Questionnaire

Angelin Ann Jacob

Dataset Name:

Open University Learning Analytics Dataset (OULAD)
Online Learning Platform Data with Student Demographics

Dataset Link:

OULAD Dataset Kaggle Link

Question 1: Which type of assessment had the highest weightage?
Ans format - string value of the assessment type
Ans: Exam

Question 2: List the
1] Number of Years
2] Number of semesters/code presentation
3] Number of names given for modules for a year
Taken into consideration for this dataset
Ans format - list of years, list of semester names, list of modules
Ans: 2; 4; 7

Question 3: What is the ratio of Female to Male who
1] belonged to 2014J
2] belonged to Scotland and
3] Post Graduation being their highest education?
Ans format - decimal/float of the fractional ratio. If its 1:10 return 0.1, 1:2 return 0.5, round to 2 positions
Ans: 0.33

Question 4: Which student had the highest number of studied credits. What semester did he belong to and where was he/she from?
Ans format - int of the ID; String of semester; String of the place
Ans: 363151, 2013J, North Western Region;

Question 5: Find the top Distinction Rate (No. of Distinction per module/Total No. of Distinction).
Ans format - float value
Ans: 22.39

Question 6:

a] For the semester 2013J, which code module needed the highest number of assessments to be submitted?

b] Also on average, how many assessments per code module were needed in 2013J? Hint: in 2013J the courses were AAA, BBB, DDD, EEE, FFF, GGG each having some number of assessments to be submitted. Find the average of these counts.

Ans format - string of code module, int value of ceil(avg float value)

Ans: FFF, 9

Question 7: Find the ratio of Female to Males students who unregistered early in a codemodule after the course commencement. *Hint - All these questions will require inner joins, use PowerBI (free) or Tableau*

Ans format - decimal/float of the fractional ratio. If its 1:10 return 0.1, 1:2 return 0.5, round to 2 positions

Ans: 0.82

Ans: In StudetnInfo.csv and StudentRegistration.csv, for all positive registration numbers, join on student_id, then find the count for Female and Male, divide and find ratio

Question 8: For the student with the id_student is *Hint - All these questions will require inner joins, use PowerBI (free) or Tableau*

A] 11391

B] 83219

C] 689246

D] 367646

Total marks received

$$\text{Total Marks}_{\text{id\_student}} = (\text{sum}_{\text{group by id\_student}}\ (\text{Score}_{\text{id\_assessment}} * \text{weight}_{\text{id\_assessment}}))$$

Ans format - float values for Percentage

Ans: 82.4; 232.63; 182.01; 175

Ans: In studentAssessment.csv and assessments.csv, join on id_assessment, query for id_student in the above mentioned list, apply formula

Question 9: Find the number of students who a had score less than 50, who had a disability and withdrew from the course and banked in the semester *Hint - All these questions will require inner joins, use PowerBI (free) or Tableau*

Ans format - int

Ans: 35

Question10: By how much did 11391 perform better 45642 in terms of total marks acquired.

$$\text{Total Marks}_{\text{id\_student}} = (\text{sum}_{\text{group by id\_student}}\ (\text{Score}_{\text{id\_assessment}} * \text{weight}_{\text{id\_assessment}}))$$

Hint - All these questions will require inner joins, use PowerBI (free) or Tableau

Ans format - int

Ans: 990

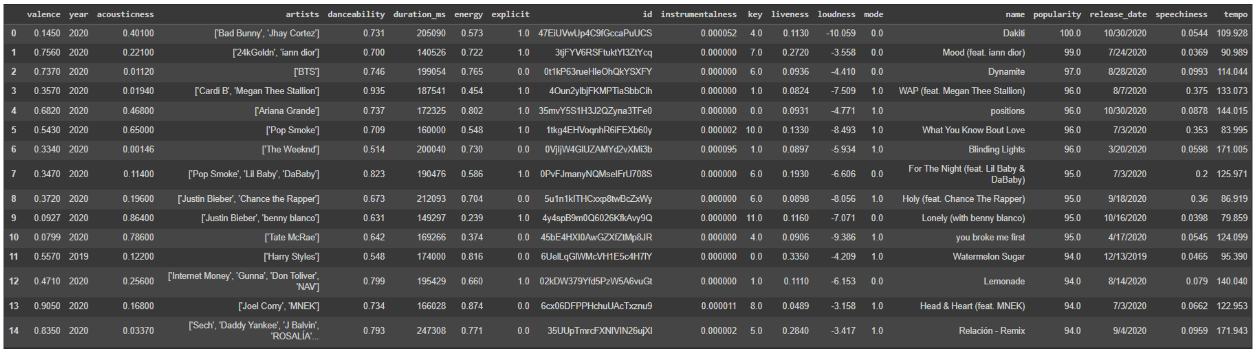
Dataset Name:

Spotify music analysis

Dataset Link:

Spotify Music Analysis

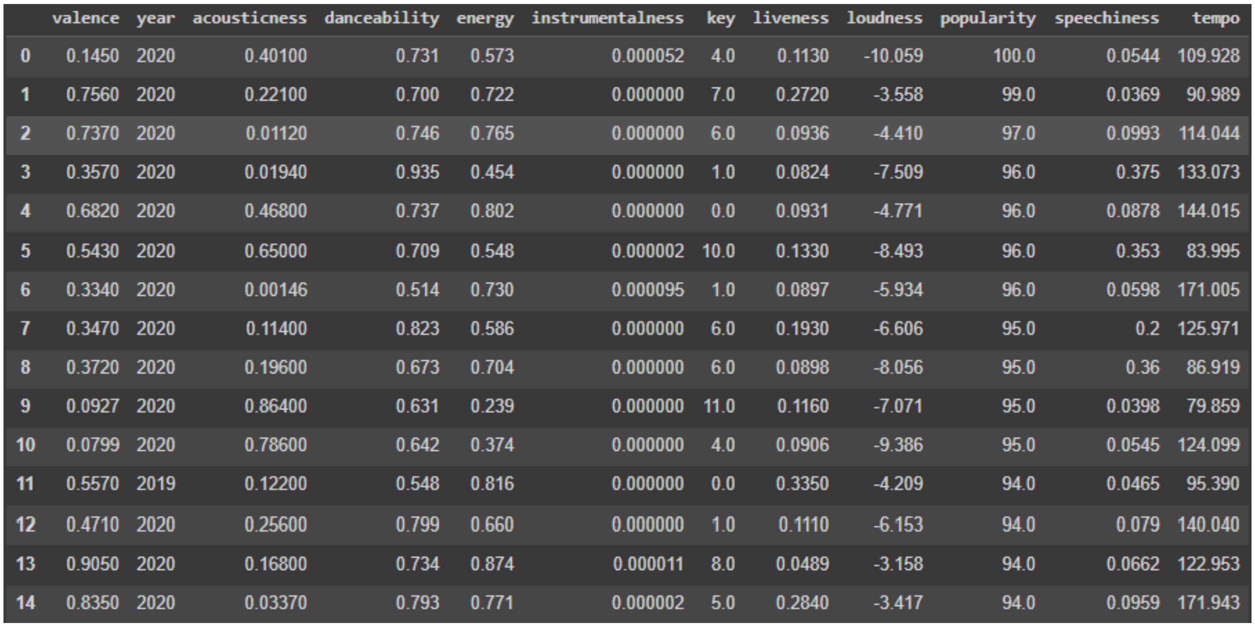
Task1: Read the csv from the given kaggle link, take only the first 15 songs (rows) and store it in a dataframe. (Make sure the names of the song match what you've got, it must be FIRST 15 ONLY, NOT sampled/randomized)



	valence	year	acousticness	artists	danceability	duration_ms	energy	explicit	id	instrumentalness	key	liveness	loudness	mode	name	popularity	release_date	speechiness	tempo
0	0.1450	2020	0.40100	['Bad Bunny', 'Jhay Cortez']	0.731	205090	0.573	1.0	47EiUVwUp4C9fGccaPuUCS	0.000052	4.0	0.1130	-10.059	0.0	Dakiti	100.0	10/30/2020	0.0544	109.928
1	0.7560	2020	0.22100	['24kGoldn', 'iann dior']	0.700	140526	0.722	1.0	3tjFYV6RSFtuktYl3ZtYcq	0.000000	7.0	0.2720	-3.558	0.0	Mood (feat. iann dior)	99.0	7/24/2020	0.0369	90.989
2	0.7370	2020	0.01120	['BTS']	0.746	199054	0.765	0.0	0t1kP63rueHleOhQkYSXFY	0.000000	6.0	0.0936	-4.410	0.0	Dynamite	97.0	8/28/2020	0.0993	114.044
3	0.3570	2020	0.01940	['Cardi B', 'Megan Thee Stallion']	0.935	187541	0.454	1.0	4Oun2ylbjFKMPTiaSbbCih	0.000000	1.0	0.0824	-7.509	1.0	WAP (feat. Megan Thee Stallion)	96.0	8/7/2020	0.375	133.073
4	0.6820	2020	0.46800	['Ariana Grande']	0.737	172325	0.802	1.0	35mvY5S1H3J2QZyna3TFe0	0.000000	0.0	0.0931	-4.771	1.0	positions	96.0	10/30/2020	0.0878	144.015
5	0.5430	2020	0.65000	['Pop Smoke']	0.709	160000	0.548	1.0	1tkg4EHVoqnhR6iFEXb60y	0.000002	10.0	0.1330	-8.493	1.0	What You Know Bout Love	96.0	7/3/2020	0.353	83.995
6	0.3340	2020	0.00146	['The Weeknd']	0.514	200040	0.730	0.0	0VjIjW4GlUZAMYd2vXMi3b	0.000095	1.0	0.0897	-5.934	1.0	Blinding Lights	96.0	3/20/2020	0.0598	171.005
7	0.3470	2020	0.11400	['Pop Smoke', 'Lil Baby', 'DaBaby']	0.823	190476	0.586	1.0	0PvFJmanyNQMseIFrU708S	0.000000	6.0	0.1930	-6.606	0.0	For The Night (feat. Lil Baby & DaBaby)	95.0	7/3/2020	0.2	125.971
8	0.3720	2020	0.19600	['Justin Bieber', 'Chance the Rapper']	0.673	212093	0.704	0.0	5u1n1kITHCxxp8twBcZxWy	0.000000	6.0	0.0898	-8.056	1.0	Holy (feat. Chance The Rapper)	95.0	9/18/2020	0.36	86.919
9	0.0927	2020	0.86400	['Justin Bieber', 'benny blanco']	0.631	149297	0.239	1.0	4y4spB9m0Q6026KfkAvy9Q	0.000000	11.0	0.1160	-7.071	0.0	Lonely (with benny blanco)	95.0	10/16/2020	0.0398	79.859
10	0.0799	2020	0.78600	['Tate McRae']	0.642	169266	0.374	0.0	45bE4HXI0AwGZXfZtMp8JR	0.000000	4.0	0.0906	-9.386	1.0	you broke me first	95.0	4/17/2020	0.0545	124.099
11	0.5570	2019	0.12200	['Harry Styles']	0.548	174000	0.816	0.0	6UelLqGlWMcVH1E5c4H7lY	0.000000	0.0	0.3350	-4.209	1.0	Watermelon Sugar	94.0	12/13/2019	0.0465	95.390
12	0.4710	2020	0.25600	['Internet Money', 'Gunna', 'Don Toliver', 'NAV']	0.799	195429	0.660	1.0	02kDW379Yfd5PzW5A6vuGt	0.000000	1.0	0.1110	-6.153	0.0	Lemonade	94.0	8/14/2020	0.079	140.040
13	0.9050	2020	0.16800	['Joel Corry', 'MNEK']	0.734	166028	0.874	0.0	6cx06DFPPHchuUAcTxznu9	0.000011	8.0	0.0489	-3.158	1.0	Head & Heart (feat. MNEK)	94.0	7/3/2020	0.0662	122.953
14	0.8350	2020	0.03370	['Sech', 'Daddy Yankee', 'J Balvin', 'ROSALÍA'...	0.793	247308	0.771	0.0	35UUpTmrcFXNIVIN26ujXI	0.000002	5.0	0.2840	-3.417	1.0	Relación - Remix	94.0	9/4/2020	0.0959	171.943

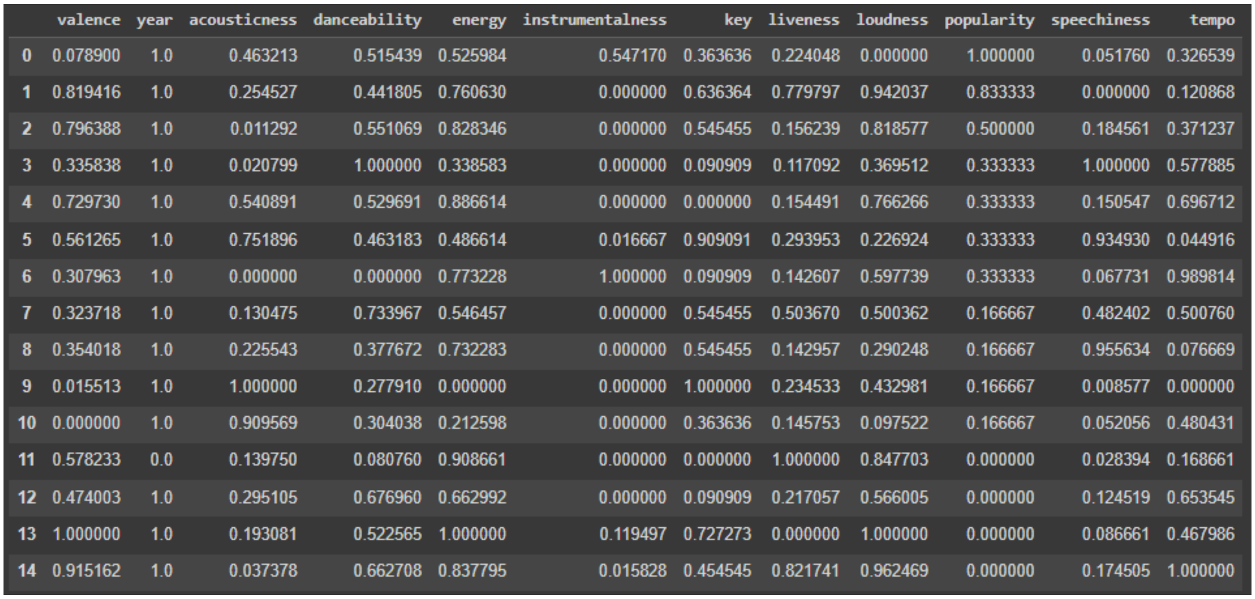
Task2: You'll have 19 columns, out of which, we make another dataframe which will contain ['valence', 'year', 'acousticness', 'danceability', 'energy', 'instrumentalness', 'key', 'liveness', 'loudness', 'popularity', 'speechiness', 'tempo'] columns. The new dataframe must now have only 12 columns.

The dataframes shape must be (15, 12)



	valence	year	acousticness	danceability	energy	instrumentalness	key	liveness	loudness	popularity	speechiness	tempo
0	0.1450	2020	0.40100	0.731	0.573	0.000052	4.0	0.1130	-10.059	100.0	0.0544	109.928
1	0.7560	2020	0.22100	0.700	0.722	0.000000	7.0	0.2720	-3.558	99.0	0.0369	90.989
2	0.7370	2020	0.01120	0.746	0.765	0.000000	6.0	0.0936	-4.410	97.0	0.0993	114.044
3	0.3570	2020	0.01940	0.935	0.454	0.000000	1.0	0.0824	-7.509	96.0	0.375	133.073
4	0.6820	2020	0.46800	0.737	0.802	0.000000	0.0	0.0931	-4.771	96.0	0.0878	144.015
5	0.5430	2020	0.65000	0.709	0.548	0.000002	10.0	0.1330	-8.493	96.0	0.353	83.995
6	0.3340	2020	0.00146	0.514	0.730	0.000095	1.0	0.0897	-5.934	96.0	0.0598	171.005
7	0.3470	2020	0.11400	0.823	0.586	0.000000	6.0	0.1930	-6.606	95.0	0.2	125.971
8	0.3720	2020	0.19600	0.673	0.704	0.000000	6.0	0.0898	-8.056	95.0	0.36	86.919
9	0.0927	2020	0.86400	0.631	0.239	0.000000	11.0	0.1160	-7.071	95.0	0.0398	79.859
10	0.0799	2020	0.78600	0.642	0.374	0.000000	4.0	0.0906	-9.386	95.0	0.0545	124.099
11	0.5570	2019	0.12200	0.548	0.816	0.000000	0.0	0.3350	-4.209	94.0	0.0465	95.390
12	0.4710	2020	0.25600	0.799	0.660	0.000000	1.0	0.1110	-6.153	94.0	0.079	140.040
13	0.9050	2020	0.16800	0.734	0.874	0.000011	8.0	0.0489	-3.158	94.0	0.0662	122.953
14	0.8350	2020	0.03370	0.793	0.771	0.000002	5.0	0.2840	-3.417	94.0	0.0959	171.943

Task3: Standardize all values in the dataset using the Min Max Scaler from SciKit.



	valence	year	acousticness	danceability	energy	instrumentalness	key	liveness	loudness	popularity	speechiness	tempo
0	0.078900	1.0	0.463213	0.515439	0.525984	0.547170	0.363636	0.224048	0.000000	1.000000	0.051760	0.326539
1	0.819416	1.0	0.254527	0.441805	0.760630	0.000000	0.636364	0.779797	0.942037	0.833333	0.000000	0.120868
2	0.796388	1.0	0.011292	0.551069	0.828346	0.000000	0.545455	0.156239	0.818577	0.500000	0.184561	0.371237
3	0.335838	1.0	0.020799	1.000000	0.338583	0.000000	0.090909	0.117092	0.369512	0.333333	1.000000	0.577885
4	0.729730	1.0	0.540891	0.529691	0.886614	0.000000	0.000000	0.154491	0.766266	0.333333	0.150547	0.696712
5	0.561265	1.0	0.751896	0.463183	0.486614	0.016667	0.909091	0.293953	0.226924	0.333333	0.934930	0.044916
6	0.307963	1.0	0.000000	0.000000	0.773228	1.000000	0.090909	0.142607	0.597739	0.333333	0.067731	0.989814
7	0.323718	1.0	0.130475	0.733967	0.546457	0.000000	0.545455	0.503670	0.500362	0.166667	0.482402	0.500760
8	0.354018	1.0	0.225543	0.377672	0.732283	0.000000	0.545455	0.142957	0.290248	0.166667	0.955634	0.076669
9	0.015513	1.0	1.000000	0.277910	0.000000	0.000000	1.000000	0.234533	0.432981	0.166667	0.008577	0.000000
10	0.000000	1.0	0.909569	0.304038	0.212598	0.000000	0.363636	0.145753	0.097522	0.166667	0.052056	0.480431
11	0.578233	0.0	0.139750	0.080760	0.908661	0.000000	0.000000	1.000000	0.847703	0.000000	0.028394	0.168661
12	0.474003	1.0	0.295105	0.676960	0.662992	0.000000	0.090909	0.217057	0.566005	0.000000	0.124519	0.653545
13	1.000000	1.0	0.193081	0.522565	1.000000	0.119497	0.727273	0.000000	1.000000	0.000000	0.086661	0.467986
14	0.915162	1.0	0.037378	0.662708	0.837795	0.015828	0.454545	0.821741	0.962469	0.000000	0.174505	1.000000

Question 1: Find the pearson correlation between the 'loudness' and the 'energy' of songs (Note: to min max scale and then perform correlation testing)

Ans format - float

Ans: 0.7062151014613735

Question 2: Find the max correlation measure amongst the columns present in this dataframe.

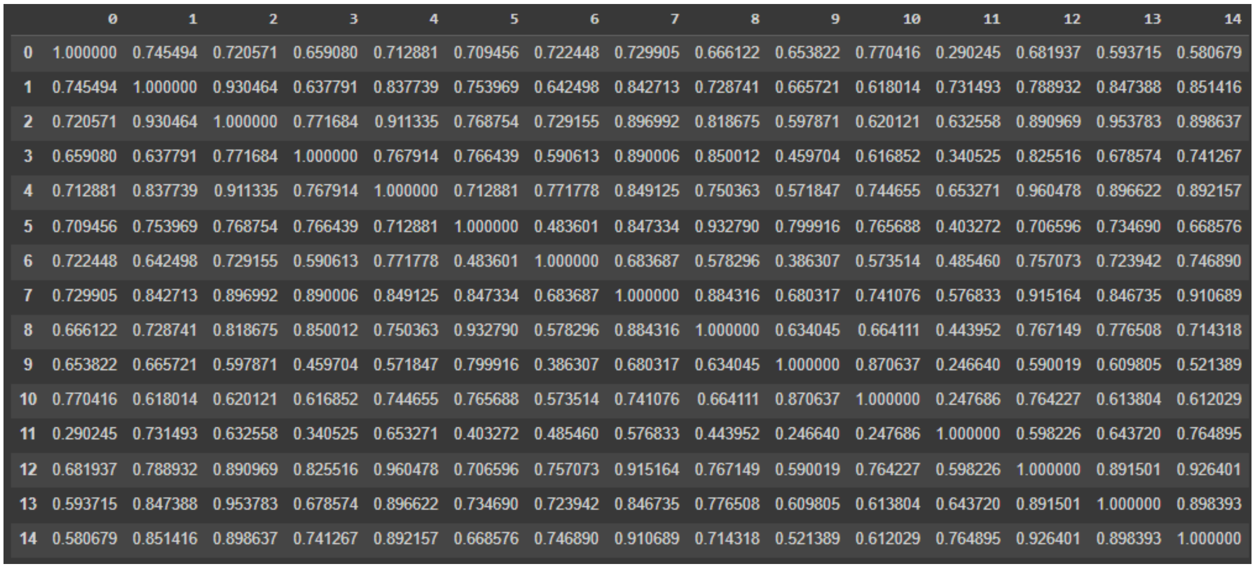
Ans format - float

Ans: 0.8498293076162325

Task4: For each song, find the similarity with every other song. Use cosine distance from SciPy for measuring the similarity. Hint: run an i loop for every song, within this loop, run a j loop for every other song. Find the Similarity and store the result in a new dataframe, where the new value will be

`Sim_df[i][j] = similarity between the song i and song j`

Resulting data frame must look like this 15 × 15 similarity matrix, where sim_df[i][j] = similarity between the ith song and jth song.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	1.000000	0.745494	0.720571	0.659080	0.712881	0.709456	0.722448	0.729905	0.666122	0.653822	0.770416	0.290245	0.681937	0.593715	0.580679
1	0.745494	1.000000	0.930464	0.637791	0.837739	0.753969	0.642498	0.842713	0.728741	0.665721	0.618014	0.731493	0.788932	0.847388	0.851416
2	0.720571	0.930464	1.000000	0.771684	0.911335	0.768754	0.729155	0.896992	0.818675	0.597871	0.620121	0.632558	0.890969	0.953783	0.898637
3	0.659080	0.637791	0.771684	1.000000	0.767914	0.766439	0.590613	0.890006	0.850012	0.459704	0.616852	0.340525	0.825516	0.678574	0.741267
4	0.712881	0.837739	0.911335	0.767914	1.000000	0.712881	0.771778	0.849125	0.750363	0.571847	0.744655	0.653271	0.960478	0.896622	0.892157
5	0.709456	0.753969	0.768754	0.766439	0.712881	1.000000	0.483601	0.847334	0.932790	0.799916	0.765688	0.403272	0.706596	0.734690	0.668576
6	0.722448	0.642498	0.729155	0.590613	0.771778	0.483601	1.000000	0.683687	0.578296	0.386307	0.573514	0.485460	0.757073	0.723942	0.746890
7	0.729905	0.842713	0.896992	0.890006	0.849125	0.847334	0.683687	1.000000	0.884316	0.680317	0.741076	0.576833	0.915164	0.846735	0.910689
8	0.666122	0.728741	0.818675	0.850012	0.750363	0.932790	0.578296	0.884316	1.000000	0.634045	0.664111	0.443952	0.767149	0.776508	0.714318
9	0.653822	0.665721	0.597871	0.459704	0.571847	0.799916	0.386307	0.680317	0.634045	1.000000	0.870637	0.246640	0.590019	0.609805	0.521389
10	0.770416	0.618014	0.620121	0.616852	0.744655	0.765688	0.573514	0.741076	0.664111	0.870637	1.000000	0.247686	0.764227	0.613804	0.612029
11	0.290245	0.731493	0.632558	0.340525	0.653271	0.403272	0.485460	0.576833	0.443952	0.246640	0.247686	1.000000	0.598226	0.643720	0.764895
12	0.681937	0.788932	0.890969	0.825516	0.960478	0.706596	0.757073	0.915164	0.767149	0.590019	0.764227	0.598226	1.000000	0.891501	0.926401
13	0.593715	0.847388	0.953783	0.678574	0.896622	0.734690	0.723942	0.846735	0.776508	0.609805	0.613804	0.643720	0.891501	1.000000	0.898393
14	0.580679	0.851416	0.898637	0.741267	0.892157	0.668576	0.746890	0.910689	0.714318	0.521389	0.612029	0.764895	0.926401	0.898393	1.000000

Question 3:

a] Which song has the highest similarity to which other song. (Give Result only in terms of the song name found in the 'name' table in the original dataset.)
b] Enter song1 here
c] Enter song2 here
Ans format - a] float   b] string1 c] string2
Ans: a] 0.960478 b] What You Know Bout Love c] Lemonade

Question 4: How similar is Ariana Grande and The Weeknd
Ans format - float
Ans: 0.771778

Question 5: How similar is the song from the month of March and the song from December.
Ans format - float
Ans: 0.485460

Question 6: What's the similarity measure of the most similar song to BTS.
Ans format - int
Ans: 0.953783

Question 7: Answer True if the fastest song is atleast 50% similar to the slowest song in the dataset.
Ans format - boolean/string
Ans: True

Question 8: What are the two most popular songs? (Give Result only in terms of the song name found in the 'name' table in the original dataset.)
a] Enter song1 here
b] Enter song2 here
c] Find the similarity measure between them.
Ans format - a] string1 b] string2 c] float
Ans: a] Dakiti b] Mood (feat. iann dior) c] 0.745494

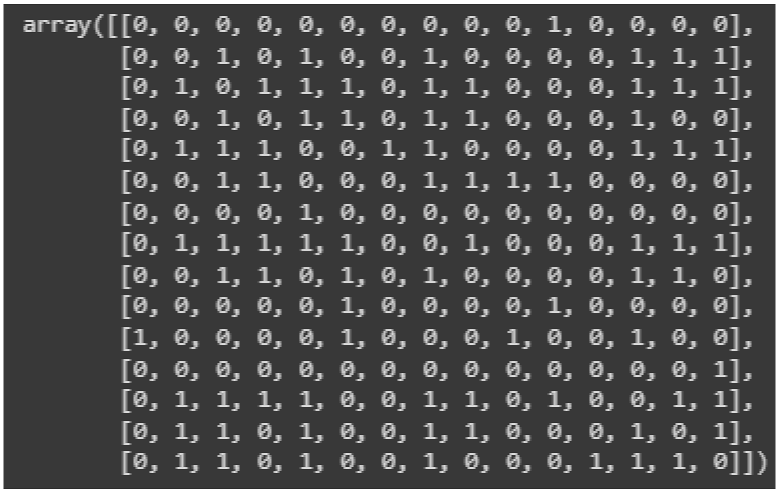
Question 9: Find the sum of similarities across the songs which are in the key "4"
Ans format - float
Ans: 0.770416

Question 10: Find the cumulative average similarity score for each song (Make sure to not include the song itself), and then find the song which is in overall highest similarity to every other song.
Ans format - give the index number for the song, int
Ans: 7

Question 11: Find the average of similarities across the two songs which were released on the same day (month - october)
Ans format - float
Ans: 0.712881

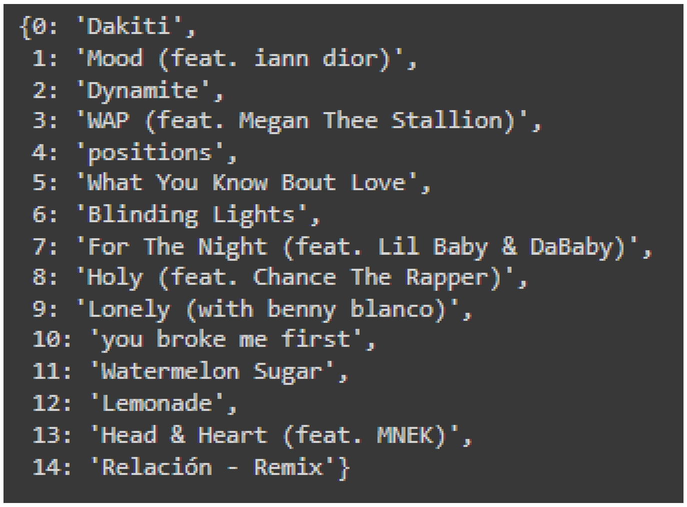
Question 12: Find the cumulative average of all the values in the dataframe (excluding the 1s, remember to not count the song itself) (divide by 14 NOT 15)
Ans format - float
Ans:0.7608877190792983

Task6: Form a matrix with only 1s and 0s, where 1 is filled if the similarity measure is more than the cumulative average, 0s filled if it is lesser than the cumulative average. The Matrix should be something like this. Save this matrix as `adj_mat` (Adjacency Matrix). (note to fill the diagonals with 0)



```
array([[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0],
       [0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1],
       [0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1],
       [0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0],
       [0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1],
       [0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0],
       [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1],
       [0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0],
       [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0],
       [1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1],
       [0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1],
       [0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1],
       [0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0]])
```

Task7: Run a loop to get a dictionary which has the index number as the key and the song name as the value. Should look like this Hint Name this dictionary as `dicts`



```
{0: 'Dakiti',
 1: 'Mood (feat. iann dior)',
 2: 'Dynamite',
 3: 'WAP (feat. Megan Thee Stallion)',
 4: 'positions',
 5: 'What You Know Bout Love',
 6: 'Blinding Lights',
 7: 'For The Night (feat. Lil Baby & DaBaby)',
 8: 'Holy (feat. Chance The Rapper)',
 9: 'Lonely (with benny blanco)',
 10: 'you broke me first',
 11: 'Watermelon Sugar',
 12: 'Lemonade',
 13: 'Head & Heart (feat. MNEK)',
 14: 'Relación - Remix'}
```

Task8: Run this code to plot out a graph for the above adjacency matrix: `adj_mat`, with the labels as the above made dictionary: `dicts`.

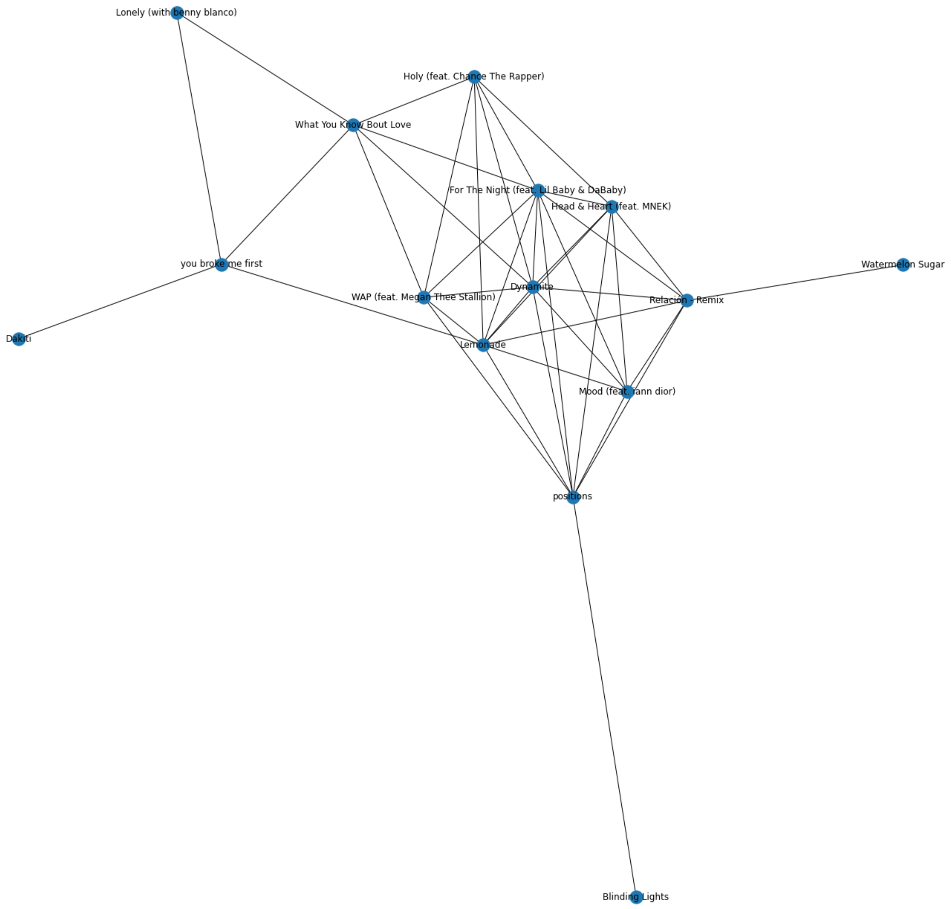
```
import networkx as nx
import matplotlib.pyplot as plt

G = nx.Graph(adj_mat, seed= 123)


fig = plt.figure(figsize=(20,20))
nx.draw(G, labels = dicts, with_labels = True)

plt.show()
```

This will make up a graph, we will now perform graph operations for the same.


Lonely (with benny blanco)
Holy (feat. Chance The Rapper)
What You Know Bout Love
For The Night (feat. Lil Baby & DaBaby)
Head & Heart (feat. MNEK)
you broke me first
Watermelon Sugar
Dynamite
WAP (feat. Megan Thee Stallion)
Relación - Remix
Dakiti
Lemonade
Mood (feat. iann dior)
positions
Blinding Lights

Question 13: Find the

A] **density** of this graph (float number)

B] **degree histogram** (list)

C] **diameter** (int)

Ans format - float

Ans: 0.3904761904761905, [0, 3, 1, 0, 1, 0, 4, 2, 1, 3], 4

Question 14: What is the size of the biggest and smallest **cliques** formed in this graph?

Ans format - int, int

Ans: 7, 2

Question 15: What is the average tempo for the most well connected/biggest clique?

Ans format - float, round it to 1, int

Ans: 130

Question 16: Who has the smallest **degree centrality** of all the songs in the biggest **clique**.

Ans format - float

Ans: 0.42857142857142855

Question 17: Make the smallest **clique** and largest clique as subgraphs. Find the **Graph Edit Distance** between these graphs.

Ans format - int

Ans: 25

Question 18: Find size of the biggest community among the 4th level **community** in this graph.

Ans format - int

Ans: 10

Question 19: What are the songs which are similar to both

a] Dakiti and Lonely

b] Blinding Lights and Mood (feat. iann dior)

c] Specify the name of the function used. (if the function is nx.function_name(), answer will be function_name)

Think of the **neighbors** of these nodes. Give in terms of node number.

Ans format - a] int; b] int; c] string

Ans: 10, 4, common_neighbors

Question 20:

a] Give the index of the first occurring song in the list which isn't a neighbor of the node ‘positions’

b] Specify the name of the function used. (if the function is nx.function_name(), answer will be function_name)

Ans format - a] int; b] String

Ans: 0, non_neighbors

Question 21: Give the index of the song which is not similar to every other songs but one, ie. has an **edge** to no nodes but one. Specify the name of the function used. (if the function is nx.function_name(), answer will be function_name)

Ans format - a] int; b] String

Ans: 0, edges