M21CS4.301: Data and Applications Due: **23:59, 16th October 2021**
Group Assignment 4 Released: 10th October, 2021

In this assignment, you have to write the MySQL queries that satisfy the questions posed below. Try not to overthink things. You are allowed to use intermediate views for these questions. Do not create new tables unless the questions asks you to do so.

Problem 1

You are a researcher working on 3D computer vision problems. You've been given a dataset comprising of 3D point cloud data - (x, y, z) coordinates of the world.

The dataset is defined as:
A table POINT with attributes X, Y, Z where each row is a single point (x, y, z). You can assume that height is in the z axis. Here's a sample dataset to make things clearer:

X	Y	Z
1	1	0
-3	0	0
1	-2	3

Solve the questions below using SQL statements to better understand your data.

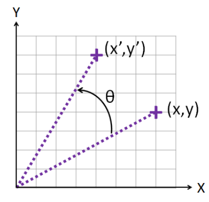
a. Display all the points in increasing order of x, then y, then z (if two records have the same value for x, order them by y. If they have the same value for both x and y, order them by z).

b. Find the highest and lowest points according to their elevation.

c. Find the k nearest points to the origin, using Euclidean distance.

d. Find the centroid of all the points.

e. Create a 3x3 rotation matrix in a table ROTATE that rotates anticlockwise about the Z axis with a variable angle α in radians as input.
If this rotation was in 2D, it would look like this



The new coordinates of x and y would be

$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

The rotation matrix R would be,

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

where

$$R \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

f. Use the ROTATE table to rotate all the points in POINTS anticlockwise about the Z axis with a variable angle α and store them in a table called ROTATED. (Hint: you can update the X,Y,Z coordinates of all points in three separate queries. Do NOT write queries to update the coordinates of each point)

Problem 2

While building search engines and similar applications, it is important to have an index which tells you the frequency of words in documents. Each record in the table given below tells us that a word X occurred in the document Y, n times. The dataset has two tables:

1. INVINDEX - with attributes 'WORD', 'DOC_ID', 'FREQ' where each row represents that the WORD W occurred in document with DOC_ID dw with frequency N

WORD	DOC_ID	FREQ
Hello	4	20
Marvel	2	4
World	12	9
Blackpink	41	5
Bonda	12	20
Hammer	6	9

2. ENTITY - with attributes 'WORD1', 'WORD2' and 'DOC_ID' where each row represents that there's a link between WORD1 W1 and WORD2 W2 obtained from document D with document id DOC ID dw.

WORD1	WORD2	DOC_ID
Pizza	Italy	5
Margherita	Pizza	21
Italy	Grammy	8
Italy	Pasta	5
Taylor	Singer	8
Modi	Dancer	13
Pasta	Pineapple	5

Answer the following questions on a similarly constructed dataset.

a. For the phrase "Hello World", print all the document IDs and the corresponding scores for the documents (only the non-zero scores). The score of a word W in a document D is:

$$\text{score} = \frac{\text{freq of W in D}}{\text{total number of words in D}}$$

For a phrase P, the score of a document D is the sum of scores of all words in P.

b. In the ENTITY relation, each link between two words is equivalent to a directed edge in a graph. In a graph, a node Y is at a distance k from a node X if there exist k edges linking node X to node Y.

That is, Y is at a k-hop distance from X. Here, a word W2 is at a distance k from word W1 if there exist k links linking W1 and W2, with all k links occurring in the same document ID.

For a given entity “Pizza” print all entities which occur at a distance 3 from “Pizza” and all the 3 links are retrieved from the same document. Also print the corresponding document IDs. You do not need to check for the shortest path between the words, if there is a path of length 3 from "Pizza", give the word and the document ID.

For example, the sample output for the above query would be:
Pineapple 5

c. For the documents retrieved in question 2, print the frequency of the word ”Lotus”.

d. The average score of a word is given by the sum of scores of the word across all documents (where it appears) divided by the number of these documents. Display the words in decreasing order of average score.

e. Display "Can’t believe the assignments are all done!".

Submission Instructions

Submit a file **<teamnumber.tar.gz>** (without the angled brackets) which, upon extracting will result in the following directory structure:

```
<teamnumber>
 |__1
 |   |__a.txt
 |   |__b.txt
 |   |__c.txt
 |   |__d.txt
 |   |__e.txt
 |   |__f.txt
 |__2
     |__a.txt
     |__b.txt
     |__c.txt
     |__d.txt
     |__e.txt
```

where each of the .txt file has the MySQL query for that particular sub-question.

Only **one** person from each team should make a submission.