**Question3:**

**Kim, Tae Young (tyk252)**

**Fang, Minda (mf3308)**

In this part we use **MySQL** to solve the question.

For the two *friend.txt* and *like.txt* files, we delete the first line of the file and then import them into DB.

Create table ***friend***:

|  |
| --- |
| drop table if exists friend;  create table friend (  　person1 int,  person2 int,  index (person1),  index (person2),  index (person1, person2)  ); |

Create table ***liketable*** *(notice that like is a reserved word in MySQL, so we choose the name liketable)*:

|  |
| --- |
| drop table if exists liketable;  create table liketable (  　person INT,  artist INT,  index (person),  index (artist),  index (person, artist)  ); |

Load table ***friend***:

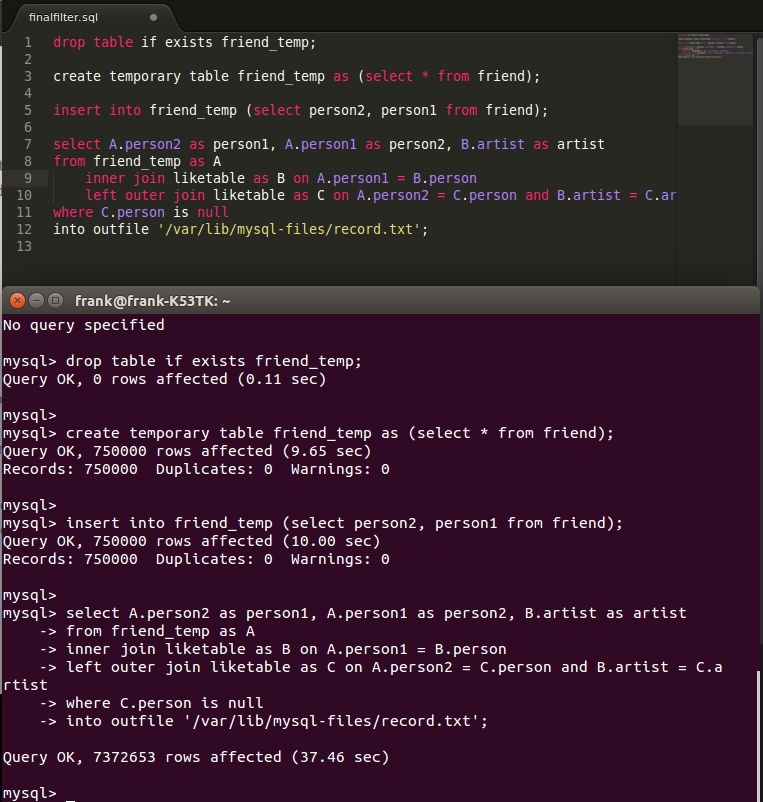
|  |
| --- |
| LOAD DATA INFILE '/var/lib/mysql-files/friends.txt' INTO TABLE question3.friend  FIELDS  TERMINATED BY ',' |

Load table ***liketable***:

|  |
| --- |
| LOAD DATA INFILE '/var/lib/mysql-files/like.txt' INTO TABLE question3.liketable  FIELDS  TERMINATED BY ',' |

***Function***:

|  |
| --- |
| drop table if exists friend\_temp;  create temporary table friend\_temp as (select \* from friend);  insert into friend\_temp (select person2, person1 from friend);  select A.person2 as person1, A.person1 as person2, B.artist as artist  from friend\_temp as A  inner join liketable as B on A.person1 = B.person  left outer join liketable as C on A.person2 = C.person and B.artist = C.artist  where C.person is null; |



The idea first of all is to create a temporary table based on the “friend” data, which lists the identities of person1 and person2. Now, the relations among all persons are symmetric, that is the temporary table will not contain record duplicates. We can therefore use ordinary joins, first by applying an inner join between the temp table and the “like” data, which lists the identities of the artists and the persons, by filtering out the artists favored by a user’s friends . Then we apply an outer join, an outer left join specifically with the bigger subset as the criterion, to mark null in records where the user too lists that artist as its favorite. This query takes roughly 37 seconds, which isn’t too bad.

The output file(“***record.txt***”) is put in the same directory as this PDF file.