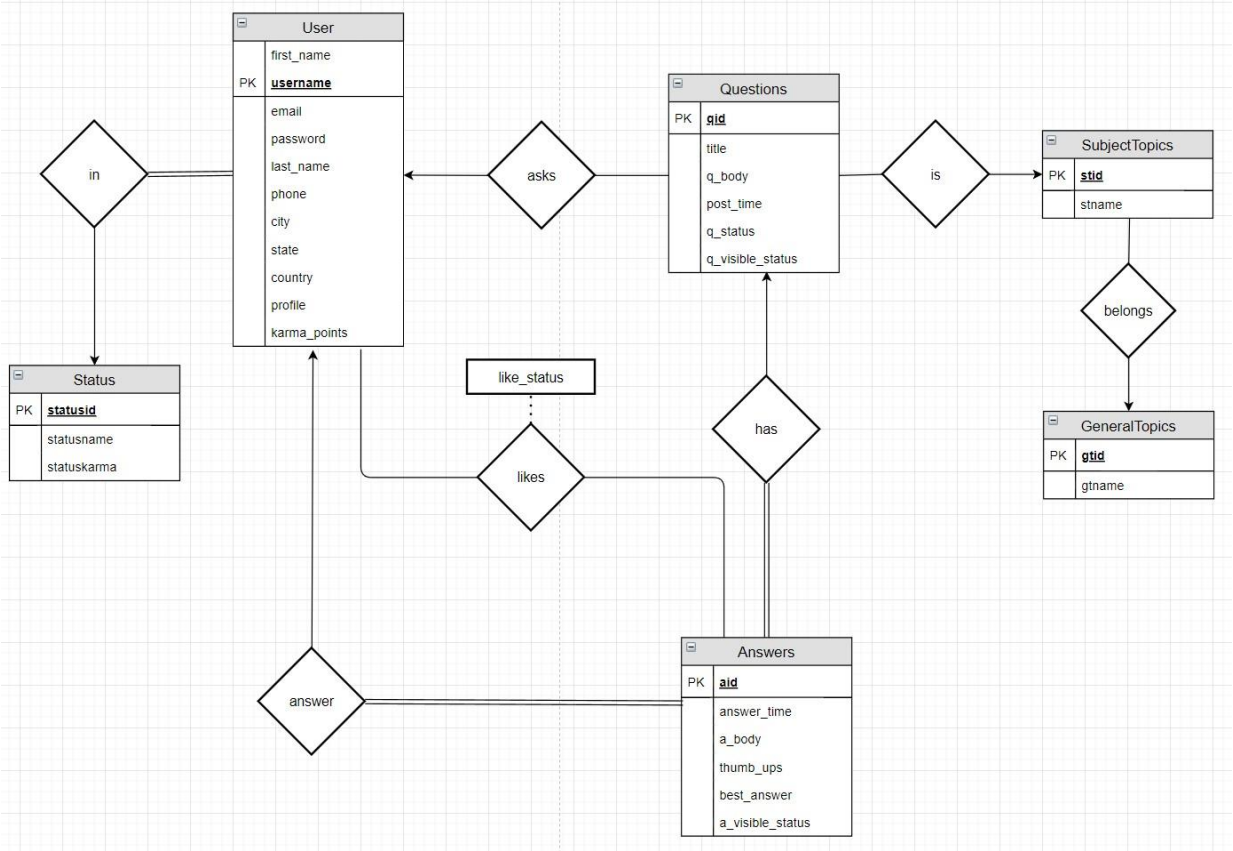
CS-GY 6083 Project Part 1 (Minghao Shao & Yiming Li)

1.1. ER Diagram



1.2. Relational Table Design

1. User Table

```
CREATE TABLE Users (
   username VARCHAR(45) NOT NULL,
   email VARCHAR(50),
   password VARCHAR(128) NOT NULL,
   firstname VARCHAR(45) NOT NULL,
   lastname VARCHAR(45) NOT NULL,
   phone VARCHAR(10),
   city VARCHAR(60),
   state VARCHAR(2),
   country VARCHAR(60),
   profile VARCHAR(512),
   karma_points INT NOT NULL default 0,
   PRIMARY KEY (username)
```

);

The data started from the user table, which was used to store all the information may required about the user. For the primary key, a unique username was used, also used for login the session, instead of AUTO_INCREMENT field. The username was designed and set by the user instead of generated by the system automatically each time a new account created. There were some benefits using this design, as a primary key, each time a user registering a new account, the system would check the uniqueness of this username, in which case the duplicate issue could be avoided since the system would check this, no manual action required. For the password field, instead of storing the password in plain text, SHA256 hashing function was used. In the test data, it was a simple SHA256 hashing, however, in the part2, more complex policy would be applied such as salting. Other information fields such as firstname, lastname, email, phone, city, state and country were designed to be nullable since sometimes user would prefer not to provide their personal information, hence this is not forced. Another field designed was the karma point. Which was used to identify the status of the user, regarding their level.

The policy regarding the karma point was according to the quantity and the quality of the answers the current user posted, say 10 points added each time the user posted an answer, and each time any answer the user posted received a "like", the user could also get 10 points bonuses. Another facet about earning the karma point was when the user's answer was selected by the question owner, the user who posted the best answer would also receive the bonuses, in test data, 20 points was set. Accordingly, like cancel was also taken into consideration, each time a user canceled the like, the 10 karma points bonuses would be dismissed also, when the best answer updated and the user's the answer was no longer the best answer, the 20 points bonuses would also be dismissed. The mechanism of question or answer deletion was also designed and would be discussed in Questions and Answers tables section respectively.

2. Status

```
CREATE TABLE Status (
   statusid INT AUTO_INCREMENT,
   statusname VARCHAR(20) NOT NULL,
   statuskarma INT NOT NULL,
   PRIMARY KEY (statusid)
);
```

A separate table called Status was also design, which was used to store the information of each level of the user, at current stage, three levels were designed, basic, advanced and expert. In this table, statusid was used as primary key which was AUTO_INCREMENT, also a status name, and the threshold called statuskarma, which stored the total karma point required for a user to reach this level. In this design, the basic level was from 0 to 50, advanced was from 50 to 100, and expert was larger then 100. Note that the numeric design was set only for the test data, to make some queries output the user with each level which could be optimized, for example, in part2, the threshold for each level would be increased, or the bonuses received by

the user would be deducted.

3. UserStatus

```
CREATE TABLE UserStatus (
   username VARCHAR(45) NOT NULL,
   statusid INT NOT NULL default 1,
   PRIMARY KEY (username),
   FOREIGN KEY (username) REFERENCES Users(username),
   FOREIGN KEY (statusid) REFERENCES Status(statusid)
);
```

The status of the user was not stored in Users table to satisfy the 3NF. Since the status of the user was totally depended on the karma points the user received, and the number of karma points received by the user was depended on the username. Hence, they were designed to divide to 2 separate table. The primary key was the username, which was enough to identity a user and the relation between the user and their status was 1 to many, which means a user could only have one status. From the statusid of each user, a join could be performed to get the exact status of the user from the status table.

4. GeneralTopics

```
CREATE TABLE GeneralTopics (
   gtid INT NOT NULL AUTO_INCREMENT,
   gtname VARCHAR(45) NOT NULL,
   PRIMARY KEY (gtid)
);
```

The structure of the topic was designed as hierarchy. Two levels were designed and the GeneralTopics table was the first layer. In this table, only the general topic could be stored such as MATH, PHYSICS or COMPUTER SCIENCE. In this table, a AUTO_INCREMENT field as primary key was design called gtid (general topic id), and another one was the name of the general topic.

The topics was aimed to be selected by a list box in the code implementation, there were several pre-defined topics, or the user may add topics on their own.

5. SubjectTopics

```
CREATE TABLE SubjectTopics (
   stid INT NOT NULL AUTO_INCREMENT,
   stname VARCHAR(50) NOT NULL,
   gtid INT NOT NULL,
   PRIMARY KEY (stid),
   FOREIGN KEY (gtid) REFERENCES GeneralTopics(gtid)
```

);

The next level of the topics was called SubjectTopics, in which the subjects were stored under their general topics. Similar as general topic, a stid (subject topic id) was design as the primary key, with AUTO_INCREMENT, also with the stname which was the name of the subject topic. Each subject topic should be able to reference to their parent topic, hence a foreign key was set which refenced to the gtid of their parent topic. Such as a topic called 'Database System', which was certainly under the COMPUTER SCIENCE general topic, hence the gtid of this subject topic would be the gtid of the general topic CS.

Also same as the general topic, in code implementation they were designed to be selected with a list box after the general topic was selected when posting a new question. Some pre-defined topics would also be provided by the system. Or users were able to add new subject topics.

6. Questions

```
CREATE TABLE Questions (
   qid INT NOT NULL AUTO_INCREMENT,
   q_username VARCHAR(45) NOT NULL,
   stid INT NOT NULL,
   title VARCHAR(45) NOT NULL,
   q_body VARCHAR(512) NOT NULL,
   post_time DATETIME NOT NULL,
   status VARCHAR(15) NOT NULL,
   q_visible_status INT NOT NULL default 1,
   PRIMARY KEY (qid),
   FOREIGN KEY (q_username) REFERENCES Users(username),
   FOREIGN KEY (stid) REFERENCES SubjectTopics(stid)
);
```

This table was designed to store the question information posted by the user. The primary key was called qid which was a AUTO_INCREMENT field with int type. For each question, the username who posted the question was also recorded into the field q_username. In consideration that users may either post questions, or give answers to questions, q_username was used instead of the username, to make the table clearer, but the q_username was still designed as a foreign key referenced to the username in Users table only the name of the field was different. In this design, general topic should not be used as the topic of the questions, but just to identity the subject topics which belonged to the general topic. Hence in the question table, a foreign key referenced to subject topic's stid field was used to identify the topic of the question. Also, the title and q_body field was used to store the title and the body of the question along with the post_time. In the test data, the post_time was inserted manually to show the test result and to simulate the real world condition, but the design for code implementation was always obtain the current time by the system by calling now().

The questions table also had a field called status, in this design, two statuses were defined, solved and unsolved, the user who posted the question could modify the status of the question. The status of the question would neither affect the karma points the user obtained nor affect how the question or the answers under this question would behave, but only a label which was used to make it convenient for the users who were also looking for the solution of similar questions. In in this, another situation was also considered for part2. Since in some cases, users would like to delete the questions they posted hence a field called a_visible_status was added to this table. Instead of deleting the question row directly, it would be safer to set the visible status of the question, if the question was not visible, then in the implementation in part2, it could also be regarded as deleted. Another benefit of using this field was that the user may be able to set the question as private/public in case they don't want others to see these questions (in case some privacy issue involved).

7. Answers

```
CREATE TABLE Answers (
   aid INT NOT NULL AUTO_INCREMENT,
   qid INT NOT NULL,
   answer_time DATETIME NOT NULL,
   a_username VARCHAR(45) NOT NULL,
   a_body VARCHAR(100) NOT NULL,
   thumb_ups INT NOT NULL,
   best_answer INT NOT NULL default 0,
   a_visible_status INT NOT NULL default 1,
   PRIMARY KEY (aid),
   FOREIGN KEY (qid) REFERENCES Questions(qid),
   FOREIGN KEY (a_username) REFERENCES Users (username)
);
```

This table was used to store all the answer posted by the users regarding the question. Same as most of strong entities in this design, aid was added as AUTO_INCREMENT primary key of this table. Also, there were two foreign keys, one was the qid references to question's id to obtain which question the current answer was belonged to. Another one was the username called a_username, with this field it was able to track the user who posted this answer, and each time the answer status changed, such as set to best answer or received a like, with this key it was also possible to provide the user with according karma points bonuses.

The table also designed with the answer's body and the answer_time which was the time the answer was posted, the likes that answer received. Another way to get the number of likes of the answer each time required such as loading the answer was to join the answer take with likes table and output the count value, in this design, a field recording the number of likes the answer received called thumb_ups was used instead with a trigger which would be discussed in the trigger section below because it was regarded as a more efficiency approach since the query of getting the count for each answer was no longer required with this field and the only operation

to do to handle a new update of like was just update or insert likes table, then the trigger would update the answer table automatically.

There was also one field called best_answer with value 0 and 1 was used to store whether the answer was selected as best answer. The best answer information was not stored in the question since the query clause could get the best answer of each question directly, since it was regarded as an attribute of the answer but not the question. Also, same as question table, the possibility of deleting the answer was taken into consideration by adding a field called a_visible_status. Same as questions table, if the user deleted this answer, instead deleting it directly, it was better to set the answer invisible.

8. Likes

```
CREATE TABLE Likes (
   username VARCHAR(45) NOT NULL,
   aid INT NOT NULL,
   like_status INT NOT NULL default 1,
   PRIMARY KEY (username, aid),
   FOREIGN KEY (username) REFERENCES Users (username),
   FOREIGN KEY (aid) REFERENCES Answers (aid)
);
```

In order to track the user who gave a like to an answer and make sure they were able to take the like back especially they gave a like by mistake, this table was added to track this information. Since for each unique answer, a user could only give a single thumb_up, hence username and aid was used as primary key. Another field added was the like_status, the logic was when a new like inserted (the user never gave a thumb_up to this answer), the record was inserted directly, while each time the user update a thumb_up, such as withdrawing a thumb_up or give the thumb_up back, the system would only update the like_status to 1 or 0.

1.3. Trigger Design

1. Likes

Likes table were designed with two triggers, an after insert trigger and an after update trigger

```
create trigger Likes_after_insert after insert on Likes for each row
begin
      update Answers
      set thumb_ups = thumb_ups + 1
      where aid = new.aid;
end;
```

Each time the a new thumb_up added by a user, the thumb_ups field in answers table would be

```
updated and added 1
create trigger Likes_after_update after update on Likes for each row
begin
    if new.like_status <> old.like_status then
        if new.like_status = 0 and old.like_status = 1 then
            update Answers
            set thumb_ups = thumb_ups - 1
            where new.aid = Answers.aid;
        elseif new.like_status = 1 and old.like_status = 0 then
            update Answers
            set thumb_ups = thumb_ups + 1
            where new.aid = Answers.aid;
        end if;
    end if;
end;
```

Each time the like_status was changed, there were two cases, the first case was a user withdraw a like, then the thumb_ups number in answers would decrease by 1, respectively, if a user gave a like back to the answer that was once withdrawn like from that user, the thumb_ups value in answers table would be added back, by 1.

2. Answers

Answers table was designed with two triggers, an after insert trigger and an after update trigger

```
create trigger Answers_after_insert after insert on Answers for each row
begin
    update Users
    set karma_points = karma_points + 10
    where Users.username = new.a_username;
end;
```

According to the design, each time a user posted a new answer, this user could obtain 10 karma points, even the answer was deleted. The Answers_after_insert trigger would implement this logic, each time a new answer inserted, which means a user posted a new answer, this user's karma point would add 10.

```
create trigger Answers_after_update after update on Answers for each row
begin
    if new.best_answer <> old.best_answer then
        if new.best_answer = 1 and old.best_answer = 0 then
            update Users
            set karma_points = karma_points + 20
            where Users.username = new.a_username;
```

```
        elseif new.best_answer = 0 and old.best_answer = 1 then
            update Users
            set karma_points = karma_points - 20
            where Users.username = new.a_username;
        end if;
    end if;

    if new.thumb_ups - old.thumb_ups <> 0 then
        update Users
        set karma_points = karma_points + 10 * (new.thumb_ups - old.thumb_ups)
        where Users.username = new.a_username;
    end if;
end;
```

There were two cases when updating the answers table. 1. The best answer status changed. In this design, each time an answer was selected as best answer, the user who posted this answer would obtain 20 karma points. Respectively, if an answer was no longer the best answer, the karma points of the user who posted the old best answer would be taken back which was -20 karma points. 2. The answer's thumb_ups changed. If the answer was deprived of a like, the karma points of the user who posted this answer would increase 10, respectively, if a new like were given, or the like was given back to this answer, for each like, the karma point of the user who posted this answer would decrease 10.

3. Users

Users table was designed with two triggers, an after insert trigger and an after update trigger

```
create trigger User_after_insert after insert on Users for each row
begin
    insert into UserStatus(username) values(new.username);
end;
```

Each time a new user account was created, the link to the status table would be added. Since for each new user, the userstatus was always 1 which was the basic level, hence as described in the schema design section, the default value for UserStatus was 1.

```
create trigger User_after_update after update on Users for each row
begin
    declare advance_thres INT;
    declare expert_thres INT;
    declare basic_status INT;
    declare acvanced_status INT;
    declare expert_status INT;
    if new.karma_points <> old.karma_points then
```

```
        select statusid into basic_status from status where statusname = 'basic';
        select statuskarma, statusid into advance_thres, acvanced_status from status where statusname = 'advanced';
        select statuskarma, statusid into expert_thres, expert_status from status where statusname = 'expert';
        if new.karma_points >= expert_thres and old.karma_points < expert_thres then
            update UserStatus
            set statusid = expert_status
            where UserStatus.username = new.username;
        elseif new.karma_points >= advance_thres and (old.karma_points < advance_thres or old.karma_points >= expert_thres) then
            update UserStatus
            set statusid = acvanced_status
            where UserStatus.username = new.username;
        elseif new.karma_points < advance_thres and old.karma_points >= advance_thres then
            update UserStatus
            set statusid = basic_status
            where UserStatus.username = new.username;
        end if;
    end if;
end;
```

Each time a user's karma point changed, the system would check the user's current status. The threshold of karma point of each status would be obtained from status table and the user's status in userstatus table would be updated according to the user's current karma point.

The default design for each level was: 0 – 50 was basic level, 50 – 100 was advanced level and karma points larger than 100 was expert level.

1.4. Sample Table Data

1. Users

username	email	password	firstname	lastname	phone	city	state	country	profile	karma_points
user1	user1@mail.com	42aff45bcfef506acda841c4c751a5e82376dc5e...	f1	l1	1112223333	New York	NY	USA	profile1	50
user2	user2@mail.com	956908ff8c5483d42acc0d332b89b86cf919f387...	f2	l2	2223334444	Boston	MA	USA	profile2	80
user3	user3@mail.com	42aff45bcfef506acda841c4c751a5e82376dc5e...	f3	l3	3334445555	Los Angles	CA	USA	profile3	120
user4	user4@mail.com	ed2047dfc9f23f56fe0c3af8996646b64a52b41d...	f4	l4	4445556666	New York	NY	USA	profile4	90
user5	user5@mail.com	64532608c19ae7a92a1cd9fb4f38dd29671955e...	f5	l5	5556667777	Boston	MA	USA	profile5	80
user6	user6@mail.com	be4b60459be9466dcaa973314cec00a3b97c2e4...	f6	l6	6667778888	Los Angles	CA	USA	profile6	10
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2. Status

statusid	statusname	statuskarma
1	basic	0
2	advanced	50
3	expert	100
NULL	NULL	NULL

3. Userstatus

username	statusid
user6	1
user1	2
user2	2
user4	2
user5	2
user3	3
NULL	NULL

3. Generaltopics

gtid	gtname
1	Computer Science
2	Physics
3	Math
4	Law
5	Music
6	Business
NULL	NULL

4. Subjecttopics

stid	stname	gtid
1	Database Systems	1
2	Computer Vision	1
3	Natural Language Processing	1
4	Astrophysics	2
5	Fusion and Plasma Physics	2
6	Nanoscience and Nanotechnology	2
7	Algebra	3
8	Combinatorics	3
9	Topology and Differential Geometry	3
10	Civil rights	4
11	Assisted suicide	4
12	Capital punishment	4
13	Composition	5
14	Songwriting	5
15	Performance	5
16	Film Scoring	5
17	Programming Languages	1
NULL	NULL	NULL

5. Questions

qid	q_username	stid	title	q_body	post_time	status	q_visible_status
1	user1	1	How to use select in mysql	As title stated, how to use it? I am a rookie of D...	2021-09-01 15:21:02	unsolved	1
2	user1	2	Good image datasets	Can anyone recommend some good datasets fo...	2020-01-03 00:05:03	resolved	1
3	user2	3	Some BERT familiy members	New starter for NLP, heard BERT is awesome, i...	2020-08-02 13:05:03	resolved	1
4	user2	4	What is a blackhole?	As title stated	2021-02-02 14:02:08	resolved	1
5	user2	5	Can anyone explain what is superconductivity?	I am interested in this topic hence I wonder can...	2021-03-02 12:02:08	resolved	1
6	user1	6	Some materials with nanoscience?	Can anyone take some examples for me with m...	2019-02-01 19:02:02	unsolved	1
7	user3	7	solve x + 1 = 2	Please solve the algebra problem for me!	2021-11-03 05:01:25	resolved	1
8	user3	8	sufficient condition	What is sufficient condition?	2022-01-04 02:01:26	resolved	1
9	user3	9	Seven Bridges of Konigsberg	What is this?	2022-01-05 04:02:26	unsolved	1
10	user4	10	Universal Declaration of Human Rights	When was it published?	2022-03-15 04:06:40	resolved	1
11	user4	11	suicide	Some examples for Assisted suicide?	2021-03-15 17:16:20	unsolved	1
12	user4	12	I am terrified	I stole some money, will I be sentenced to deat...	2020-01-15 12:16:21	unsolved	1
13	user5	13	great composer	I am preparing for my music exam, could you pl...	2020-11-15 12:13:20	resolved	1
14	user5	14	how to become good song writter	As title stated.	2020-01-17 12:13:20	resolved	1
15	user6	15	movie with great performance	I am looking for some movies best in performance	2019-12-25 19:11:34	unsolved	1
16	user6	16	Good film music	Could you please provide me with some classic ...	2022-04-03 12:14:24	resolved	1
17	user3	14	is songwriting hard to study?	i am a freshmen in college	2020-01-17 12:12:20	unsolved	1
18	user1	14	songwriting is so easy	i am a musician	2021-01-17 12:12:21	unsolved	1
19	user6	14	songwriting is so easy	i am also a musician	2021-01-17 12:12:22	unsolved	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

6. Answers

aid	qid	answer_time	a_username	a_body	thumb_ups	best_answer	a_visible_status
1	1	2021-09-01 17:29:02	user2	select attribute from table where condition!	0	0	1
2	2	2020-01-04 00:05:03	user3	MINST or CIFAR or MS-COCO were all good dat...	2	1	1
3	3	2020-08-02 18:02:01	user1	2 examples, ELMo, ERNIE	1	1	1
4	4	2021-02-04 13:02:08	user1	They just absorb everything	0	0	1
5	5	2021-05-02 12:02:08	user5	conductor with very very low resistance value	1	1	1
6	7	2021-11-03 08:01:25	user2	1	1	1	1
7	7	2021-11-04 08:01:25	user6	Super stupid question	0	0	1
8	8	2022-01-06 03:01:26	user2	conditions that will produce the event	2	0	1
9	10	2022-03-15 08:06:40	user3	1948-12-10	1	1	1
10	12	2020-01-15 15:16:21	user5	No... But you should be arrested	0	0	1
11	13	2020-11-16 11:13:20	user4	Beethoven or Mozart?	3	1	1
12	13	2020-11-18 11:14:20	user4	Taylor Swift! She will attend NYU graduation cel...	2	0	1
13	14	2020-01-18 12:13:20	user3	Work hard from now on...	0	1	1
14	15	2019-12-25 19:11:34	user5	I dont know, just post it in case I get karma poi...	0	0	1
15	16	2022-04-03 13:15:12	user5	Titanic and Star Wars Main Title!	1	0	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

7. Likes

Result Grid | Filter Rows:

username	aid	like_status
user1	2	1
user1	9	1
user1	12	1
user2	3	1
user2	11	1
user3	8	1
user3	11	1
user4	2	1
user4	5	1
user4	11	1
user5	8	1
user6	6	1
user6	12	1
user6	15	1
NULL	NULL	NULL

1.5. Test queries

1.

insert into Users (username, email, password, firstname, lastname, phone, city, state, country, profile, karma_points)
values ('user7', 'newuser@mail.com',
'8a56499fb55029bf22c416dcf2ff3dc0ef2049c4fa60bd9a629139e07e272152', 'f7', 'l7',
'0101010101', 'Jersey City', 'NJ', 'USA', 'profile', 0);

username	email	password	firstname	lastname	phone	city	state	country	profile	karma_points
user1	user1@mail.com	42aff45bcfef506acda841c4c751a5e82376dc5e...	f1	l1	1112223333	New York	NY	USA	profile1	50
user2	user2@mail.com	956908ff8c5483d42acc0d332b89b86cf919f387...	f2	l2	2223334444	Boston	MA	USA	profile2	80
user3	user3@mail.com	42aff45bcfef506acda841c4c751a5e82376dc5e...	f3	l3	3334445555	Los Angles	CA	USA	profile3	120
user4	user4@mail.com	ed2047dfc9f23f56fe0c3af8996646b64a52b41d...	f4	l4	4445556666	New York	NY	USA	profile4	90
user5	user5@mail.com	64532608c19ae7a92a1cd9fb4f38dd29671955e...	f5	l5	5556667777	Boston	MA	USA	profile5	80
user6	user6@mail.com	be4b60459be9466dcaa973314cec00a3b97c2e4...	f6	l6	6667778888	Los Angles	CA	USA	profile6	10
user7	newuser@mail.com	8a56499fb55029bf22c416dcf2ff3dc0ef2049c4f...	f7	l7	0101010101	Jersey City	NJ	USA	profile	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2.

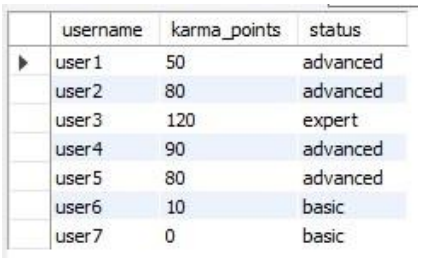
```
insert into Questions (q_username, stid, title, q_body, post_time, status)
values ('user4', 11, 'suicide', 'What is suicide?', '2021-03-13 17:11:15', 'unsolved');
```

qid	q_username	stid	title	q_body	post_time	status	q_visible_status
1	user1	1	How to use select in mysql	As title stated, how to use it? I am a rookie of D...	2021-09-01 15:21:02	unsolved	1
2	user1	2	Good image datasets	Can anyone recommend some good datasets fo...	2020-01-03 00:05:03	resolved	1
3	user2	3	Some BERT familiy members	New starter for NLP, heard BERT is awesome, i...	2020-08-02 13:05:03	resolved	1
4	user2	4	What is a blackhole?	As title stated	2021-02-02 14:02:08	resolved	1
5	user2	5	Can anyone explain what is superconductivity?	I am interested in this topic hence I wonder can...	2021-03-02 12:02:08	resolved	1
6	user1	6	Some materials with nanoscience?	Can anyone take some examples for me with m...	2019-02-01 19:02:02	unsolved	1
7	user3	7	solve x + 1 = 2	Please solve the algebra problem for me!	2021-11-03 05:01:25	resolved	1
8	user3	8	sufficient condition	What is sufficient condition?	2022-01-04 02:01:26	resolved	1
9	user3	9	Seven Bridges of Konigsberg	What is this?	2022-01-05 04:02:26	unsolved	1
10	user4	10	Universal Declaration of Human Rights	When was it published?	2022-03-15 04:06:40	resolved	1
11	user4	11	suicide	Some examples for Assisted suicide?	2021-03-15 17:16:20	unsolved	1
12	user4	12	I am terrified	I stole some money, will I be sentenced to deat...	2020-01-15 12:16:21	unsolved	1
13	user5	13	great composer	I am preparing for my music exam, could you pl...	2020-11-15 12:13:20	resolved	1
14	user5	14	how to become good song writter	As title stated.	2020-01-17 12:13:20	resolved	1
15	user6	15	movie with great performance	I am looking for some movies best in performance	2019-12-25 19:11:34	unsolved	1
16	user6	16	Good film music	Could you please provide me with some classic ...	2022-04-03 12:14:24	resolved	1
17	user3	14	is songwriting hard to study?	i am a freshmen in college	2020-01-17 12:12:20	unsolved	1
18	user1	14	songwriting is so easy	i am a musician	2021-01-17 12:12:21	unsolved	1
19	user6	14	songwriting is so easy	i am also a musician	2021-01-17 12:12:22	unsolved	1
20	user4	11	suicide	What is suicide?	2021-03-13 17:11:15	unsolved	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

3.

```
DELIMITER //
create procedure Get_Karma()
begin
	declare advance_thres INT;
	declare expert_thres INT;
	 select statuskarma into advance_thres from status where statusname = 'advanced';
	select statuskarma into expert_thres from status where statusname = 'expert';
	select username, karma_points,
	case
		when karma_points < advance_thres then 'basic'
		when karma_points < expert_thres then 'advanced'
		else 'expert'
	end as status
from (select username, sum(score) as karma_points from (
	select users.username, 10 * count(*) as score from users left join answers on
answers.a_username = users.username inner join likes on likes.aid = answers.aid where
likes.like_status = 1 group by users.username
	union all
	 select username, 20 * count(aid) as score from users left join answers on
answers.a_username = users.username where best_answer = 1 group by username
	 union all
	 select username, 10 * count(aid) as score from users left join answers on
answers.a_username = users.username group by username
) as s group by username) as l;
```

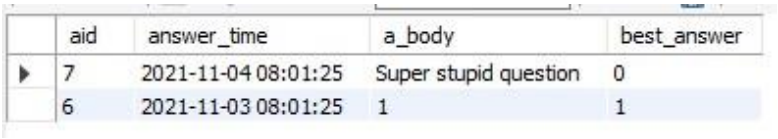
```
end //
call Get_Karma();
DROP PROCEDURE Get_Karma;
```



username	karma_points	status
user1	50	advanced
user2	80	advanced
user3	120	expert
user4	90	advanced
user5	80	advanced
user6	10	basic
user7	0	basic

4.

select aid, answer_time, a_body, best_answer from Questions natural join Answers where qid = 7 order by answer_time desc



aid	answer_time	a_body	best_answer
7	2021-11-04 08:01:25	Super stupid question	0
6	2021-11-03 08:01:25	1	1

5.

```
select GeneralTopics.gtid, stid as stid, IFNULL(s.question_count, 0) as question_count,
IFNULL(s.answer_count, 0) as answer_count
from GeneralTopics left join (
    select SubjectTopics.gtid, SubjectTopics.stid, question_count, answer_count
    from SubjectTopics left join (
        select q.stid, question_count, answer_count
        from (
            select stid, count(*) as question_count
            from Questions
            group by stid
        ) as q left join (
            select stid, count(*) as answer_count
            from Questions natural join Answers
            group by stid
        ) as a
        on q.stid = a.stid
    ) as c
    on SubjectTopics.stid = c.stid
) as s
on GeneralTopics.gtid = s.gtid
```

gtid	stid	question_count	answer_count
1	1	1	1
1	2	1	1
1	3	1	1
1	17	0	0
2	4	1	1
2	5	1	1
2	6	1	0
3	7	1	2
3	8	1	1
3	9	1	0
4	10	1	1
4	11	2	0
4	12	1	1
5	13	1	2
5	14	4	1
5	15	1	1
5	16	1	1
6	NULL	0	0

6.

```
select qid, title, q_body, sum(weight), timestamp from(
select qid, q_username as username, title, q_body, post_time as timestamp, 0.3 as weight
from Questions
where stid = 14 and title like '%e%'

union all

select qid, q_username as username, title, q_body, post_time as timestamp, 0.6 as weight
from Questions
where stid = 14 and q_body like '%e%'

union all

select qid, q_username as username, title, q_body, answer_time as timestamp, 0.1 as weight
from Questions natural join Answers
where stid = 14 and a_body like '%e%'
) as q group by qid, title, q_body, timestamp order by sum(weight) desc, timestamp desc
```

qid	title	q_body	sum(weight)	timestamp
14	how to become good song writter	As title stated.	0.9	2020-01-17 12:13:20
17	is songwriting hard to study?	i am a freshmen in college	0.6	2020-01-17 12:12:20
19	songwriting is so easy	i am also a musician	0.3	2021-01-17 12:12:22
18	songwriting is so easy	i am a musician	0.3	2021-01-17 12:12:21

1.6. Another Design

1. About query design

In this design, three aspects were taken into consideration and weights were given regarding each aspect. Given a query for a specific topic, first, if the key words existed in the question

title, 0.3 points would be given, then second, if the key words existed in the question body, 0.6 points would be given, last, for each answer 0.1 * number of answers points obtained with these key words would be given. In the end the total score regarding these three aspects would be calculated by summing up and the final result was ordered mainly by the relevance score obtained. For questions with same relevance score, the it would be ordered by the time_posted of the question, from newer to older. In this design, karma points, or status was not considered as the facet of question ranking since sometimes it was not fair to the users with lower karma points or level. To make the ranking standard clear, the formula below could be referenced:

score = 0.1 * number_of_answers + 0.3 if key words in title + 0.6 if key words in body

Ranked by score desc and post_time desc