Report - Assignment 2

CS3550: DBMS-1

Prof. Manohar Kaul

Naitik Malav | CS19BTECH11026

1. Counting:

I have left join the artists with the albums on ArtistId which gives us the count of the artists who have albums to their name, including those artists that do not.
Then I have taken the first alphabet of the artist name and the count of artists per alphabet.

So below is the count.

A|32
B|27
C|25
D|19
E|14
F|15
G|15
H|6
I|24
J|19
K|5
L|31
M|33
N|8
O|18
P|15
Q|3
R|16
S|32
T|29
U|11
V|14
W|2
X|1
Y|3
Z|1

2. Full Outer Join:

First, I have created an instructor and student table. Then I have feed the data into the tables.
Below I have attached the screenshot for the tables.

```
sqlite> .schema
CREATE TABLE instructor (
name TEXT,
course_id TEXT
);
CREATE TABLE student (
name TEXT,
course_id TEXT
);
sqlite>
sqlite>
sqlite> select * from instructor;
Amy|CS1000
Aaron|CS700
Anne|CS400
Amy|CS1000
Aaron|CS700
Anne|CS400
sqlite>
sqlite>
sqlite> select * from student;
Jack|CS800
Jones|CS1000
Jason|CS450
Jack|CS800
Jones|CS1000
Jason|CS450
sqlite>
```

After that I have taken the left join of instructor and student; and left join of student and instructor, then join of both. It gives us the full outer join.

Below is the output generated.

CS1000|Amy|Jones
CS400|Anne|
CS450||Jason
CS700|Aaron|
CS800||Jack

3. Track counts

I have classified the tracks based on the play length.
   a. "short": less than a minute,
   b. "medium": between 1 and 5 minutes,
   c. "long": greater than 5 minutes.

Below is the count-

```
1069|long
2407|medium
27|short
```

4. Generate some combinations!

I have created the table X and inserted the given data. Below attached is the screenshot of the table.

```
sqlite> .schema
CREATE TABLE X (
id_num integer,
id_str text
);
sqlite> select * from X;
1|A
2|B
3|C
4|D
5|E
```

Below is the generated combinations satisfying given conditions.

```
1,2,3|A,B,C
1,2,4|A,B,D
1,2,5|A,B,E
1,3,4|A,C,D
```

```
1,3,5|A,C,E
1,4,5|A,D,E
2,3,4|B,C,D
2,3,5|B,C,E
2,4,5|B,D,E
3,4,5|C,D,E
```