In [ ]:

```python
from config import views
from spark import createSession

from typing import List, Tuple

from matplotlib import pyplot as plt
from pyspark.sql.dataframe import DataFrame

import pyspark.sql.functions as F
import pyspark.sql.types as T

from IPython.display import display
```

In [ ]:

```python
def get_columns_of_type(data_frame: DataFrame, type: str) -> List[str]:
    return [column[0] for column in data_frame.dtypes if column[1] == type]
```

In [ ]:

```python
LENGTH = 80
def show_table_name(table: str) -> None:
    print('=' * LENGTH)
    print(' ' * ((LENGTH - len(table)) // 2), table.upper())
    print('=' * LENGTH)

def show_column_name(column: str) -> None:
    print(column.upper())
```

In [ ]:

```python
VERSION = 'v1'

VIEWS = views(VERSION)
spark = createSession()

for view, file in VIEWS.items():
    df = spark.read.json(file)
    for column in get_columns_of_type(df, 'boolean'):
        df = df.withColumn(column, F.col(column).cast(T.IntegerType()))

    for column in df.columns:
        if column in ['timestamp', 'release_date']:
            df = df.withColumn(f'{column}_s', F.unix_timestamp(column, "yyyy[-MM[-dd[['T'][' ']HH:mm[:ss[.SSSSSS]]]]]"))

    df.createOrReplaceTempView(view)
```

```
your 131072x1 screen size is bogus. expect trouble
23/04/05 19:14:41 WARN Utils: Your hostname, LAPTOP-7KCON786 resolves to a loopback address: 127.0.1.1; using 192.168.18.206 instead (on interface eth0)
23/04/05 19:14:41 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/04/05 19:14:43 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

In [ ]:

```python
DATA_FRAMES = list(zip(VIEWS.keys(), [spark.sql(f"SELECT * FROM {view}") for view in VIEWS.keys()]))
```

In [ ]:

```python
for view, df in DATA_FRAMES:
    show_table_name(view)
    for column, type in df.dtypes:
        print(column.upper(), '-', type)

    try:
        dfp = df.limit(100_000).toPandas()
        display(dfp)
    except Exception as e:
        df.show()
        print(df.count(), 'rows')
```

```
================================================================================
                                     ARTISTS
================================================================================
GENRES - array<string>
ID - string
NAME - string
```

	genres	id	name
0	[filmi, indian folk, indian rock, kannada pop]	72578usTM6Cj5qWsi471Nc	Raghu Dixit
1	[desi pop, hindi indie, indian indie, indian r...	7b6Ui7JVaBDEfZB9k6nHL0	The Local Train
2	[indian folk]	4bvGDTEPFnllKiJaEZGuXk	Achint
3	[opm, pinoy hip hop, pinoy r&b, pinoy trap, ta...	0n4a5imdLBN24fIrBWoqrv	Because
4	[hindi indie, indian indie, indian singer-song...	4gdMJYnopf2nEUcanAwstx	Anuv Jain
...	...	...	...
27519	[italian hip hop]	2My6j5BEgOi8VHi5WGVyfw	Apocalypshit Army
27520	[belgian pop]	0bzW9kGcTyMxXuG9dUdj7E	GRANDGEORGE
27521	[thai indie]	4iS19hLpsgRd8jLPKI4Ni3	Blissonic
27522	[thai indie]	3JGC3LkYrwlrTscixVwY72	พราว
27523	[indie folk]	-1	Haroula Rose

27524 rows × 3 columns

```
================================================================================
                                     SESSIONS
================================================================================
EVENT_TYPE - string
SESSION_ID - bigint
TIMESTAMP - string
TRACK_ID - string
USER_ID - bigint
TIMESTAMP_S - bigint
```

	event_type	session_id	timestamp	track_id	user_id	timestamp_s
0	PLAY	124	2022-04-19T10:14:08	2FPjk7EjEHD4qgLSSnsWEL	101.0	1650356048
1	PLAY	124	2022-04-19T10:18:17.973000	6yUmkCTAkHECG4btrYw3cM	101.0	1650356297
2	SKIP	124	2022-04-19T10:19:03.410000	6yUmkCTAkHECG4btrYw3cM	101.0	1650356343
3	ADVERTISEMENT	124	2022-04-19T10:19:03.410000		101.0	1650356343
4	BUY_PREMIUM	124	2022-04-19T10:19:20.410000		NaN	1650356360
...	...	...	...	...	...	...
3800	PLAY	794	2023-02-19T04:44:21.306000	1A9dNiCCSsiDkIj1RsqMvL	150.0	1676778261
3801	PLAY	794	2023-02-19T04:47:51.453000	11lkONbH7vsMZEVy012slM	150.0	1676778471
3802	PLAY	794	2023-02-19T04:51:02.119000	2aMsfiqLC8bMHT6FrrGWY4	150.0	1676778662
3803	SKIP	794	2023-02-19T04:52:58.495000	2aMsfiqLC8bMHT6FrrGWY4	NaN	1676778778
3804	PLAY	794	2023-02-19T04:52:58.495000	6Lphoo9KKcpy0QwJWj1vdG	NaN	1676778778

3805 rows × 6 columns

```
================================================================================
                                  TRACK_STORAGE
================================================================================
DAILY_COST - double
STORAGE_CLASS - string
TRACK_ID - string
```

	daily_cost	storage_class	track_id
0	0.003752	SLOW	708ZiYL3ydBWHS2a7gvJB3
1	0.014561	SLOW	48SFtLr5URCI97X2Ynfdnc
2	0.008304	SLOW	1y0U0HAe5QfTRzOsz74bOt
3	0.012207	SLOW	2TlbZ8JhF9ORa7lJylxABw
4	0.011799	SLOW	7ij5kN8jwXr8fZD54M0xb6
...	...	...	...
99995	0.012688	SLOW	3flurnTXJlSjMa9yj2uvY0
99996	0.010389	SLOW	6UjVlcCLMmwfyZfumUhsgN
99997	0.011977	SLOW	2OXAWAySnYPJHLvgLX5fFT
99998	0.008842	SLOW	1hQreq8n3jTwLWD1sjVb3t
99999	0.011849	SLOW	6DVY3lXlOgbu0iD5BhkWXj

100000 rows × 3 columns

```
================================================================================
                                     TRACKS
================================================================================
ACOUSTICNESS - double
DANCEABILITY - double
DURATION_MS - bigint
ENERGY - double
EXPLICIT - bigint
ID - string
ID_ARTIST - string
INSTRUMENTALNESS - double
KEY - bigint
LIVENESS - double
LOUDNESS - double
NAME - string
POPULARITY - bigint
RELEASE_DATE - string
SPEECHINESS - double
TEMPO - double
VALENCE - double
RELEASE_DATE_S - bigint
```

	acousticness	danceability	duration_ms	energy	explicit	id	id_artist	instrumentalness	key	liveness	loudness	name	popularity	release_date	speechiness	tempo	valence	release_date_s
0	0.8390	0.740	75040	0.8910	0	None	None	0.000000	7	0.869	-7.480	031 - Der Schatz im Silbersee I - Teil 39	13.0	1968-09-11	0.8920	51.496	0.557	-41216400
1	0.6950	0.603	291227	0.5170	0	48SFtLr5URCI97X2Ynfdnc	2yTUYhIf8fxptTly3KLuJD	0.000003	6	0.744	-8.504	Par Avion (Live) (2014 - Remaster) - Live; 20...	0.0	2014	0.0235	96.181	0.327	1388530800
2	0.9530	0.313	166080	0.1160	0	1y0U0HAe5QfTRzOsz74bOt	338mC0yGyX0C9of8QMJ5hK	0.331000	0	0.161	-12.645	My Foolish Heart	25.0	1950-01-01	0.0319	74.071	0.255	-631155600
3	0.1670	0.958	244133	0.6350	0	2TlbZ8JhF9ORa7lJylxABw	5A4ExW2nMBFRy2JDoYUcUE	0.000000	11	0.362	-7.853	Kathysterisi	14.0	1998	0.2590	108.024	0.866	883609200
4	0.1200	0.684	235974	0.8390	0	7ij5kN8jwXr8fZD54M0xb6	48CUA59SDed3IdCctKndud	0.000000	4	0.354	-6.457	Aleni Aleni	51.0	2015	0.0658	128.051	0.580	1420066800
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
99995	0.4180	0.874	253755	0.6250	1	3flurnTXJlSjMa9yj2uvY0	2QDHxmDObOuv9MCeBYiFtq	0.000136	5	0.131	-8.277	Şampanya	60.0	2019-07-19	0.0656	117.094	0.461	1563487200
99996	0.7090	0.610	207771	0.5380	0	6UjVlcCLMmwfyZfumUhsgN	3iVIrcJmrV7GawrxVWsBUF	0.002230	7	0.302	-11.594	Başıma Gelenler	20.0	1978	0.0379	105.682	0.677	252457200
99997	0.0469	0.693	239533	0.9050	1	2OXAWAySnYPJHLvgLX5fFT	4oLZx5FplbgfM8DEe9U8LB	0.000000	0	0.268	-8.701	Luchini Aka This Is It	45.0	1990-01-01	0.3030	82.911	0.832	631148400
99998	0.9940	0.462	176842	0.0444	0	1hQreq8n3jTwLWD1sjVb3t	2e42axkOGHNvACKRN4MfDU	0.874000	7	0.148	-21.646	Agg Lagi	0.0	1946-01-01	0.0381	128.364	0.314	-757386000
99999	0.1030	0.737	236987	0.5750	1	None	None	0.000016	11	0.655	-7.001	My Lady - P-Money Mix	36.0	2003-01-01	0.2010	82.549	0.671	1041375600

100000 rows × 18 columns

```
================================================================================
                                      USERS
================================================================================
CITY - string
FAVOURITE_GENRES - array<string>
ID - bigint
NAME - string
PREMIUM_USER - int
STREET - string
USER_ID - bigint
```

	city	favourite_genres	id	name	premium_user	street	user_id
0	Warszawa	[motown, soul, regional mexican]	NaN	Marika Pilipczuk	1.0	ul. Księżycowa 31	101
1	Gdynia	[regional mexican, psychedelic rock, new roman...	NaN	Anita Pioch	1.0	plac Sadowa 527	102
2	Kraków	[soul, mellow gold, blues rock]	NaN	Jan Gryga	1.0	plac Wyspiańskiego 73/43	103
3	Wrocław	[permanent wave, post-teen pop, mandopop]	NaN	Ksawery Klus	1.0	ulica Długosza 71/06	104
4	Gdynia	[metal, new wave, argentine rock]	NaN	Maciej Bandyk	1.0	ul. Rybacka 07	105
5	Kraków	None	NaN	Nikodem Kopciuch	1.0	pl. Promienna 59/43	106
6	Poznań	[europop, folk, tropical]	NaN	Kacper Osojca	1.0	pl. Staszica 343	107
7	Warszawa	[new wave, psychedelic rock, soft rock]	NaN	Maurycy Szoka	1.0	al. Tęczowa 332	108
8	Szczecin	[roots rock, latin pop, alternative metal]	NaN	Sebastian Molka	1.0	al. Armii Krajowej 564	109
9	Kraków	[lounge, hoerspiel, album rock]	NaN	Filip Kalinka	1.0	aleja Bema 889	110
10	Poznań	[classic rock, pop rock, soft rock]	NaN	Krzysztof Wojtach	1.0	aleja Prusa 830	111
11	Gdynia	[rock en espanol, new wave pop, italian adult ...	NaN	Melania Gałat	1.0	pl. Mazurska 345	112
12	Poznań	[new romantic, art rock, new wave]	NaN	Stefan Bisaga	1.0	aleja Tartaczna 95	113
13	Radom	[pop, new wave pop, motown]	-1.0	Dawid Koperek	1.0	al. Podleśna 00	114
14	Kraków	[new romantic, country rock, brill building pop]	NaN	Albert Brzeźniak	1.0	plac Floriana 59/72	115
15	Wrocław	[c-pop, motown, tropical]	NaN	Borys Matula	1.0	al. Szeroka 27/38	116
16	Warszawa	[roots rock, modern rock, j-pop]	NaN	Julianna Więckiewicz	1.0	aleja Urocza 19	117
17	Warszawa	[modern rock, adult standards, pop rock]	NaN	Oskar Jarosik	1.0	ul. Konopnickiej 038	118
18	Radom	[pop rock, europop, hoerspiel]	NaN	Blanka Szklarek	1.0	al. Jesionowa 47	119
19	Poznań	[modern rock, tropical, adult standards]	NaN	Monika Sypień	1.0	pl. Daszyńskiego 80/41	120
20	Kraków	[alternative rock, alternative metal, vocal jazz]	NaN	Kornel Dacko	1.0	plac Kazimierza Wielkiego 51	121
21	Gdynia	[soul, lounge, pop rock]	NaN	Bartek Garczyk	1.0	ulica Wiązowa 07/54	122
22	Kraków	[blues rock, lounge, post-teen pop]	NaN	Maurycy Hutyra	1.0	aleja Stolarska 554	123
23	Radom	[alternative rock, permanent wave, latin pop]	NaN	Oliwier Smalec	1.0	ul. Diamentowa 44	124
24	Wrocław	[funk, classic rock, europop]	NaN	Fryderyk Chabior	1.0	ulica Torowa 80	125
25	Warszawa	[motown, vocal jazz, mandopop]	NaN	Nicole Gajdzik	1.0	pl. Orzeszkowej 21	126
26	Poznań	[italian adult pop, lounge, folk rock]	NaN	Krzysztof Żuchowicz	1.0	pl. Radosna 86/89	127
27	Radom	[adult standards, mpb, funk]	NaN	Janina Delekta	NaN	ulica Mokra 71	128
28	Radom	[vocal jazz, pop rock, soul]	NaN	Nikodem Wawrzynowicz	1.0	plac Słowicza 73	129
29	Radom	[rock en espanol, rock, latin]	NaN	Anna Maria Ignatiuk	1.0	ul. Ciasna 73	130
30	Kraków	[mellow gold, c-pop, argentine rock]	NaN	Arkadiusz Krzywoń	1.0	plac Składowa 526	131
31	Warszawa	[j-pop, folk rock, metal]	NaN	Gustaw Pilipczuk	1.0	aleja Zaułek 750	132
32	Radom	[rock, lounge, metal]	NaN	Łukasz Pielka	1.0	ulica Irysowa 483	133
33	Warszawa	[mpb, permanent wave, hoerspiel]	NaN	Jerzy Husak	NaN	pl. Jagiellońska 607	134
34	Szczecin	[regional mexican, mellow gold, folk rock]	NaN	Filip Łukowiak	1.0	ul. Brzoskwiniowa 81	135
35	Radom	[latin rock, rock, folk rock]	NaN	Eryk Kołata	1.0	ulica Księżycowa 11	136
36	Gdynia	[motown, regional mexican, folk]	NaN	Cezary Getka	1.0	ulica Szpitalna 18	137
37	Warszawa	[ranchera, new romantic, adult standards]	NaN	Kazimierz Stypa	1.0	ulica Konwaliowa 74	138
38	Warszawa	[regional mexican, rock en espanol, argentine ...	-1.0	Andrzej Doktor	1.0	pl. Jana 300	139
39	Gdynia	[italian adult pop, funk, italian adult pop]	NaN	Sylwia Feret	1.0	al. Konwaliowa 33	140
40	Gdynia	[post-teen pop, psychedelic rock, latin altern...	NaN	Radosław Musiolik	1.0	pl. Słonecznikowa 79	141
41	Wrocław	[alternative rock, adult standards, pop]	NaN	Ignacy Pniak	1.0	ulica Witosa 98/28	142

	city	favourite_genres	id	name	premium_user	street	user_id
42	Warszawa	[classic rock, new romantic, latin alternative]	NaN	Julita Kuliberda	1.0	plac Hutnicza 22	143
43	Poznań	[new romantic, rock, modern rock]	NaN	Liwia Chylak	1.0	pl. Bolesława Chrobrego 047	144
44	Wrocław	[permanent wave, pop rock, hoerspiel]	-1.0	Adrianna Golak	1.0	plac Krucza 84/60	145
45	Wrocław	[tropical, latin alternative, tropical]	-1.0	Jacek Nałęcz	1.0	ulica Prusa 95	146
46	Kraków	[mpb, rock, latin]	NaN	Nicole Batóg	1.0	plac Jarzębinowa 87/73	147
47	Warszawa	[psychedelic rock, mandopop, vocal jazz]	NaN	Kalina Kuster	1.0	al. Okrzei 69	148
48	Warszawa	[adult standards, alternative metal, album rock]	-1.0	Nikodem Bródka	1.0	plac Storczykowa 23	149
49	Warszawa	[rock, pop rock, new wave]	NaN	Patryk Jarmuła	1.0	aleja Słoneczna 47/98	150

In [ ]:

```
for view, data_frame in DATA_FRAMES:
    show_table_name(view)
    for column, type in data_frame.dtypes:
        show_column_name(column)
        group_by_column = f"""--sql
            SELECT
                {column},
                COUNT(*) AS length
            FROM {view}
            GROUP BY {column}
            ORDER BY {column} IS NULL DESC, length DESC, {column} NULLS FIRST
        """
        df = spark.sql(group_by_column)
        display(df.limit(100_000).toPandas())

        count_distinct = f"""--sql
            SELECT
                COUNT(DISTINCT {column})
            FROM {view}
        """
        df = spark.sql(count_distinct)
        display(df.toPandas())
```

```
================================================================================
                                     ARTISTS
================================================================================
GENRES
```

	genres	length
0	None	1352
1	[indonesian pop]	74
2	[classic thai pop]	68
3	[thai pop]	60
4	[classic turkish pop]	57
...	...	...
13066	[yiddish folk]	1
13067	[yoga]	1
13068	[yugoslav new wave]	1
13069	[zhongguo feng]	1
13070	[zolo]	1

13071 rows × 2 columns

	count(DISTINCT genres)
0	13070

```
ID
```

	id	length
0	-1	1371
1	0001wHqxbF2YYRQxGdbyER	1
2	000p4jMMhpEHq1h6PFCyO1	1
3	001aJOc7CSQVo3XzoLG4DK	1
4	0027wHZDQXpRll4ckwDGad	1
...	...	...
26149	7zup4xIPjtv50lM7x3n4qW	1
26150	7zw8gWmNncuk2QZHIc70So	1
26151	7zwF847GE2hY5ApGSOLmBG	1
26152	7zwiFdY90oXzLh1Wz22oEq	1
26153	7zzsdcNemyhcNk2wpNsXZt	1

26154 rows × 2 columns

	count(DISTINCT id)
0	26154

```
NAME
```

	name	length
0	TNT	4
1	Kali	3
2	Sebastian	3
3	Akcent	2
4	Alice	2
...	...	...
27411	黃韻玲	1
27412	黑豹	1
27413	龍飄飄	1
27414	龔秋霞	1
27415	龔詩嘉	1

27416 rows × 2 columns

	count(DISTINCT name)
0	27416

```
================================================================================
                                     SESSIONS
================================================================================
EVENT_TYPE
```

	event_type	length
0	None	181
1	PLAY	2157
2	SKIP	795
3	LIKE	595
4	BUY_PREMIUM	49
5	ADVERTISEMENT	28

	count(DISTINCT event_type)
0	5

```
SESSION_ID
```

	session_id	length
0	304	37
1	302	31
2	326	29
3	679	29
4	721	29
...	...	...
617	773	1
618	782	1
619	786	1
620	791	1
621	792	1

622 rows × 2 columns

	count(DISTINCT session_id)
0	622

```
TIMESTAMP
```

	timestamp	length
0	2022-03-28T18:05:22.260000	2
1	2022-03-28T18:06:58.255000	2
2	2022-03-28T18:26:22.362000	2
3	2022-03-29T15:45:55.903000	2
4	2022-03-31T03:28:25.326000	2
...	...	...
2972	2023-03-28T07:17:20.214000	1
2973	2023-03-28T07:18:17.652000	1
2974	2023-03-28T07:19:51.547000	1
2975	2023-03-28T07:27:49.972000	1
2976	2023-03-28T07:33:21.857000	1

2977 rows × 2 columns

	count(DISTINCT timestamp)
0	2977

```
TRACK_ID
```

	track_id	length
0	None	200
1		76
2	18mSX3KXGDCkrHDT5gmZTY	5
3	25iHbm3dv9BYhW7sQWbMg9	5
4	4qCYYhzI5bCz7JxV7VD4HH	5
...	...	...
2185	7y5x64GInqHxe7e2LXOfay	1
2186	7yVrrN3wUi6xKOsMjddCic	1
2187	7ycMIXNZZoMgtsZAGK5QEw	1
2188	7z9Ez0NdQESqdAjpvdpIcW	1
2189	7zvwxa2s4zIX7y49plhrmo	1

2190 rows × 2 columns

	count(DISTINCT track_id)
0	2189

```
USER_ID
```

	user_id	length
0	NaN	183
1	147.0	151
2	106.0	141
3	114.0	138
4	120.0	125
5	149.0	124
6	141.0	118
7	143.0	113
8	119.0	112
9	130.0	103
10	137.0	97
11	110.0	96
12	148.0	95
13	139.0	93
14	133.0	91
15	136.0	88
16	105.0	86
17	108.0	86
18	144.0	80
19	138.0	79
20	124.0	76
21	134.0	74
22	101.0	73
23	132.0	70
24	117.0	69
25	103.0	65
26	145.0	65
27	109.0	63
28	135.0	63
29	125.0	62
30	116.0	61
31	118.0	61
32	140.0	61
33	107.0	60
34	121.0	57
35	122.0	54
36	129.0	53
37	113.0	52
38	112.0	51
39	102.0	49
40	111.0	49
41	104.0	47

	user_id	length
42	115.0	43
43	131.0	40
44	123.0	35
45	146.0	35
46	142.0	32
47	126.0	27
48	150.0	25
49	127.0	19
50	128.0	15

	count(DISTINCT user_id)
0	50

```
TIMESTAMP_S
```

	timestamp_s	length
0	1661889343	4
1	1650950094	3
2	1663398418	3
3	1668219039	3
4	1675084191	3
...	...	...
2966	1679980640	1
2967	1679980697	1
2968	1679980791	1
2969	1679981269	1
2970	1679981601	1

2971 rows × 2 columns

	count(DISTINCT timestamp_s)
0	2971

```
================================================================================
                                  TRACK_STORAGE
================================================================================
DAILY_COST
```

	daily_cost	length
0	0.009600	44
1	0.011700	41
2	0.008000	39
3	0.010000	39
4	0.010800	38
...	...	...
47433	0.229282	1
47434	0.236263	1
47435	0.239629	1
47436	0.239863	1
47437	0.249754	1

47438 rows × 2 columns

	count(DISTINCT daily_cost)
0	47438

```
STORAGE_CLASS
```

	storage_class	length
0	SLOW	128369
1	MEDIUM	1275
2	FAST	4

	count(DISTINCT storage_class)
0	3

```
TRACK_ID
```

	track_id	length
0	000jBcNljWTnyjB4YO7ojf	1
1	000u1dTg7y1XCDXi80hbBX	1
2	0017A6SJgTbfQVU2EtsPNo	1
3	001UI3J6PKAEnBgqrwGGQC	1
4	001gx41rQo0bKh063TrC1I	1
...	...	...
99995	5ye1yhnGkhvf4G5yDIP6fq	1
99996	5yeBQ7lI2Qi9Ez0ZBDCYgT	1
99997	5yeCt0MReP9i652S9l1fOa	1
99998	5yeXw1L7CqKXkHaJ0W4RrT	1
99999	5yeoAPpSg8eD4MRRojxtpY	1

100000 rows × 2 columns

	count(DISTINCT track_id)
0	129648

```
================================================================================
                                     TRACKS
================================================================================
ACOUSTICNESS
```

	acousticness	length
0	0.99500	525
1	0.99400	426
2	0.99300	355
3	0.99200	317
4	0.99100	312
...	...	...
4535	0.00853	1
4536	0.00868	1
4537	0.00926	1
4538	0.00960	1
4539	0.00986	1

4540 rows × 2 columns

	count(DISTINCT acousticness)
0	4540

```
DANCEABILITY
```

	danceability	length
0	0.629	359
1	0.565	350
2	0.549	348
3	0.652	348
4	0.611	345
...	...	...
1023	0.980	1
1024	0.982	1
1025	0.984	1
1026	0.985	1
1027	0.988	1

1028 rows × 2 columns

	count(DISTINCT danceability)
0	1028

```
DURATION_MS
```

	duration_ms	length
0	192000	44
1	234000	41
2	160000	39
3	200000	39
4	224000	39
...	...	...
46735	4585640	1
46736	4725264	1
46737	4792587	1
46738	4797258	1
46739	4995083	1

46740 rows × 2 columns

	count(DISTINCT duration_ms)
0	46740

```
ENERGY
```

	energy	length
0	0.5380	230
1	0.4990	227
2	0.6340	217
3	0.4840	212
4	0.7160	211
...	...	...
1873	0.0920	1
1874	0.0957	1
1875	0.0960	1
1876	0.0987	1
1877	0.0996	1

1878 rows × 2 columns

	count(DISTINCT energy)
0	1878

```
EXPLICIT
```

	explicit	length
0	0	124929
1	1	4719

	count(DISTINCT explicit)
0	2

```
ID
```

	id	length
0	None	6530
1	000jBcNljWTnyjB4YO7ojf	1
2	000u1dTg7y1XCDXi80hbBX	1
3	0017A6SJgTbfQVU2EtsPNo	1
4	001UI3J6PKAEnBgqrwGGQC	1
...	...	...
99995	6IUhPMJf4iJQ3Go1CkHDsa	1
99996	6IUiqtI8tE49sqGbmtrNd8	1
99997	6IUjR7tioorwwRP3d9tSJa	1
99998	6IUoqFptVXfEO22DvxECDF	1
99999	6IV6vOdfb5Jhxctp9IO6iw	1

100000 rows × 2 columns

	count(DISTINCT id)
0	123118

```
ID_ARTIST
```

	id_artist	length
0	None	6504
1	3meJIgRw7YleJrmbpbJK6S	1057
2	0i38tQX5j4gZ0KS3eCMoIl	549
3	1l6d0RIxTL3JytlLGvWzYe	446
4	3t2iKODSDyzoDJw7AsD99u	437
...	...	...
26861	7zjX652bWyemXyFFVhBnch	1
26862	7zlWN2A8mV2thjdvAyMrEJ	1
26863	7zmk5lkmCMVvfvwF3H8FWC	1
26864	7zpw4vmlZNCUlwbdnFwxwO	1
26865	7zw8gWmNncuk2QZHIc70So	1

26866 rows × 2 columns

	count(DISTINCT id_artist)
0	26865

```
INSTRUMENTALNESS
```

	instrumentalness	length
0	0.000000	46190
1	0.000010	83
2	0.897000	74
3	0.000012	73
4	0.000104	72
...	...	...
5392	0.099100	1
5393	0.099900	1
5394	0.993000	1
5395	0.994000	1
5396	0.995000	1

5397 rows × 2 columns

	count(DISTINCT instrumentalness)
0	5397

```
KEY
```

	key	length
0	0	16686
1	7	16466
2	9	15219
3	2	15118
4	5	11655
5	4	11090
6	11	8781
7	1	8522
8	10	7921
9	8	7182
10	6	6607
11	3	4401

	count(DISTINCT key)
0	12

```
LIVENESS
```

	liveness	length
0	0.1110	1209
1	0.1080	1178
2	0.1100	1164
3	0.1070	1116
4	0.1090	1113
...	...	...
1735	0.0239	1
1736	0.0250	1
1737	0.0262	1
1738	0.0284	1
1739	0.9990	1

1740 rows × 2 columns

	count(DISTINCT liveness)
0	1740

```
LOUDNESS
```

	loudness	length
0	-8.026	36
1	-5.797	32
2	-7.679	28
3	-7.338	26
4	-12.502	25
...	...	...
20356	2.534	1
20357	2.639	1
20358	2.695	1
20359	3.273	1
20360	4.362	1

20361 rows × 2 columns

	count(DISTINCT loudness)
0	20361

```
NAME
```

	name	length
0	None	6547
1	Hold On	40
2	Home	21
3	Summertime	21
4	99 Year Blues	19
...	...	...
99995	Танцы	1
99996	Твое сердце должно быть моим	1
99997	Твои глаза	1
99998	Твой	1
99999	Твой папа был прав	1

100000 rows × 2 columns

	count(DISTINCT name)
0	108892

```
POPULARITY
```

	popularity	length
0	NaN	6469
1	0.0	4255
2	35.0	2919
3	36.0	2859
4	23.0	2839
...	...	...
91	89.0	2
92	91.0	1
93	92.0	1
94	97.0	1
95	99.0	1

96 rows × 2 columns

	count(DISTINCT popularity)
0	95

```
RELEASE_DATE
```

	release_date	length
0	1998-01-01	750
1	1997-01-01	738
2	1998	720
3	1995	718
4	1996	692
...	...	...
14936	2021-03-23	1
14937	2021-03-27	1
14938	2021-03-28	1
14939	2021-04-03	1
14940	2021-04-04	1

14941 rows × 2 columns

	count(DISTINCT release_date)
0	14941

```
SPEECHINESS
```

	speechiness	length
0	0.0315	531
1	0.0312	514
2	0.0310	510
3	0.0308	502
4	0.0309	501
...	...	...
1632	0.8040	1
1633	0.8240	1
1634	0.8470	1
1635	0.9680	1
1636	0.9690	1

1637 rows × 2 columns

	count(DISTINCT speechiness)
0	1637

```
TEMPO
```

	tempo	length
0	0.000	48
1	139.980	29
2	119.996	22
3	127.997	22
4	130.022	22
...	...	...
70580	233.013	1
70581	236.134	1
70582	238.895	1
70583	239.906	1
70584	243.507	1

70585 rows × 2 columns

	count(DISTINCT tempo)
0	70585

```
VALENCE
```

	valence	length
0	0.9610	614
1	0.9620	536
2	0.9630	469
3	0.9640	445
4	0.9600	387
...	...	...
1623	0.0888	1
1624	0.0891	1
1625	0.0919	1
1626	0.0939	1
1627	0.0979	1

1628 rows × 2 columns

	count(DISTINCT valence)
0	1628

```
RELEASE_DATE_S
```

	release_date_s	length
0	883609200	1470
1	852073200	1418
2	820450800	1351
3	788914800	1349
4	631148400	1288
...	...	...
14678	1616454000	1
14679	1616799600	1
14680	1616886000	1
14681	1617400800	1
14682	1617487200	1

14683 rows × 2 columns

	count(DISTINCT release_date_s)
0	14683

```
================================================================================
                                     USERS
================================================================================
CITY
```

	city	length
0	Warszawa	13
1	Kraków	8
2	Radom	8
3	Gdynia	7
4	Poznań	6
5	Wrocław	6
6	Szczecin	2

	count(DISTINCT city)
0	7

```
FAVOURITE_GENRES
```

	favourite_genres	length
0	None	1
1	[adult standards, alternative metal, album rock]	1
2	[adult standards, mpb, funk]	1
3	[alternative rock, adult standards, pop]	1
4	[alternative rock, alternative metal, vocal jazz]	1
5	[alternative rock, permanent wave, latin pop]	1
6	[blues rock, lounge, post-teen pop]	1
7	[c-pop, motown, tropical]	1
8	[classic rock, new romantic, latin alternative]	1
9	[classic rock, pop rock, soft rock]	1
10	[europop, folk, tropical]	1
11	[funk, classic rock, europop]	1
12	[italian adult pop, funk, italian adult pop]	1
13	[italian adult pop, lounge, folk rock]	1
14	[j-pop, folk rock, metal]	1
15	[latin rock, rock, folk rock]	1
16	[lounge, hoerspiel, album rock]	1
17	[mellow gold, c-pop, argentine rock]	1
18	[metal, new wave, argentine rock]	1
19	[modern rock, adult standards, pop rock]	1
20	[modern rock, tropical, adult standards]	1
21	[motown, regional mexican, folk]	1
22	[motown, soul, regional mexican]	1
23	[motown, vocal jazz, mandopop]	1
24	[mpb, permanent wave, hoerspiel]	1
25	[mpb, rock, latin]	1
26	[new romantic, art rock, new wave]	1
27	[new romantic, country rock, brill building pop]	1
28	[new romantic, rock, modern rock]	1
29	[new wave, psychedelic rock, soft rock]	1
30	[permanent wave, pop rock, hoerspiel]	1
31	[permanent wave, post-teen pop, mandopop]	1
32	[pop, new wave pop, motown]	1
33	[pop rock, europop, hoerspiel]	1
34	[post-teen pop, psychedelic rock, latin altern...	1
35	[psychedelic rock, mandopop, vocal jazz]	1
36	[ranchera, new romantic, adult standards]	1
37	[regional mexican, mellow gold, folk rock]	1
38	[regional mexican, psychedelic rock, new roman...	1
39	[regional mexican, rock en espanol, argentine ...	1
40	[rock, lounge, metal]	1
41	[rock, pop rock, new wave]	1

	favourite_genres	length
42	[rock en espanol, new wave pop, italian adult ...	1
43	[rock en espanol, rock, latin]	1
44	[roots rock, latin pop, alternative metal]	1
45	[roots rock, modern rock, j-pop]	1
46	[soul, lounge, pop rock]	1
47	[soul, mellow gold, blues rock]	1
48	[tropical, latin alternative, tropical]	1
49	[vocal jazz, pop rock, soul]	1

	count(DISTINCT favourite_genres)
0	49

```
ID
```

	id	length
0	NaN	45
1	-1.0	5

	count(DISTINCT id)
0	1

```
NAME
```

	name	length
0	Adrianna Golak	1
1	Albert Brzeźniak	1
2	Andrzej Doktor	1
3	Anita Pioch	1
4	Anna Maria Ignatiuk	1
5	Arkadiusz Krzywoń	1
6	Bartek Garczyk	1
7	Blanka Szklarek	1
8	Borys Matula	1
9	Cezary Getka	1
10	Dawid Koperek	1
11	Eryk Kołata	1
12	Filip Kalinka	1
13	Filip Łukowiak	1
14	Fryderyk Chabior	1
15	Gustaw Pilipczuk	1
16	Ignacy Pniak	1
17	Jacek Nałęcz	1
18	Jan Gryga	1
19	Janina Delekta	1
20	Jerzy Husak	1
21	Julianna Więckiewicz	1
22	Julita Kuliberda	1
23	Kacper Osojca	1
24	Kalina Kuster	1
25	Kazimierz Stypa	1
26	Kornel Dacko	1
27	Krzysztof Wojtach	1
28	Krzysztof Żuchowicz	1
29	Ksawery Klus	1
30	Liwia Chylak	1
31	Maciej Bandyk	1
32	Marika Pilipczuk	1
33	Maurycy Hutyra	1
34	Maurycy Szoka	1
35	Melania Gałat	1
36	Monika Sypień	1
37	Nicole Batóg	1
38	Nicole Gajdzik	1
39	Nikodem Bródka	1
40	Nikodem Kopciuch	1
41	Nikodem Wawrzynowicz	1

	name	length
42	Oliwier Smalec	1
43	Oskar Jarosik	1
44	Patryk Jarmuła	1
45	Radosław Musiolik	1
46	Sebastian Molka	1
47	Stefan Bisaga	1
48	Sylwia Feret	1
49	Łukasz Pielka	1

	count(DISTINCT name)
0	50

```
PREMIUM_USER
```

	premium_user	length
0	NaN	2
1	1.0	48

	count(DISTINCT premium_user)
0	1

```
STREET
```

	street	length
0	al. Armii Krajowej 564	1
1	al. Jesionowa 47	1
2	al. Konwaliowa 33	1
3	al. Okrzei 69	1
4	al. Podleśna 00	1
5	al. Szeroka 27/38	1
6	al. Tęczowa 332	1
7	aleja Bema 889	1
8	aleja Prusa 830	1
9	aleja Stolarska 554	1
10	aleja Słoneczna 47/98	1
11	aleja Tartaczna 95	1
12	aleja Urocza 19	1
13	aleja Zaułek 750	1
14	pl. Bolesława Chrobrego 047	1
15	pl. Daszyńskiego 80/41	1
16	pl. Jagiellońska 607	1
17	pl. Jana 300	1
18	pl. Mazurska 345	1
19	pl. Orzeszkowej 21	1
20	pl. Promienna 59/43	1
21	pl. Radosna 86/89	1
22	pl. Staszica 343	1
23	pl. Słonecznikowa 79	1
24	plac Floriana 59/72	1
25	plac Hutnicza 22	1
26	plac Jarzębinowa 87/73	1
27	plac Kazimierza Wielkiego 51	1
28	plac Krucza 84/60	1
29	plac Sadowa 527	1
30	plac Składowa 526	1
31	plac Storczykowa 23	1
32	plac Słowicza 73	1
33	plac Wyspiańskiego 73/43	1
34	ul. Brzoskwiniowa 81	1
35	ul. Ciasna 73	1
36	ul. Diamentowa 44	1
37	ul. Konopnickiej 038	1
38	ul. Księżycowa 31	1
39	ul. Rybacka 07	1
40	ulica Długosza 71/06	1
41	ulica Irysowa 483	1

	street	length
42	ulica Konwaliowa 74	1
43	ulica Księżycowa 11	1
44	ulica Mokra 71	1
45	ulica Prusa 95	1
46	ulica Szpitalna 18	1
47	ulica Torowa 80	1
48	ulica Witosa 98/28	1
49	ulica Wiązowa 07/54	1

	count(DISTINCT street)
0	50

```
USER_ID
```

	user_id	length
0	101	1
1	102	1
2	103	1
3	104	1
4	105	1
5	106	1
6	107	1
7	108	1
8	109	1
9	110	1
10	111	1
11	112	1
12	113	1
13	114	1
14	115	1
15	116	1
16	117	1
17	118	1
18	119	1
19	120	1
20	121	1
21	122	1
22	123	1
23	124	1
24	125	1
25	126	1
26	127	1
27	128	1
28	129	1
29	130	1
30	131	1
31	132	1
32	133	1
33	134	1
34	135	1
35	136	1
36	137	1
37	138	1
38	139	1
39	140	1
40	141	1
41	142	1

	user_id	length
42	143	1
43	144	1
44	145	1
45	146	1
46	147	1
47	148	1
48	149	1
49	150	1

	count(DISTINCT user_id)
0	50

In [ ]:

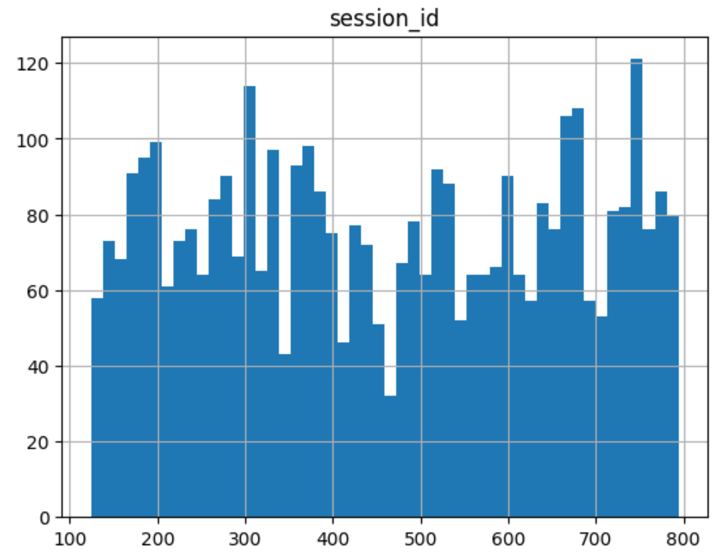
```python
def aggregate_numeric_column(view: str, column: str) -> str:
    return f"""--sql
            SELECT
                "{column}" AS name,
                COUNT({column}) AS count,
                MIN({column}) AS min,
                MAX({column}) AS max,
                AVG({column}) AS average,
                SUM({column}) AS sum,
                SUM(DISTINCT {column}) AS sum_distinct,
                KURTOSIS({column}) AS kurtosis,
                SKEWNESS({column}) AS skewness,
                STDDEV({column}) AS standard_deviation,
                STDDEV_POP({column}) AS population_standard_deviation,
                VARIANCE({column}) AS variance,
                VAR_POP({column}) AS population_variance
            FROM {view}
            WHERE {column} IS NOT NULL
        """

for view, data_frame in DATA_FRAMES:
    show_table_name(view)
    for column, type in data_frame.dtypes:
        if type in ['double', 'bigint']:
            show_column_name(column)
            df = spark.sql(aggregate_numeric_column(view, column))
            display(df.toPandas())

            dfp = spark.sql(f"SELECT {column} FROM {view}").toPandas()
            dfp.hist(bins=50)
            plt.show()
```

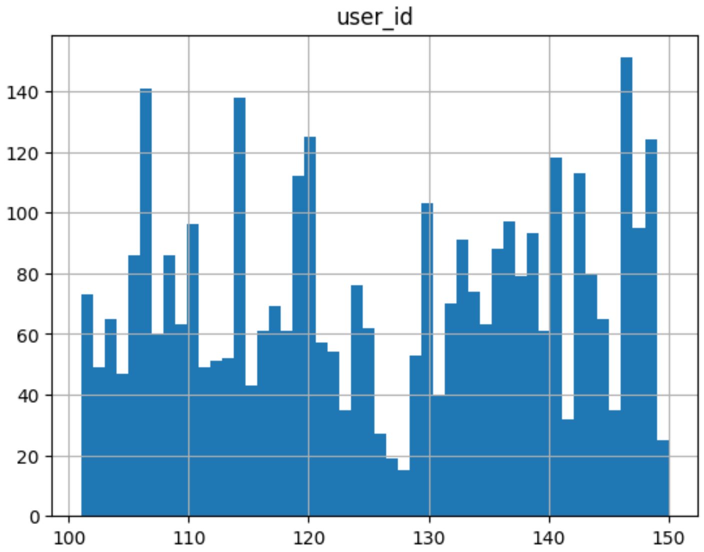
```
================================================================================
                                     ARTISTS
================================================================================
================================================================================
                                     SESSIONS
================================================================================
SESSION_ID
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	session_id	3805	124	794	461.138765	1754633	285854	-1.280354	0.018784	197.815899	197.789903	39131.130056	39120.845922



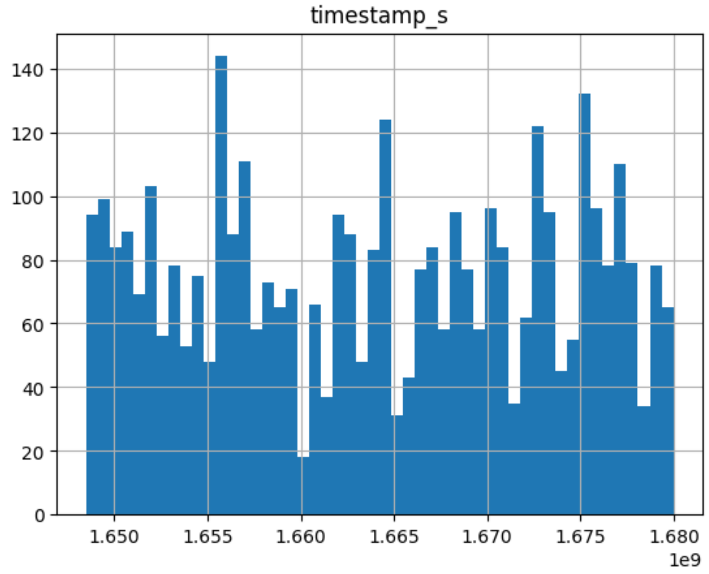
```
USER_ID
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	user_id	3622	101	150	126.139978	456879	6275	-1.331799	-0.042814	14.900369	14.898312	222.020997	221.959699



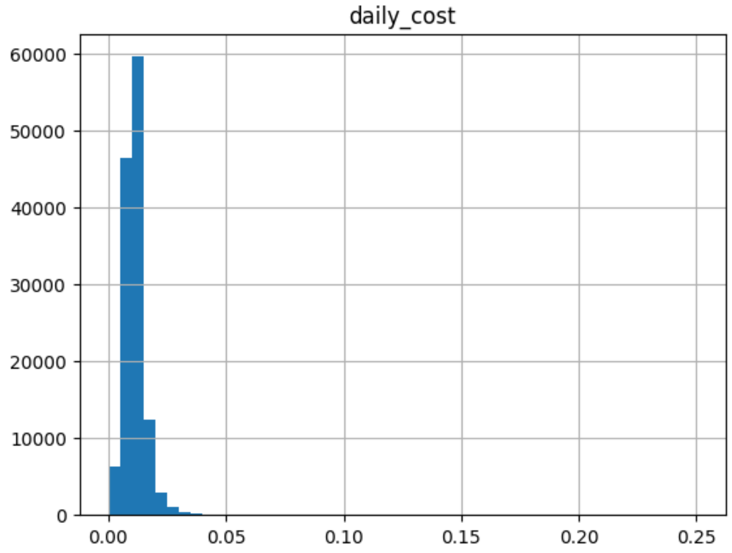
```
TIMESTAMP_S
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	timestamp_s	3805	1648483145	1679981601	1.664111e+09	6331942400811	4943637845814	-1.279597	-0.021085	9.327247e+06	9.326021e+06	8.699753e+13	8.697467e+13



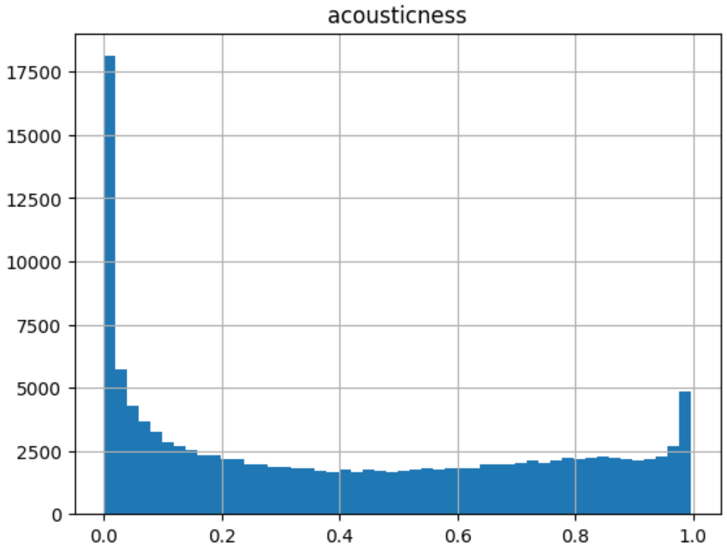
```
================================================================================
                                  TRACK_STORAGE
================================================================================
DAILY_COST
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	daily_cost	129648	0.000167	0.249754	0.011535	1495.508148	591.933795	259.234276	10.35695	0.005815	0.005815	0.000034	0.000034



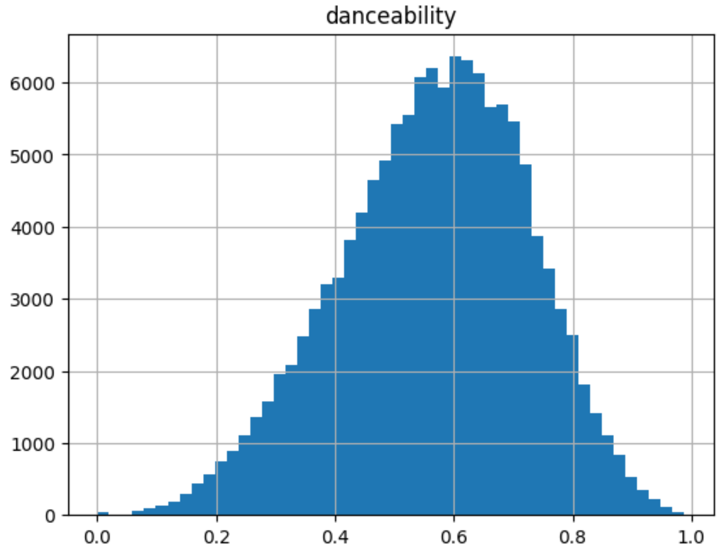
```
================================================================================
                                      TRACKS
================================================================================
ACOUSTICNESS
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	acousticness	129648	0.0	0.996	0.41755	54134.576468	546.440307	-1.383039	0.250805	0.335652	0.335651	0.112662	0.112661



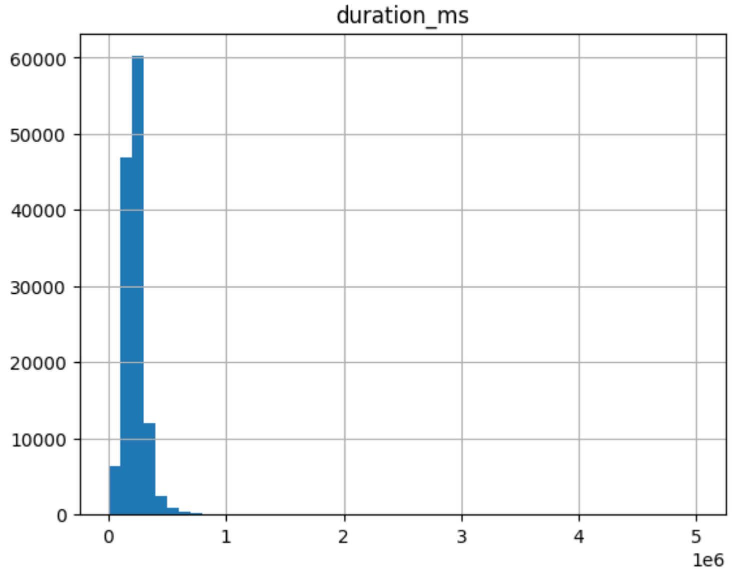
```
DANCEABILITY
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	danceability	129648	0.0	0.988	0.564894	73237.4093	491.2168	-0.258259	-0.28432	0.159114	0.159113	0.025317	0.025317



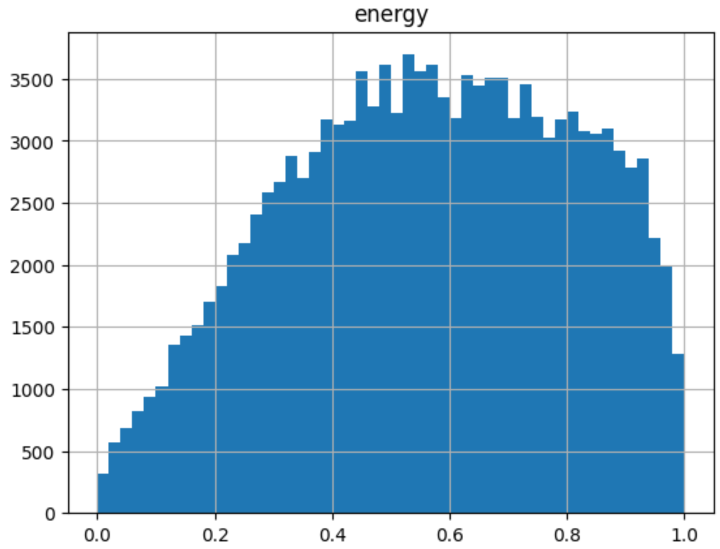
```
DURATION_MS
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	duration_ms	129648	3344	4995083	228526.632274	29628020821	11430854470	281.491889	10.884919	113801.507474	113801.068587	1.295078e+10	1.295068e+10



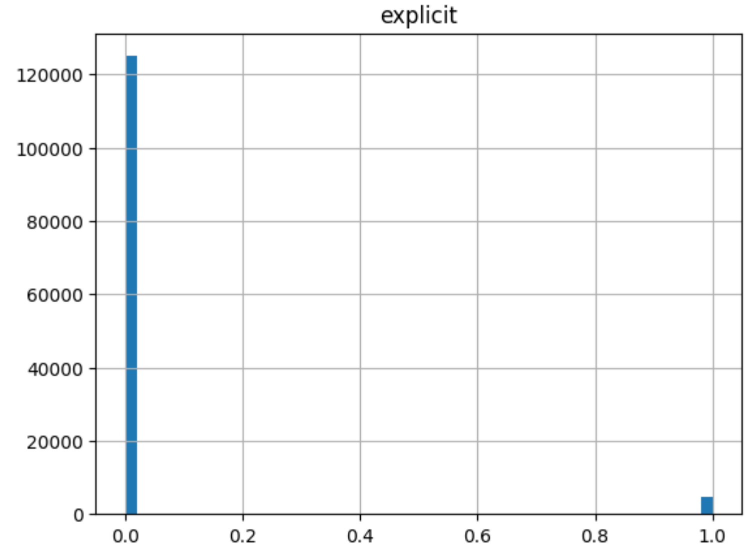
ENERGY

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	energy	129648	0.0	1.0	0.562776	72962.72439	543.752618	-0.899073	-0.168391	0.241957	0.241956	0.058543	0.058543



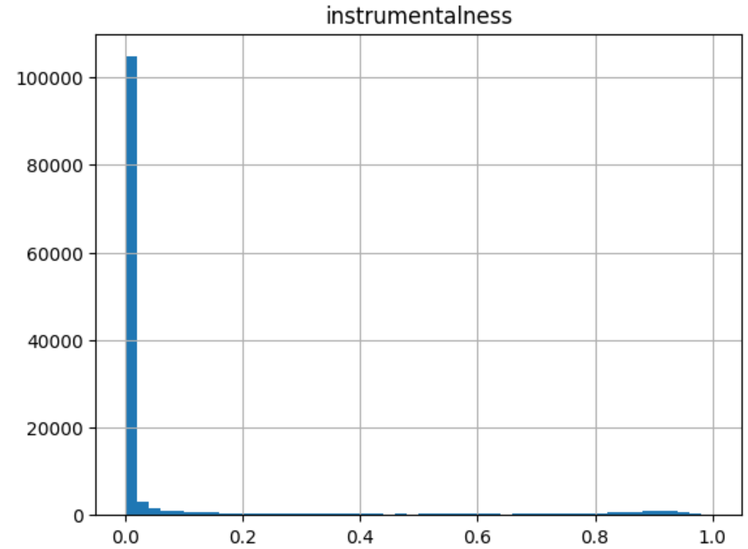
EXPLICIT

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	explicit	129648	0	1	0.036399	4719	1	22.511391	4.950898	0.18728	0.18728	0.035074	0.035074



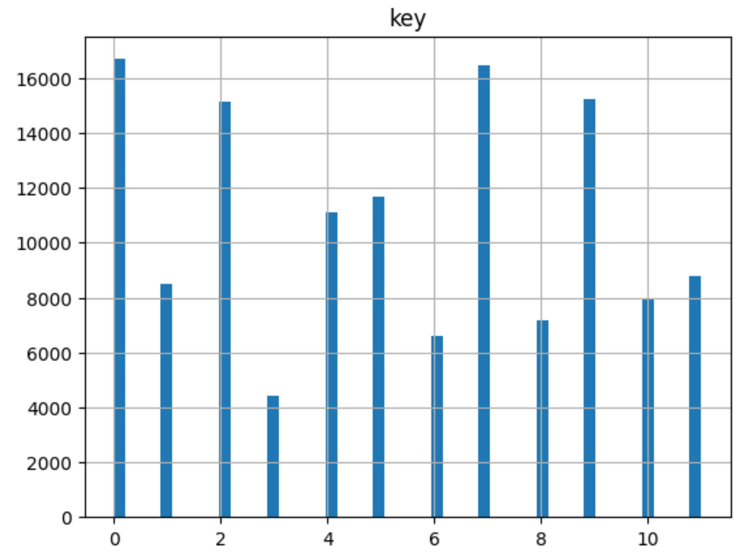
INSTRUMENTALNESS

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	instrumentalness	129648	0.0	1.0	0.086754	11247.463381	549.236231	6.200105	2.759591	0.232285	0.232284	0.053956	0.053956



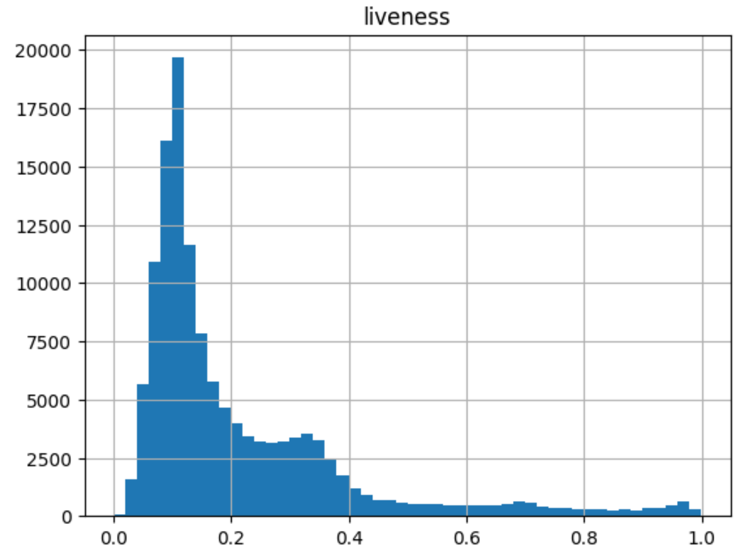
KEY

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	key	129648	0	11	5.242873	679728	66	-1.265013	-0.011349	3.518889	3.518876	12.382581	12.382485



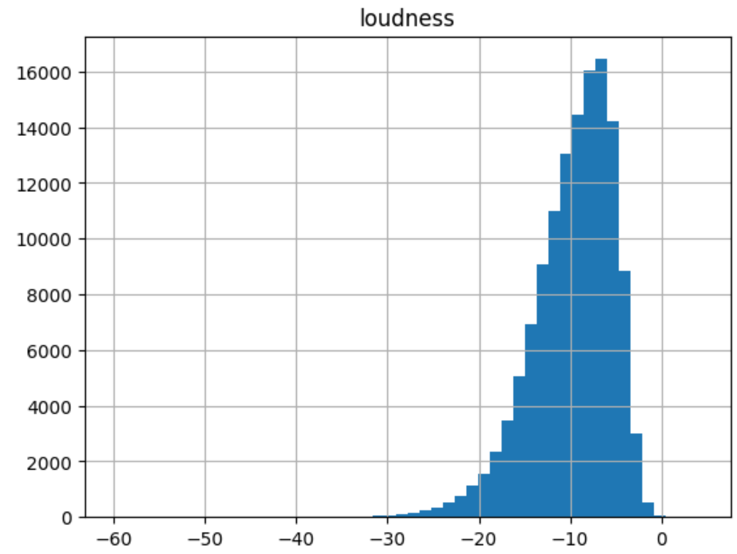
LIVENESS

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	liveness	129648	0.0	0.999	0.21406	27752.50933	543.09323	4.380976	2.072202	0.186901	0.1869	0.034932	0.034932



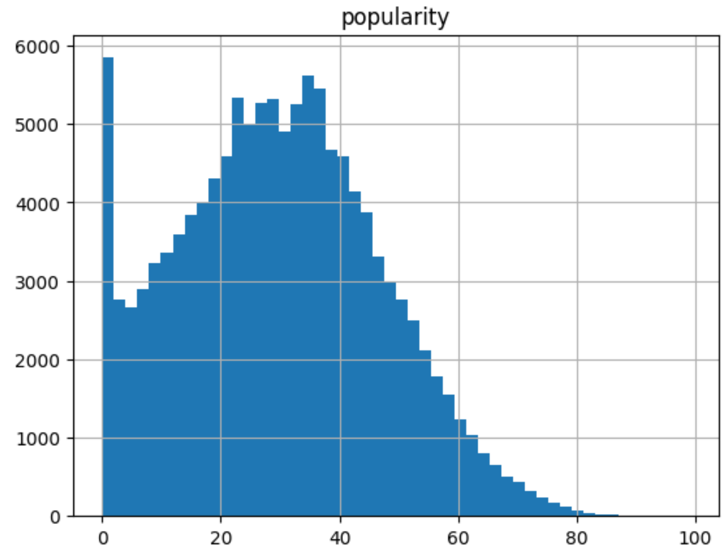
LOUDNESS

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	loudness	129648	-60.0	4.362	-9.734177	-1262016.64	-252312.279	2.778514	-1.104693	4.5213	4.521283	20.442158	20.442



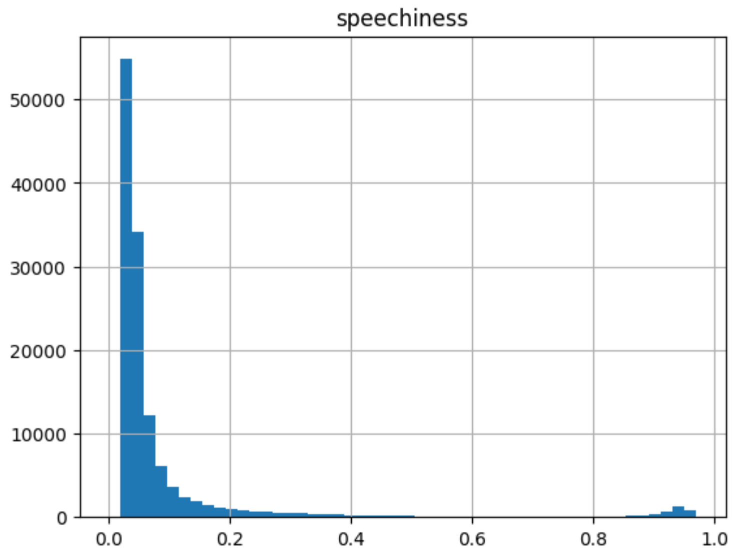
POPULARITY

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	popularity	123179	0	99	29.677981	3655704	4474	-0.481352	0.22448	17.129474	17.129405	293.418896	293.416514



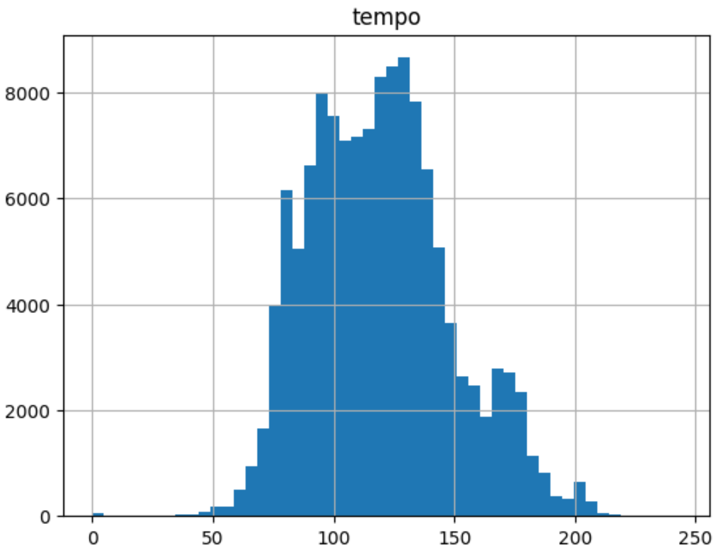
SPEECHINESS

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	speechiness	129648	0.0	0.969	0.095068	12325.3914	503.1898	16.456687	4.045176	0.166167	0.166166	0.027611	0.027611



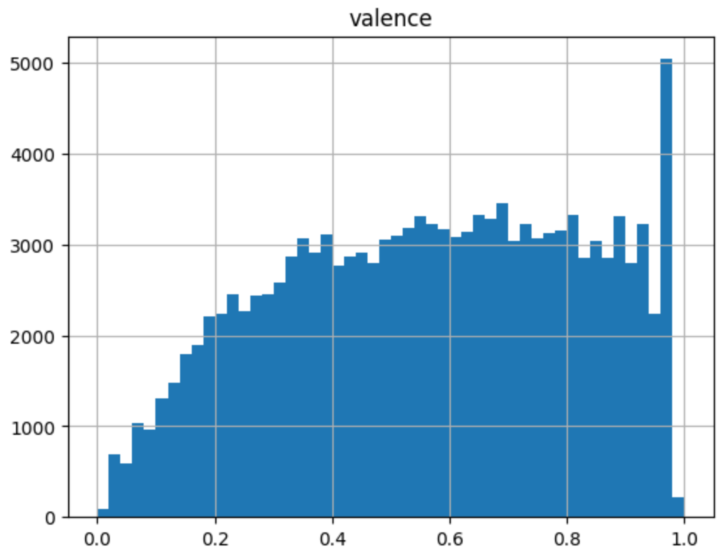
TEMPO

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	tempo	129648	0.0	243.507	119.53864	1.549795e+07	8607442.191	-0.106043	0.402869	29.653393	29.653278	879.323707	879.316925



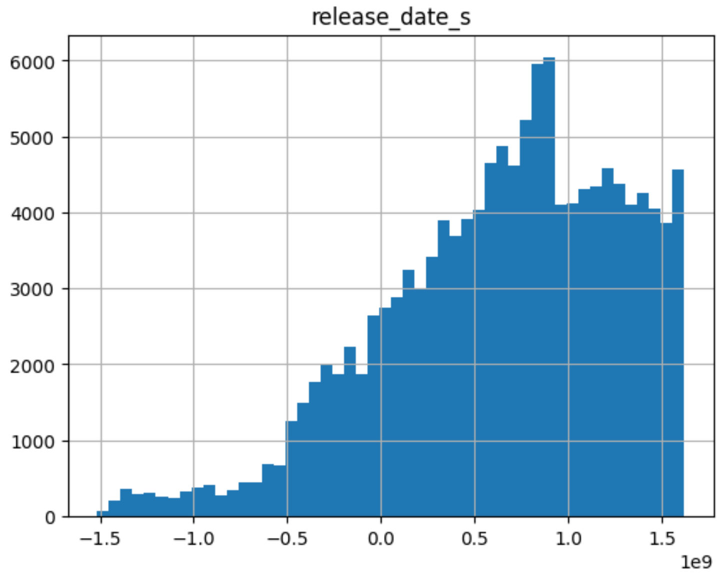
VALENCE

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	valence	129648	0.0	1.0	0.563443	73049.2694	537.05768	-1.035815	-0.154964	0.252581	0.25258	0.063797	0.063796



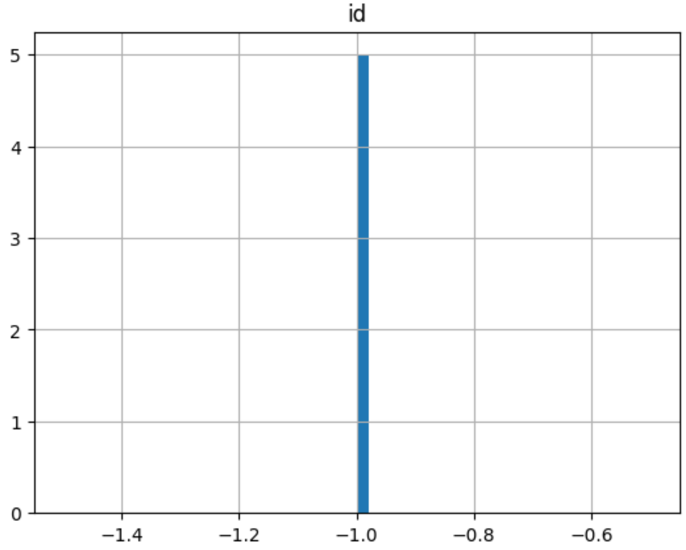
```
RELEASE_DATE_S
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	release_date_s	129648	-1514772000	1618524000	6.407151e+08	83067436238400	10982866910400	0.075787	-0.656014	6.358551e+08	6.358526e+08	4.043117e+17	4.043086e+17



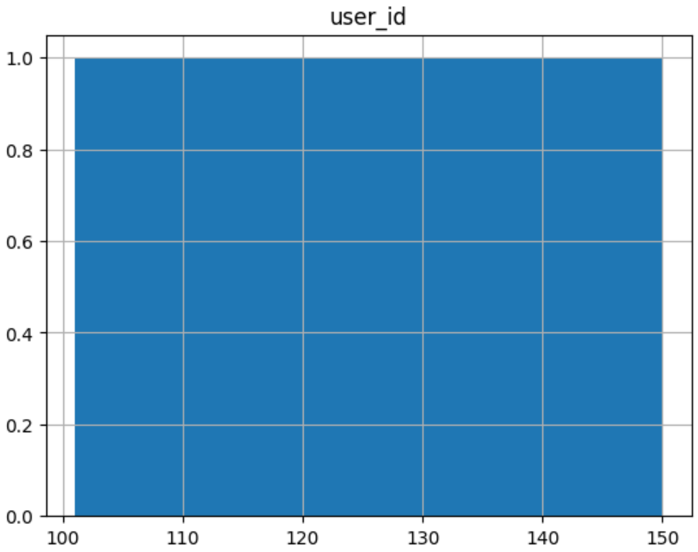
```
==============================================================================
                                    USERS
==============================================================================
ID
```

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	id	5	-1	-1	-1.0	-5	-1	NaN	NaN	0.0	0.0	0.0	0.0



USER_ID

	name	count	min	max	average	sum	sum_distinct	kurtosis	skewness	standard_deviation	population_standard_deviation	variance	population_variance
0	user_id	50	101	150	125.5	6275	6275	-1.20096	2.542155e-16	14.57738	14.43087	212.5	208.25



In [ ]:
```
def explode_column(view: str, column: str) -> str:
    return f"""--sql
            SELECT
                DISTINCT EXPLODE({column}) AS distinct_{column}
            FROM {view}
            ORDER BY distinct_{column} NULLS FIRST
        """


def count_exploded_column(view: str, column: str) -> str:
    exploded = f"""--sql
        SELECT
            DISTINCT EXPLODE({column}) AS {column}
        FROM {view}
    """

    return f"""--sql
            SELECT
                COUNT(*) AS length
            FROM ({exploded})
        """

for view, data_frame in DATA_FRAMES:
    show_table_name(view)
    for column, type in data_frame.dtypes:
        if type.startswith('array'):
            show_column_name(column)
            df = spark.sql(explode_column(view, column))
            display(df.toPandas())
            df = spark.sql(count_exploded_column(view, column))
            display(df.toPandas())
```

```
================================================================================
                                     ARTISTS
================================================================================
GENRES
```

	distinct_genres
0	48g
1	a cappella
2	abstract
3	abstract hip hop
4	accordeon
...	...
3867	zolo
3868	zouglou
3869	zouk
3870	zouk riddim
3871	zydeco

3872 rows × 1 columns

	length
0	3872

```
================================================================================
                                     SESSIONS
================================================================================
================================================================================
                                  TRACK_STORAGE
================================================================================
================================================================================
                                      TRACKS
================================================================================
================================================================================
                                      USERS
================================================================================
FAVOURITE_GENRES
```

	distinct_favourite_genres
0	adult standards
1	album rock
2	alternative metal
3	alternative rock
4	argentine rock
5	art rock
6	blues rock
7	brill building pop
8	c-pop
9	classic rock
10	country rock
11	europop
12	folk
13	folk rock
14	funk
15	hoerspiel
16	italian adult pop
17	j-pop
18	latin
19	latin alternative
20	latin pop
21	latin rock
22	lounge
23	mandopop
24	mellow gold
25	metal
26	modern rock
27	motown
28	mpb
29	new romantic
30	new wave
31	new wave pop
32	permanent wave
33	pop
34	pop rock
35	post-teen pop
36	psychedelic rock
37	ranchera
38	regional mexican
39	rock
40	rock en espanol
41	roots rock

	distinct_favourite_genres
42	soft rock
43	soul
44	tropical
45	vocal jazz

	length
0	46

In [ ]:
```
JOINS = {
    ('artists', 'tracks') : ('id', 'id_artist'),
    ('tracks', 'track_storage') : ('id', 'track_id'),
    ('tracks', 'sessions') : ('id', 'track_id'),
    ('users', 'sessions') : ('user_id', 'user_id'),
}
```

In [ ]:
```
def count_everything(table: str) -> str:
    return f"""--sql
        SELECT
            COUNT(*) AS length_{table}
        FROM {table}
    """

def count_joined(tables: Tuple[str, str], ids: Tuple[str, str]) -> str:
    return f"""--sql
        SELECT
            COUNT(*) AS length_{tables[0]}_{tables[1]}
        FROM {tables[0]} AS first
        INNER JOIN {tables[1]} AS second ON first.{ids[0]} == second.{ids[1]}
    """

def count_joined_distinct(tables: Tuple[str, str], ids: Tuple[str, str]) -> str:
    return f"""--sql
        SELECT
            COUNT(DISTINCT first.{ids[0]}) AS length_{tables[0]}_{tables[1]}_distinct
        FROM {tables[0]} AS first
        INNER JOIN {tables[1]} AS second ON first.{ids[0]} == second.{ids[1]}
    """

for tables, ids in JOINS.items():
    print(tables[0].upper(), '-', tables[1].upper())
    df = spark.sql(count_everything(tables[0]))
    display(df.toPandas())
    df = spark.sql(count_everything(tables[1]))
    display(df.toPandas())
    df = spark.sql(count_joined(tables, ids))
    display(df.toPandas())
    df = spark.sql(count_joined_distinct(tables, ids))
    display(df.toPandas())
```

```
ARTISTS - TRACKS
```

	length_artists
0	27524

	length_tracks
0	129648

	length_artists_tracks
0	116488

	length_artists_tracks_distinct
0	25532

```
TRACKS - TRACK_STORAGE
```

	length_tracks
0	129648

	length_track_storage
0	129648

	length_tracks_track_storage
0	123118

	length_tracks_track_storage_distinct
0	123118

```
TRACKS - SESSIONS
```

	length_tracks
0	129648

	length_sessions
0	3805

	length_tracks_sessions
0	3371

	length_tracks_sessions_distinct
0	2087

```
USERS - SESSIONS
```

	length_users
0	50

	length_sessions
0	3805

	length_users_sessions
0	3622

	length_users_sessions_distinct
0	50

In [ ]:

```python
def select_unknown(tables: Tuple[str, str], ids: Tuple[str, str]) -> str:
    spark.sql(f'SELECT DISTINCT {ids[1]} AS id FROM {tables[1]}') \
        .createOrReplaceTempView('temporary')

    return f"""--sql
        SELECT
            *
        FROM {tables[0]}
        WHERE {ids[0]} NOT IN (SELECT id FROM temporary)
    """

for tables, ids in JOINS.items():
    print(tables[0].upper(), '-', tables[1].upper())
    df = spark.sql(select_unknown(tables, ids))
    display(df.toPandas())
    df = spark.sql(select_unknown(tables[::-1], ids[::-1]))
    display(df.toPandas())
```

```
ARTISTS - TRACKS
```

genres	id	name

	acousticness	danceability	duration_ms	energy	explicit	id	id_artist	instrumentalness	key	liveness	loudness	name	popularity	release_date	speechiness	tempo	valence	release_date_s
0	0.5660	0.629	235227	0.632	0	3vs0E2QJ5DToHw14Q7BDmT	1fa0cOhromAZdq2xRA4vv8	0.000052	8	0.1030	-7.531	You've Got It - 2008 Remaster	47.0	1989	0.0264	82.020	0.799	599612400
1	0.6680	0.502	201600	0.433	0	2LtpyfWWnr5V96l3Js7LLX	0elA30wLp3RmiPaGtU2jhQ	0.002000	8	0.1670	-14.619	Good Morning Little Schoolgirl	41.0	1994-01-01	0.0339	103.467	0.566	757378800
2	0.8470	0.629	153586	0.488	0	61K7dM7FIxTRf0vLM5rZBP	5EBH204cwRkvAWknwTAjCQ	0.005050	9	0.1020	-10.248	Los mismos clavos	48.0	2007-04-24	0.0373	101.724	0.801	1177365600
3	0.3360	0.694	209440	0.773	0	0cliJhcTCsETccUVjgTqK1	6zg73gIYCTCBvxgcKFDACs	0.000000	0	0.0891	-9.332	Bruta Ansiedade	20.0	1988	0.0304	133.559	0.902	567990000
4	0.5640	0.343	250728	0.406	0	4hiIlD3oy2M2ZGz636Oe19	66R6lheFKAZbWWGzGLNCVc	0.000000	10	0.1620	-7.897	Dali cekas stara majcice	1.0	2009-05-25	0.0281	95.962	0.554	1243202400
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
6651	0.0389	0.816	234867	0.601	0	4shT9NEj5vPscunxGHwANS	5B8ApeENp4bE4EE3LI8jK2	0.009000	4	0.1940	-8.001	Siempre Te Amaré (Every Breath You Take)	NaN	2006-10-27	0.0317	124.007	0.791	1161900000
6652	0.0131	0.481	231320	0.860	0	5FF5SlqmbdbYtnJC4yekZE	14T8NkbwXVZgbOvwnuGV89	0.000000	1	0.1530	-4.343	None	35.0	2012-01-01	0.0521	177.988	0.705	1325372400
6653	0.8750	0.601	137667	0.758	0	6DSTcW93SM0daJxYsAwh9p	1WPcVNert9hn7mHsPKDn7j	0.000000	3	0.6780	-6.628	L'Homme à la moto - Live À L'Olympia 1956	3.0	1956	0.8080	146.960	0.784	-441853200
6654	0.0639	0.474	284947	0.434	0	7xgsy9hOPgQSoDR9g1308P	4HHdjvdn30koo54zQ6QeF5	0.000000	9	0.1270	-4.993	Kekasih Gelapku	45.0	2007-08-01	0.0301	117.915	0.364	1185919200
6655	0.2230	0.516	133706	0.741	0	766SLubGdNRCoBFfTue0AY	1kupwLFpHALpmhp5qol8xH	0.000108	3	0.1520	-9.039	Here I Go Again	27.0	1969	0.0381	162.074	0.862	-31539600

6656 rows × 18 columns

```
TRACKS - TRACK_STORAGE
```

acousticness	danceability	duration_ms	energy	explicit	id	id_artist	instrumentalness	key	liveness	loudness	name	popularity	release_date	speechiness	tempo	valence	release_date_s

daily_cost	storage_class	track_id

```
TRACKS - SESSIONS
```

acousticness	danceability	duration_ms	energy	explicit	id	id_artist	instrumentalness	key	liveness	loudness	name	popularity	release_date	speechiness	tempo	valence	release_date_s

event_type	session_id	timestamp	track_id	user_id	timestamp_s

```
USERS - SESSIONS
```

city	favourite_genres	id	name	premium_user	street	user_id

event_type	session_id	timestamp	track_id	user_id	timestamp_s

In [ ]:

```
premium_user_comparison = f"""--sql
    SELECT
        COUNT_IF(premium_user == 1) AS premium_users,
        COUNT_IF(premium_user == 0) AS non_premium_users,
        COUNT_IF(premium_user == 0) / COUNT(*) * 100 AS non_premium_users_percentage,
        COUNT_IF(premium_user == 1) / COUNT(*) * 100 AS premium_users_percentage
    FROM users
"""
df = spark.sql(premium_user_comparison)
display(df.toPandas())
```

	premium_users	non_premium_users	non_premium_users_percentage	premium_users_percentage
0	48	0	0.0	96.0