('^Unna
```

Checking for the existance of null values

In [32]:

```
reaction_type.isna().sum()
```

Out[32]:

```
Type         0
Sentiment    0
Score        0
dtype: int64
```

Rename the Type column to Reaction Type

In [33]:

```
reaction_type.rename(columns={'Type':'Reaction Type'}, inplace=True)
```

In [34]:

```
reaction_type.head(5)
```

Out[34]:

	Reaction Type	Sentiment	Score
0	heart	positive	60
1	want	positive	70
2	disgust	negative	0
3	hate	negative	5
4	interested	positive	30

The Reaction Type table seems now to be clean and ready for further analysis

Data Merging

Step 1: Let use the Reaction Table as base table, then join the relevant columns from the Content data set. Step 2: Use result from step 1 as base table to join with the relevant columns from the Reaction Type data set.

In [35]:

```
# Step 1:

first_merge = reaction.set_index('Content ID').join(content.set_index('Content
```

In [36]:
```
first_merge
```

Out[36]:

	Reaction Type	Datetime	Content Type	Category
Content ID				
97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2020-11-07 09:43:50	photo	Studying
97522e57-d9ab-4bd6-97bf-c24d952602d2	dislike	2021-06-17 12:22:51	photo	Studying
97522e57-d9ab-4bd6-97bf-c24d952602d2	scared	2021-04-18 05:13:58	photo	Studying
97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2021-01-06 19:13:01	photo	Studying
97522e57-d9ab-4bd6-97bf-c24d952602d2	interested	2020-08-23 12:25:58	photo	Studying
...	...	...	...	...
75d6b589-7fae-4a6d-b0d0-752845150e56	dislike	2020-06-27 09:46:48	audio	technology
75d6b589-7fae-4a6d-b0d0-752845150e56	intrigued	2021-02-16 17:17:02	audio	technology
75d6b589-7fae-4a6d-b0d0-752845150e56	interested	2020-09-12 03:54:58	audio	technology
75d6b589-7fae-4a6d-b0d0-752845150e56	worried	2020-11-04 20:08:31	audio	technology
75d6b589-7fae-4a6d-b0d0-752845150e56	cherish	2021-01-04 04:55:11	audio	technology

24573 rows × 4 columns

In [37]:
```
# Step 2
final_data = first_merge.set_index('Reaction Type').join(reaction_type.set_inde
```

In [38]:
```
# Setting the datetime column as index column

final_data.set_index('Datetime', inplace=True)
```

In [39]:
```
# Displaying 10 first rows of the final data

final_data.head(10)
```

Out[39]:

	Content Type	Category	Sentiment	Score
Datetime				
2020-11-07 09:43:50	photo	Studying	negative	0
2021-06-17 12:22:51	photo	Studying	negative	10
2021-04-18 05:13:58	photo	Studying	negative	15
2021-01-06 19:13:01	photo	Studying	negative	0
2020-08-23 12:25:58	photo	Studying	positive	30
2020-12-07 06:27:54	photo	Studying	neutral	35
2021-04-11 17:35:49	photo	Studying	positive	70
2021-01-27 08:32:09	photo	Studying	negative	5
2021-04-01 22:54:23	photo	Studying	neutral	35
2020-08-04 05:05:02	photo	Studying	positive	65

The final data looks reasonable, therefore let export it to a csv file for further analysis

In [40]:
```
final_data.to_csv('Accenture_data_Exploration.csv')
```

Next: I will use SQL to figure out the top 5 categories with the large popularity

In [ ]:
```
```