Principles of Database Systems
Project Report

Group Members

Name	Net id
Sakshi Garg	sdg8172
Ritik LNU	rl4017

Table of Content

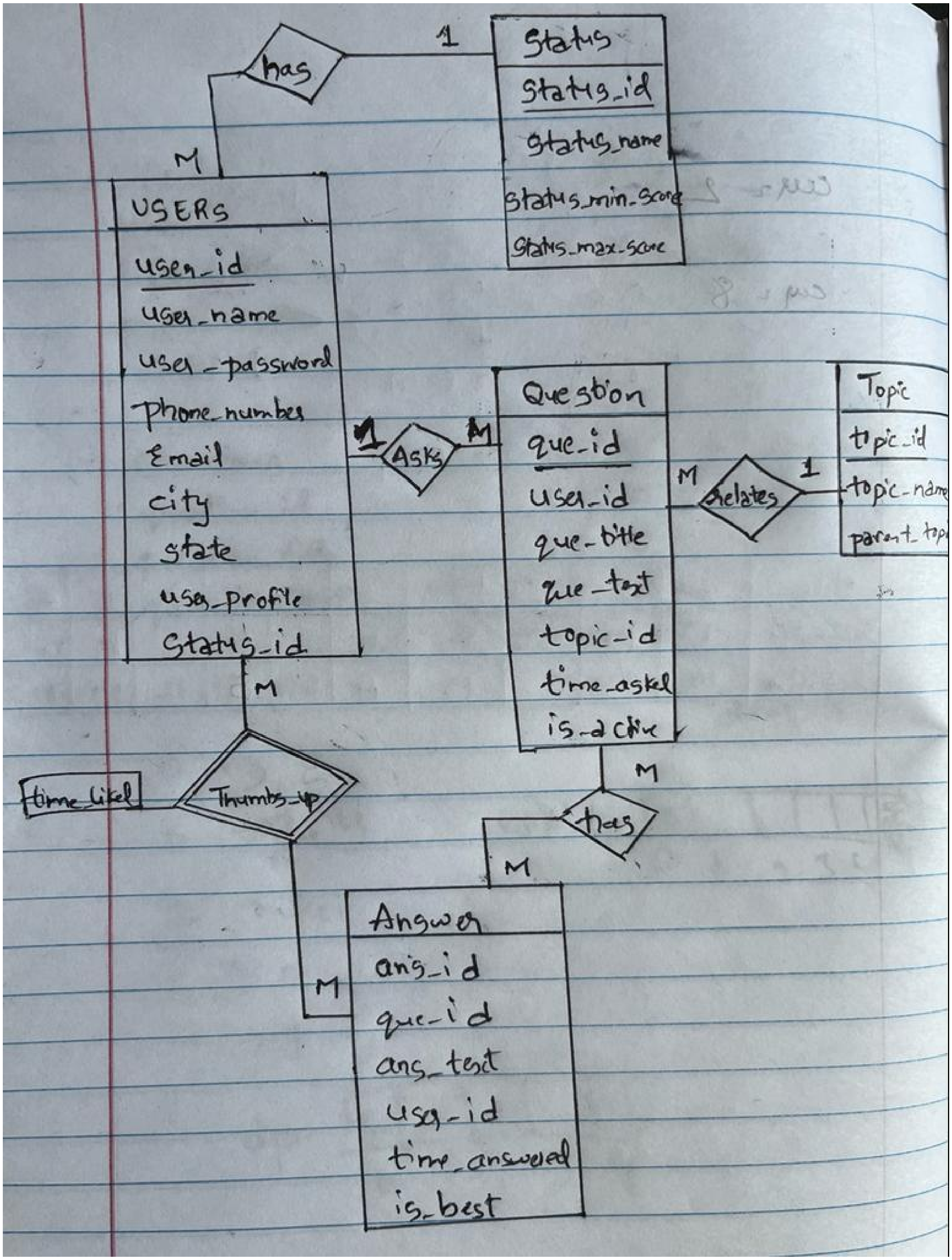
1. Introduction
2. ER Diagram
3. Relational Schema
4. Assumptions
5. Create Table Statements
6. SQL Queries
7. Populated Tables
8. Site Features
    a. Registration
    b. Login
    c. Main Page
    d. Search
    e. Topics
    f. User Profile
        i. Updating User Status
        ii. Event Scheduler
    g. Signout
    h. Post a Question
    i. View Selected Question

Introduction

We have designed the database for the question/answering platform in the following manner -

1. We have created a users table which would store the information related to all the users adding questions or answers to the platform.
2. When a user asks a question they can also add in a topic tag by selecting from the already available list of parent topics and sub topics. We have used another table Topic to store this information. There is a column called parent_topic_id where the parent topic id is stored, in the case of a parent topic the value in this column is null.
3. The questions that are asked by the user along with the topic id associated with them are stored in the questions table.
4. The answers to these questions are stored in the answers table where to store the best answer we have created another column called is_best which stores 0, 1 as values. Only the user who has asked the question will be able to choose an answer as the best answer. The other users will still be able to give a thumbs up to the answers posted.
5. The thumbs up that users give to an answer are stored in a table called thumbs_up where the user_id the ans_id and the datetime of all the thumbs ups are stored.
6. All the possible statuses of the user like beginner, advance and expert are stored in the status table along with the score range.
7. A procedure will run to update the status/score of these users regularly.

ER Diagram



has
1
M
Status
Status_id
Status_name
Status_min_Score
Status_max_Score
USERS
user_id
user_name
user_password
phone_number
Email
city
state
user_profile
Status_id
1
Asks
M
Question
que_id
user_id
que_title
que_text
topic_id
time_asked
is_active
M
Relates
1
Topic
topic_id
topic_name
parent_topi
M
Thumbs_up
time_liked
M
has
M
Answer
ans_id
que_id
ans_text
user_id
time_answered
is_best
M

Relational Schema

Primary keys are underlined and highlighted in red

Users (user_id, user_name, user_password, phone_number, email, city, state, user_profile, status_id)
Foreign Key: status_id references status(status_id)

Question (que_id, user_id, que_title, que_text, topic_id, time_asked, is_active)
Foreign Key: user_id references User(user_id)
topic_id references Topic(topic_id)

Topic (topic_id, topic_name, parent_topic_id)

Answer (ans_id, que_id, ans_text, user_id, time_answered, is_best)
Foreign Key: que_id references Question(que_id)
user_id references User(user_id)

Thumbs_up (ans_id, user_id, time_liked)
Foreign Key: ans_id references answer(ans_id)
user_id references User(user_id)

Status (status_id, status_name, status_min_score,status_max_score)

Assumptions

1. Created a separate table to store thumbs_up, so the thumbs_up of every user who has liked an answer and the time that they have liked the answer can be stored.
2. The three different statuses beginner, advanced & Expert will be stored in the Status table.
3. Only the user who has asked the question will be able to select the best answer - This will be taken care of using PHP by making the option available only to the user who has asked the question.
4. The parameters that will define the status of each user would be -
   Scores can be calculated by giving weightage to questions & answers posted in a particular time frame -

   a. (0.25 * No of questions + answers posted in last 6 months + (0.5 * (before 6 months)+ No of best answers)

   Expert - Score > 50
   Advanced - 20 < Score <= 50
   Beginner - 0 < Score <= 20
5. Keyword search query uses the FULL TEXT SEARCH using the context indexes created to perform this operation. The indexes are created on que_text column in question table and ans_text column in answer table. Using these indexes improves the functionality of searching over LIKE.
   The weightage is divided into :
   a. if the keyword is matched using LIKE keyword in the question_title it scores 10.
   b. if the keyword is matched on que_text column in question table using contains after indexes the score produced by the function is the score for the question.
   c. if the keyword is matched on ans_text column in answer table using contains after indexes the score produced by the function is the score for the question.

   Total weightage is then calculated for each criteria above and results are order by final_score.

Create Table Statements

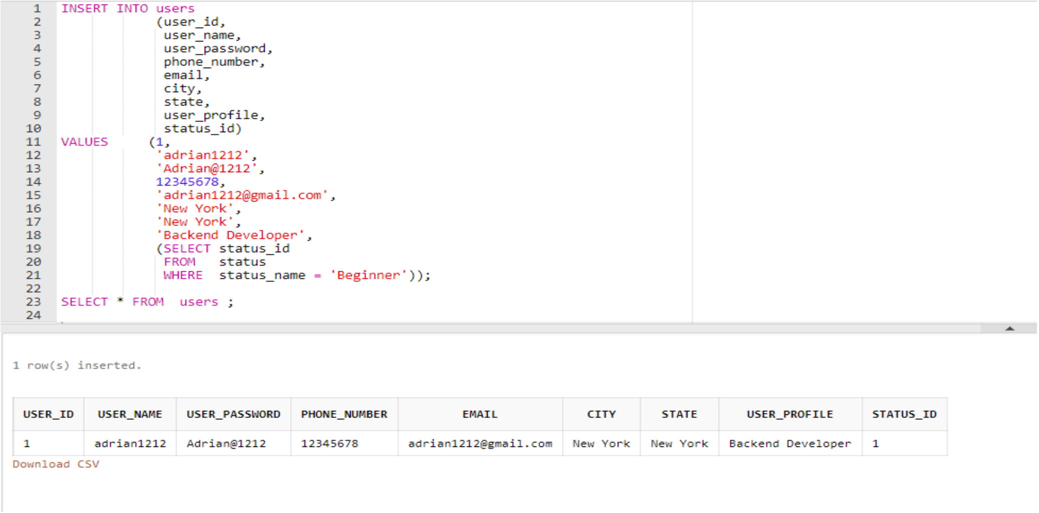
```
CREATE TABLE topic
  (
     topic_id        INTEGER PRIMARY KEY,
     topic_name      VARCHAR(20) NOT NULL,
     parent_topic_id INTEGER
  );

CREATE TABLE status
  (
     status_id        INTEGER PRIMARY KEY,
     status_name      VARCHAR(20) NOT NULL,
     status_min_score INTEGER NOT NULL,
     status_max_score INTEGER
  );

CREATE TABLE users
  (
     user_id       INTEGER PRIMARY KEY,
     user_name     VARCHAR2(20) NOT NULL,
     user_password VARCHAR2(20) NOT NULL,
     phone_number  INTEGER NOT NULL,
     email         VARCHAR2(30),
     city          VARCHAR2(20),
     state         VARCHAR2(20),
     user_profile  VARCHAR2(50),
     status_id     INTEGER NOT NULL,
     CONSTRAINT unique_user_name UNIQUE (user_name),
     FOREIGN KEY (status_id) REFERENCES status(status_id)
  );

CREATE TABLE question
  (
     que_id      INTEGER PRIMARY KEY,
     user_id     INTEGER NOT NULL,
     que_title   VARCHAR(100) NOT NULL,
     que_text    VARCHAR(200) NOT NULL,
      topic_id   INTEGER NOT NULL,
      time_asked TIMESTAMP NOT NULL,
      is_active  INTEGER NOT NULL,
      FOREIGN KEY (user_id) REFERENCES users(user_id),
      FOREIGN KEY (topic_id) REFERENCES topic(topic_id)
   );

CREATE TABLE answer
  (
     ans_id        INTEGER PRIMARY KEY,
     que_id        INTEGER NOT NULL,
     ans_text      varchar2(4000) NOT NULL,
     user_id       INTEGER NOT NULL,
     time_answered TIMESTAMP NOT NULL,
     is_best       NUMBER(1, 0) NOT NULL,
     FOREIGN KEY (user_id) REFERENCES users(user_id),
     FOREIGN KEY (que_id) REFERENCES question(que_id)
  );

CREATE TABLE thumbs_up
  (
     ans_id     INTEGER NOT NULL,
     user_id    INTEGER NOT NULL,
     time_liked TIMESTAMP NOT NULL,
     FOREIGN KEY (user_id) REFERENCES users(user_id),
     FOREIGN KEY (ans_id) REFERENCES answer(ans_id)
  );
```

SQL Queries

1.

```
INSERT INTO users
            (user_id,
             user_name,
             user_password,
             phone_number,
             email,
             city,
             state,
             user_profile,
             status_id)
VALUES     (1,
            'adrian1212',
            'Adrian@1212',
            12345678,
            'adrian1212@gmail.com',
            'New York',
            'New York',
            'Backend Developer',
            (SELECT status_id
             FROM   status
             WHERE  status_name = 'Beginner'));
```

RESULT :



```
INSERT INTO users
            (user_id,
             user_name,
             user_password,
             phone_number,
             email,
             city,
             state,
             user_profile,
             status_id)
VALUES     (1,
            'adrian1212',
            'Adrian@1212',
            12345678,
            'adrian1212@gmail.com',
            'New York',
            'New York',
            'Backend Developer',
            (SELECT status_id
             FROM   status
             WHERE  status_name = 'Beginner'));

SELECT * FROM  users ;
```

```
1 row(s) inserted.
```

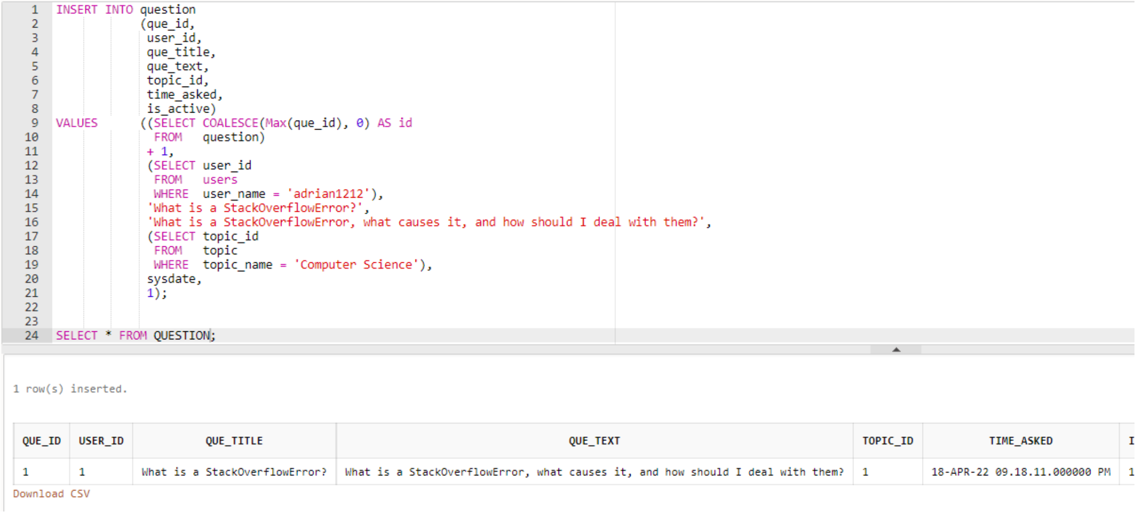
USER_ID	USER_NAME	USER_PASSWORD	PHONE_NUMBER	EMAIL	CITY	STATE	USER_PROFILE	STATUS_ID
1	adrian1212	Adrian@1212	12345678	adrian1212@gmail.com	New York	New York	Backend Developer	1

Download CSV

2.

```
INSERT INTO question
            (que_id,
             user_id,
             que_title,
             que_text,
             topic_id,
             time_asked,
             is_active)
VALUES      ((SELECT COALESCE(Max(que_id), 0) AS id
              FROM   question)
             + 1,
             (SELECT user_id
              FROM   users
              WHERE  user_name = 'adrian1212'),
             'What is a StackOverflowError?',
             'What is a StackOverflowError, what causes it, and
how should I deal with them?',
             (SELECT topic_id
              FROM   topic
              WHERE  topic_name = 'Computer Science'),
             sysdate,
             1);
```

RESULT:



```
INSERT INTO question
            (que_id,
             user_id,
             que_title,
             que_text,
             topic_id,
             time_asked,
             is_active)
VALUES      ((SELECT COALESCE(Max(que_id), 0) AS id
              FROM   question)
             + 1,
             (SELECT user_id
              FROM   users
              WHERE  user_name = 'adrian1212'),
             'What is a StackOverflowError?',
             'What is a StackOverflowError, what causes it, and how should I deal with them?',
             (SELECT topic_id
              FROM   topic
              WHERE  topic_name = 'Computer Science'),
             sysdate,
             1);


SELECT * FROM QUESTION;
```

```
1 row(s) inserted.
```

QUE_ID	USER_ID	QUE_TITLE	QUE_TEXT	TOPIC_ID	TIME_ASKED	I
1	1	What is a StackOverflowError?	What is a StackOverflowError, what causes it, and how should I deal with them?	1	18-APR-22 09.18.11.000000 PM	1

3.

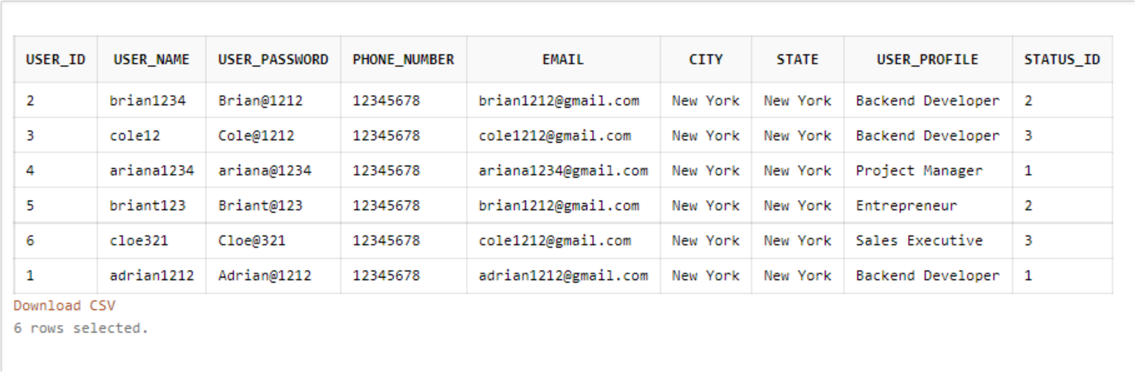
```
//PROCEDURE TO UPDATE USER STATUS

CREATE OR REPLACE PROCEDURE update_user_status IS
BEGIN
            MERGE INTO users t1
        USING
        (
            with ques as(
        SELECT user_id
              ,COUNT(que_id) as no_of_ques
        FROM question
        GROUP BY user_id
        ),
        recent_ans as(
        SELECT user_id
              ,COUNT(ans_id) as no_of_recent_ans
        FROM answer
        WHERE MONTHS_BETWEEN(CAST(sysdate AS
DATE),CAST(time_answered AS DATE))<6
        GROUP BY user_id
        ),
        old_ans as(
        SELECT user_id
              ,COUNT(ans_id) as no_of_old_ans
        FROM answer
        WHERE MONTHS_BETWEEN(CAST(sysdate AS
DATE),CAST(time_answered AS DATE))>=6
        GROUP BY user_id
        ),
        best_ans as(
        SELECT user_id
              ,COUNT(ans_id) as no_of_best_ans
        FROM answer
        WHERE is_best= 1
        GROUP BY user_id
        ),
        total_score as(
        SELECT users.user_id
              ,0.25*(COALESCE(ques.no_of_ques,0)) +
COALESCE(recent_ans.no_of_recent_ans,0) + 0.75 *
(COALESCE(old_ans.no_of_old_ans,0)) +
COALESCE(best_ans.no_of_best_ans,0) as total_score
        FROM  users
        LEFT JOIN  ques
        on   users.user_id = ques.user_id
        LEFT JOIN  recent_ans
        on   users.user_id = recent_ans.user_id
        LEFT JOIN  old_ans
        on   users.user_id = old_ans.user_id
        LEFT JOIN  best_ans
        on   users.user_id = best_ans.user_id
        ),
        user_status as(
        SELECT total_score.user_id
              ,status.status_id
        FROM total_score
        CROSS JOIN status
        WHERE total_score.total_score >=
status.status_min_score
        AND   total_score.total_score <
COALESCE(status.status_max_score, 2147483647)
        ),
        changed_user_status as(
        SELECT u.user_id
              , us.status_id AS changed_status_id
        FROM users u
        LEFT JOIN  user_status us
        ON u.user_id = us.user_id
        WHERE u.status_id <> us.status_id
        )
        SELECT * FROM changed_user_status
        )t2
        ON(t1.user_id = t2.user_id)
        WHEN MATCHED THEN UPDATE SET
        t1.status_id = t2.changed_status_id;
```

```
END;
```

RESULT :

BEFORE RUNNING THE ABOVE QUERY/PROCEDURE :

USERS TABLE:



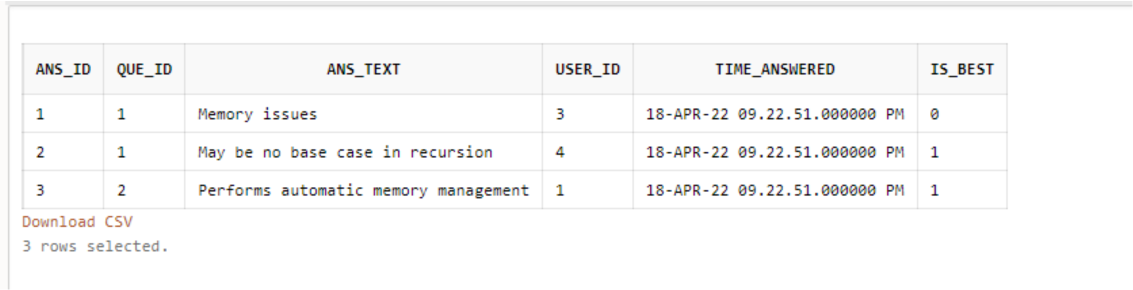
USER_ID	USER_NAME	USER_PASSWORD	PHONE_NUMBER	EMAIL	CITY	STATE	USER_PROFILE	STATUS_ID
2	brian1234	Brian@1212	12345678	brian1212@gmail.com	New York	New York	Backend Developer	2
3	cole12	Cole@1212	12345678	cole1212@gmail.com	New York	New York	Backend Developer	3
4	ariana1234	ariana@1234	12345678	ariana1234@gmail.com	New York	New York	Project Manager	1
5	briant123	Briant@123	12345678	brian1212@gmail.com	New York	New York	Entrepreneur	2
6	cloe321	Cloe@321	12345678	cole1212@gmail.com	New York	New York	Sales Executive	3
1	adrian1212	Adrian@1212	12345678	adrian1212@gmail.com	New York	New York	Backend Developer	1

Download CSV

6 rows selected.

We see that there are users with status id above 1 that is more than basic(above a score of 20).

ANSWER TABLE:



ANS_ID	QUE_ID	ANS_TEXT	USER_ID	TIME_ANSWERED	IS_BEST
1	1	Memory issues	3	18-APR-22 09.22.51.000000 PM	0
2	1	May be no base case in recursion	4	18-APR-22 09.22.51.000000 PM	1
3	2	Performs automatic memory management	1	18-APR-22 09.22.51.000000 PM	1

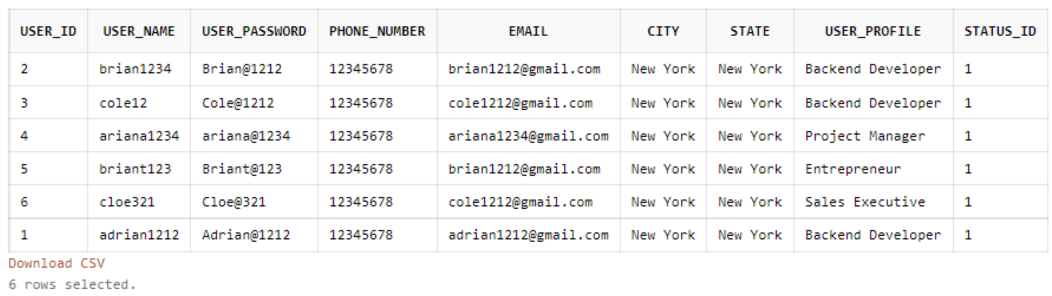
Download CSV

3 rows selected.

THE answer table consists of very few answers which would actually mean that no users should have a score above 20 based on our defined calculations for status.

AFTER RUNNING THE QUERY/PROCEDURE:

USERS TABLE:



USER_ID	USER_NAME	USER_PASSWORD	PHONE_NUMBER	EMAIL	CITY	STATE	USER_PROFILE	STATUS_ID
2	brian1234	Brian@1212	12345678	brian1212@gmail.com	New York	New York	Backend Developer	1
3	cole12	Cole@1212	12345678	cole1212@gmail.com	New York	New York	Backend Developer	1
4	ariana1234	ariana@1234	12345678	ariana1234@gmail.com	New York	New York	Project Manager	1
5	briant123	Briant@123	12345678	brian1212@gmail.com	New York	New York	Entrepreneur	1
6	cloe321	Cloe@321	12345678	cole1212@gmail.com	New York	New York	Sales Executive	1
1	adrian1212	Adrian@1212	12345678	adrian1212@gmail.com	New York	New York	Backend Developer	1

Download CSV

6 rows selected.

All users have status updated to basic.

We run the above procedure daily at night to update the status of users whose status have changed based on answering and our calculations.

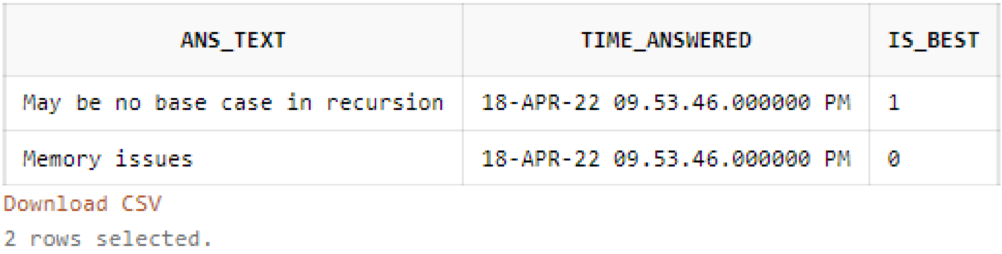
We are doing this using a job scheduler by oracle as mentioned below.

```
// SCHEDULING A DAILY JOB TO RUN ABOVE PROCEDURE
BEGIN
  DBMS_SCHEDULER.create_job (
    job_name        => 'Update user status',
    job_type        => 'PLSQL_BLOCK',
    job_action      => 'BEGIN update_user_status; END;',
    start_date      => SYSTIMESTAMP,
    repeat_interval => 'freq=daily',
    end_date        => NULL,
    enabled         => TRUE,
    comments        => 'This job does update user_status on a daily basis'
  );
END;
```

4.

```
SELECT ans_text,
       time_answered,
       is_best
FROM   answer
WHERE  que_id = 1
ORDER  BY time_answered ASC;
```

RESULT:



ANS_TEXT	TIME_ANSWERED	IS_BEST
May be no base case in recursion	18-APR-22 09.53.46.000000 PM	1
Memory issues	18-APR-22 09.53.46.000000 PM	0

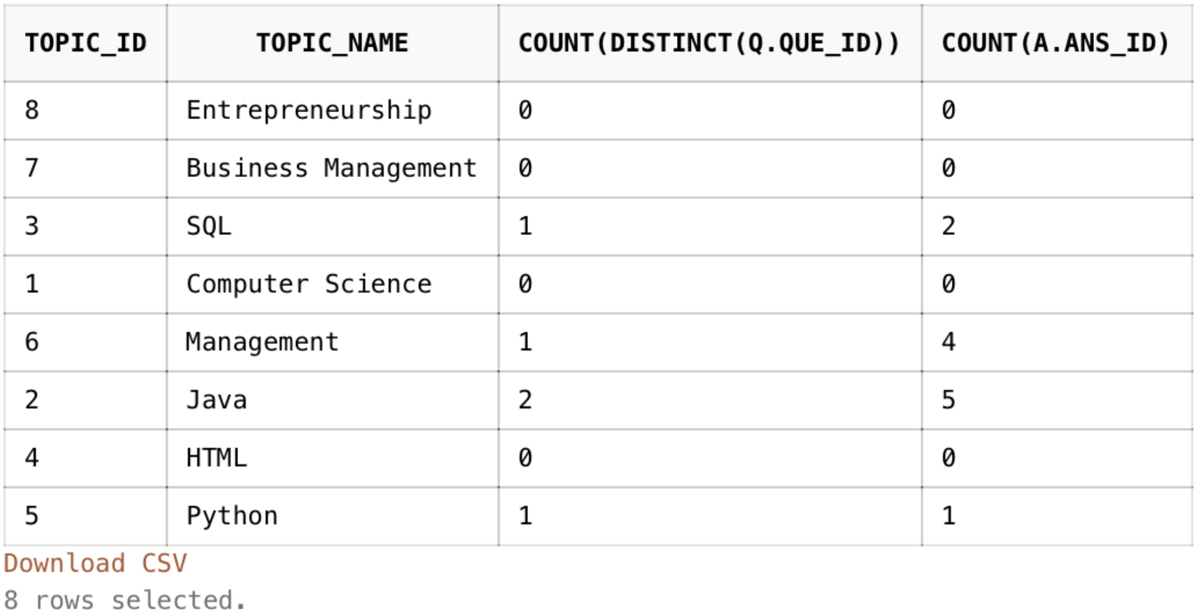
Download CSV

2 rows selected.

5.

```
SELECT t.topic_id,
       t.topic_name,
       Count(q.que_id),
       Count(a.ans_id)
FROM   topic t
       LEFT JOIN question q
              ON t.topic_id = q.topic_id
       LEFT JOIN answer a
              ON a.que_id = q.que_id
GROUP  BY t.topic_id,
          t.topic_name;
```

RESULT:



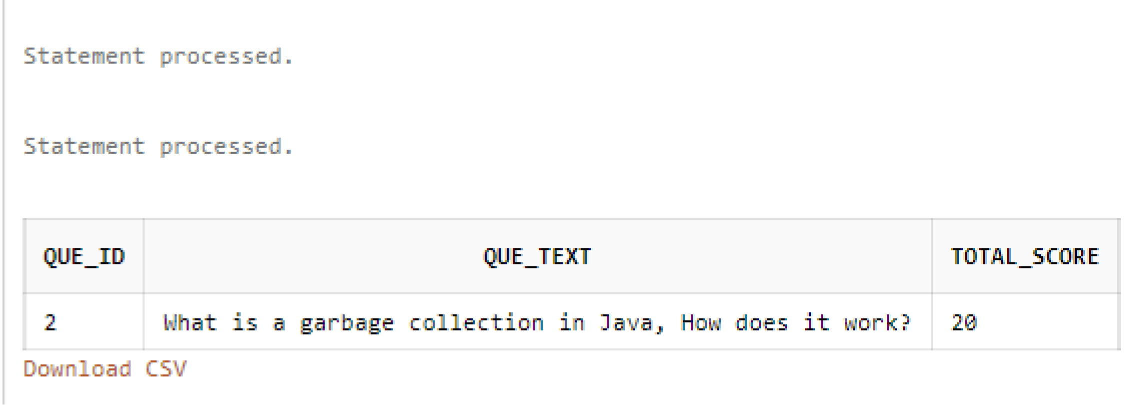
TOPIC_ID	TOPIC_NAME	COUNT(DISTINCT(Q.QUE_ID))	COUNT(A.ANS_ID)
8	Entrepreneurship	0	0
7	Business Management	0	0
3	SQL	1	2
1	Computer Science	0	0
6	Management	1	4
2	Java	2	5
4	HTML	0	0
5	Python	1	1

Download CSV
8 rows selected.

6.

```
CREATE INDEX question_idx ON
question(que_text)
INDEXTYPE IS CTXSYS.CONTEXT;

CREATE INDEX answer_idx ON
answer(ans_text)
INDEXTYPE IS CTXSYS.CONTEXT;


//SYNCHRONIZATION OF INDEX before each time the query runs
exec ctx_ddl.sync_index('question_idx');
exec ctx_ddl.sync_index('answer_idx');

//query to output relevant questions

//Given Values :search query keyword and topic_id
//In below case we used (search keyword ='garbage' and topic_id
= 2)

with que_title_score as(
    select que.que_id ,que.que_text, 10 as score
    FROM QUESTION QUE
    JOIN TOPIC    TOP
    ON   QUE.TOPIC_ID = TOP.TOPIC_ID
    WHERE QUE.QUE_TITLE LIKE '%garbage%'
    AND   TOP.TOPIC_ID = 2
),que_text_score as(
    select QUE.que_id,que_text,score(10) as score
    FROM QUESTION QUE
    JOIN TOPIC    TOP
    ON   QUE.TOPIC_ID = TOP.TOPIC_ID
    WHERE CONTAINS (QUE.que_text,'garbage',10)>0
    AND   TOP.TOPIC_ID = 2
),

ans_text_score as(
    select distinct QUE.que_id,que_text,score(10) as score
    FROM ANSWER ANS
    JOIN QUESTION QUE
    ON ANS.QUE_ID = QUE.QUE_ID

    JOIN TOPIC TOP
    ON TOP.TOPIC_ID = QUE.TOPIC_ID
    WHERE CONTAINS (ANS.ans_text,'garbage',10)>0
    AND TOP.TOPIC_ID = 2
)
SELECT
    COALESCE(q_title.que_id,q_text.que_id,a_text.que_id)
as que_id
    ,COALESCE(q_title.que_text,q_text.que_text,a_text.que_text)
as que_text
    ,(COALESCE( q_title.score ,0) +  COALESCE( q_text.score ,0)
+ COALESCE( a_text.score ,0)) as total_score

FROM que_title_score            q_title
FULL OUTER JOIN que_text_score  q_text
ON q_title.que_id = q_text.que_id

FULL OUTER JOIN ans_text_score  a_text
ON a_text.que_id = COALESCE(q_title.que_id,q_text.que_id)
Order by  total_score desc;
```

RESULT :



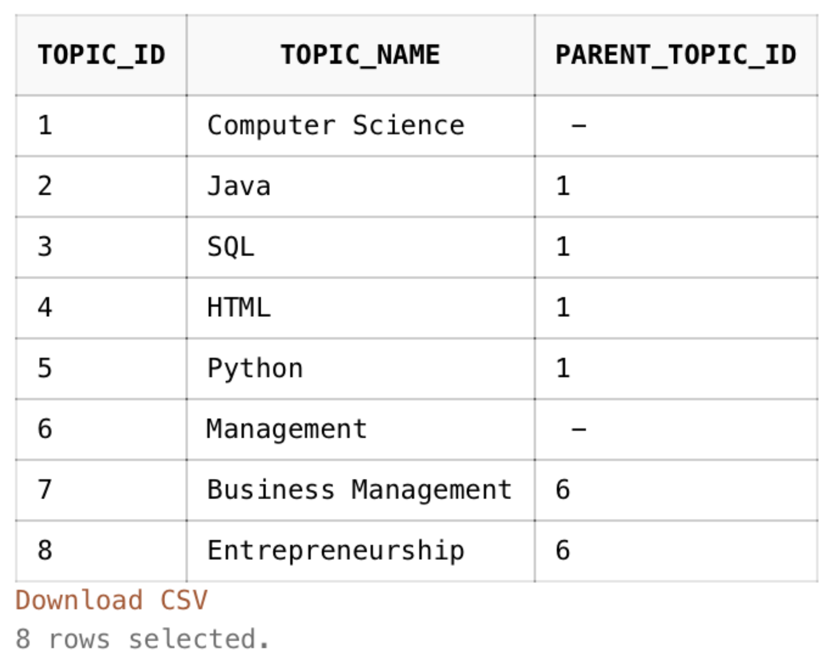
```
Statement processed.

Statement processed.
```

QUE_ID	QUE_TEXT	TOTAL_SCORE
2	What is a garbage collection in Java, How does it work?	20

Download CSV

Populated Tables

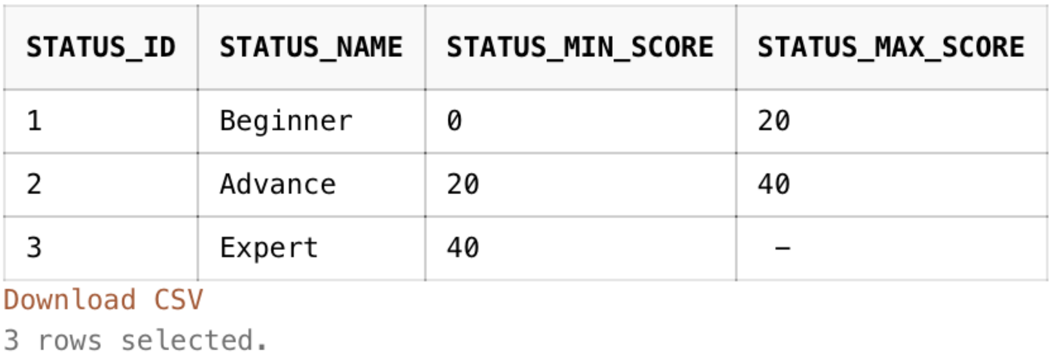
Topic -



TOPIC_ID	TOPIC_NAME	PARENT_TOPIC_ID
1	Computer Science	-
2	Java	1
3	SQL	1
4	HTML	1
5	Python	1
6	Management	-
7	Business Management	6
8	Entrepreneurship	6

Download CSV
8 rows selected.

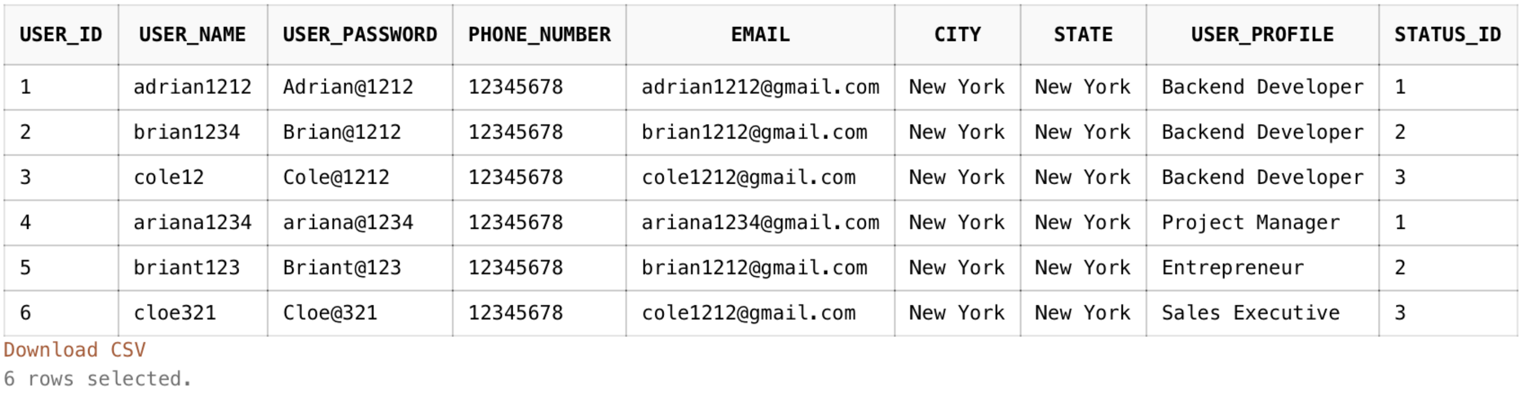
Status -



STATUS_ID	STATUS_NAME	STATUS_MIN_SCORE	STATUS_MAX_SCORE
1	Beginner	0	20
2	Advance	20	40
3	Expert	40	-

Download CSV
3 rows selected.

Users -

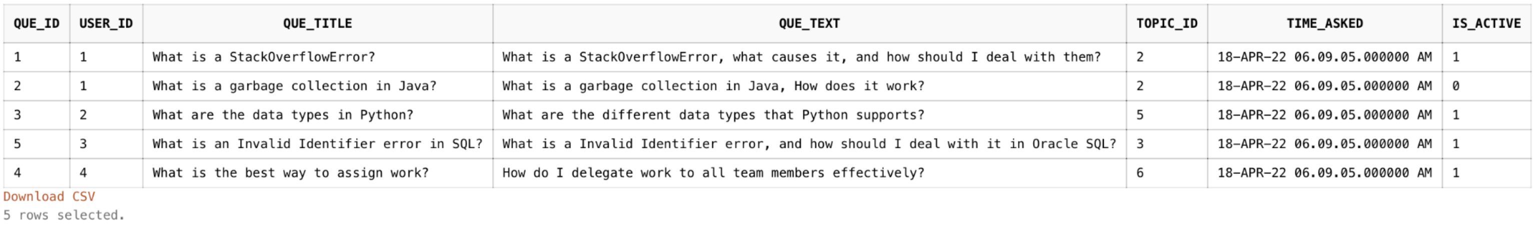


USER_ID	USER_NAME	USER_PASSWORD	PHONE_NUMBER	EMAIL	CITY	STATE	USER_PROFILE	STATUS_ID
1	adrian1212	Adrian@1212	12345678	adrian1212@gmail.com	New York	New York	Backend Developer	1
2	brian1234	Brian@1212	12345678	brian1212@gmail.com	New York	New York	Backend Developer	2
3	cole12	Cole@1212	12345678	cole1212@gmail.com	New York	New York	Backend Developer	3
4	ariana1234	ariana@1234	12345678	ariana1234@gmail.com	New York	New York	Project Manager	1
5	briant123	Briant@123	12345678	brian1212@gmail.com	New York	New York	Entrepreneur	2
6	cloe321	Cloe@321	12345678	cole1212@gmail.com	New York	New York	Sales Executive	3

Download CSV

6 rows selected.

Question -

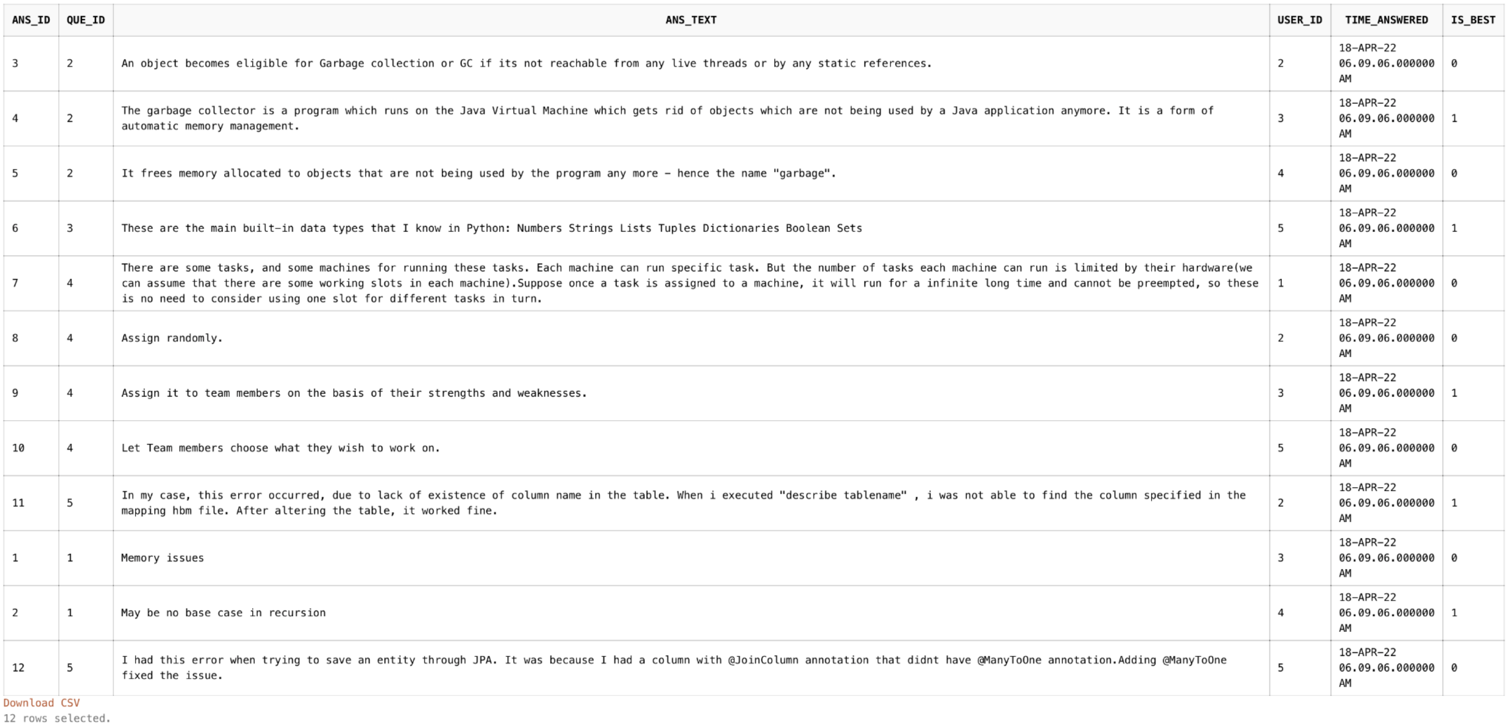


QUE_ID	USER_ID	QUE_TITLE	QUE_TEXT	TOPIC_ID	TIME_ASKED	IS_ACTIVE
1	1	What is a StackOverflowError?	What is a StackOverflowError, what causes it, and how should I deal with them?	2	18-APR-22 06.09.05.000000 AM	1
2	1	What is a garbage collection in Java?	What is a garbage collection in Java, How does it work?	2	18-APR-22 06.09.05.000000 AM	0
3	2	What are the data types in Python?	What are the different data types that Python supports?	5	18-APR-22 06.09.05.000000 AM	1
5	3	What is an Invalid Identifier error in SQL?	What is a Invalid Identifier error, and how should I deal with it in Oracle SQL?	3	18-APR-22 06.09.05.000000 AM	1
4	4	What is the best way to assign work?	How do I delegate work to all team members effectively?	6	18-APR-22 06.09.05.000000 AM	1

Download CSV

5 rows selected.

Answer -

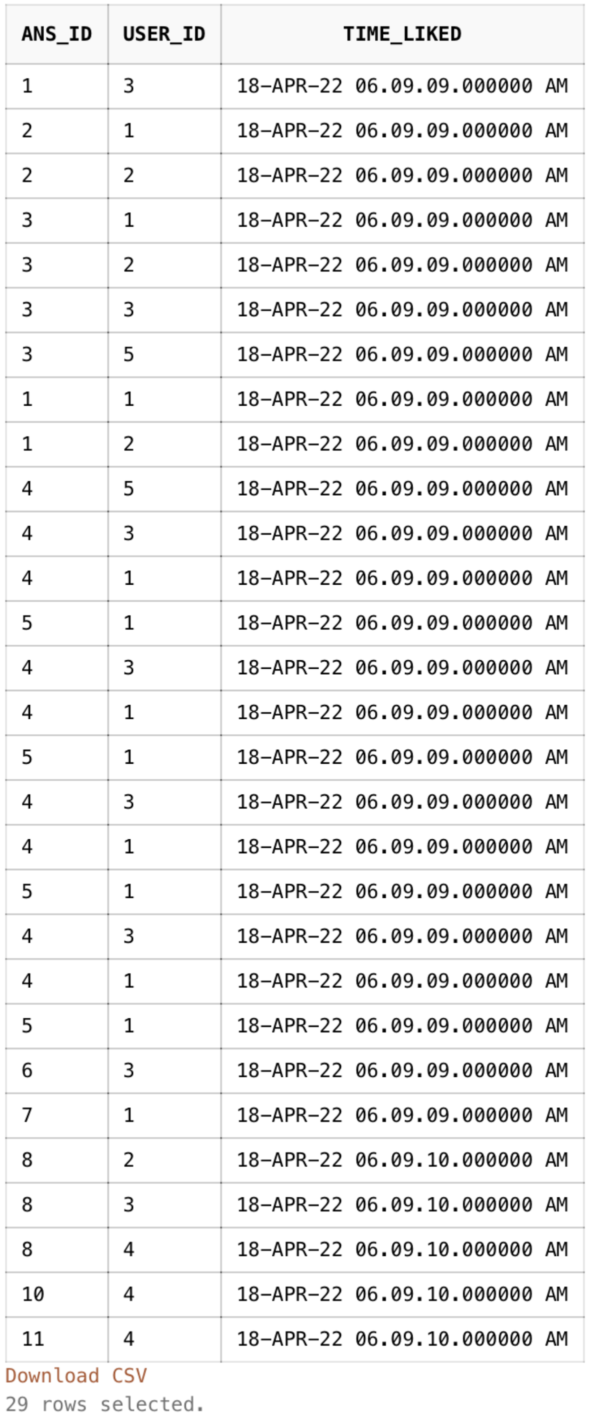


ANS_ID	QUE_ID	ANS_TEXT	USER_ID	TIME_ANSWERED	IS_BEST
3	2	An object becomes eligible for Garbage collection or GC if its not reachable from any live threads or by any static references.	2	18-APR-22 06.09.06.000000 AM	0
4	2	The garbage collector is a program which runs on the Java Virtual Machine which gets rid of objects which are not being used by a Java application anymore. It is a form of automatic memory management.	3	18-APR-22 06.09.06.000000 AM	1
5	2	It frees memory allocated to objects that are not being used by the program any more - hence the name "garbage".	4	18-APR-22 06.09.06.000000 AM	0
6	3	These are the main built-in data types that I know in Python: Numbers Strings Lists Tuples Dictionaries Boolean Sets	5	18-APR-22 06.09.06.000000 AM	1
7	4	There are some tasks, and some machines for running these tasks. Each machine can run specific task. But the number of tasks each machine can run is limited by their hardware(we can assume that there are some working slots in each machine).Suppose once a task is assigned to a machine, it will run for a infinite long time and cannot be preempted, so these is no need to consider using one slot for different tasks in turn.	1	18-APR-22 06.09.06.000000 AM	0
8	4	Assign randomly.	2	18-APR-22 06.09.06.000000 AM	0
9	4	Assign it to team members on the basis of their strengths and weaknesses.	3	18-APR-22 06.09.06.000000 AM	1
10	4	Let Team members choose what they wish to work on.	5	18-APR-22 06.09.06.000000 AM	0
11	5	In my case, this error occurred, due to lack of existence of column name in the table. When i executed "describe tablename" , i was not able to find the column specified in the mapping hbm file. After altering the table, it worked fine.	2	18-APR-22 06.09.06.000000 AM	1
1	1	Memory issues	3	18-APR-22 06.09.06.000000 AM	0
2	1	May be no base case in recursion	4	18-APR-22 06.09.06.000000 AM	1
12	5	I had this error when trying to save an entity through JPA. It was because I had a column with @JoinColumn annotation that didnt have @ManyToOne annotation.Adding @ManyToOne fixed the issue.	5	18-APR-22 06.09.06.000000 AM	0

Download CSV

12 rows selected.

Thumbs Up -



ANS_ID	USER_ID	TIME_LIKED
1	3	18-APR-22 06.09.09.000000 AM
2	1	18-APR-22 06.09.09.000000 AM
2	2	18-APR-22 06.09.09.000000 AM
3	1	18-APR-22 06.09.09.000000 AM
3	2	18-APR-22 06.09.09.000000 AM
3	3	18-APR-22 06.09.09.000000 AM
3	5	18-APR-22 06.09.09.000000 AM
1	1	18-APR-22 06.09.09.000000 AM
1	2	18-APR-22 06.09.09.000000 AM
4	5	18-APR-22 06.09.09.000000 AM
4	3	18-APR-22 06.09.09.000000 AM
4	1	18-APR-22 06.09.09.000000 AM
5	1	18-APR-22 06.09.09.000000 AM
4	3	18-APR-22 06.09.09.000000 AM
4	1	18-APR-22 06.09.09.000000 AM
5	1	18-APR-22 06.09.09.000000 AM
4	3	18-APR-22 06.09.09.000000 AM
4	1	18-APR-22 06.09.09.000000 AM
5	1	18-APR-22 06.09.09.000000 AM
4	3	18-APR-22 06.09.09.000000 AM
4	1	18-APR-22 06.09.09.000000 AM
5	1	18-APR-22 06.09.09.000000 AM
6	3	18-APR-22 06.09.09.000000 AM
7	1	18-APR-22 06.09.09.000000 AM
8	2	18-APR-22 06.09.10.000000 AM
8	3	18-APR-22 06.09.10.000000 AM
8	4	18-APR-22 06.09.10.000000 AM
10	4	18-APR-22 06.09.10.000000 AM
11	4	18-APR-22 06.09.10.000000 AM

Download CSV

29 rows selected.

Site Features

Registration :

A new user account is created using the registration page.
All the registration fields are mandatory

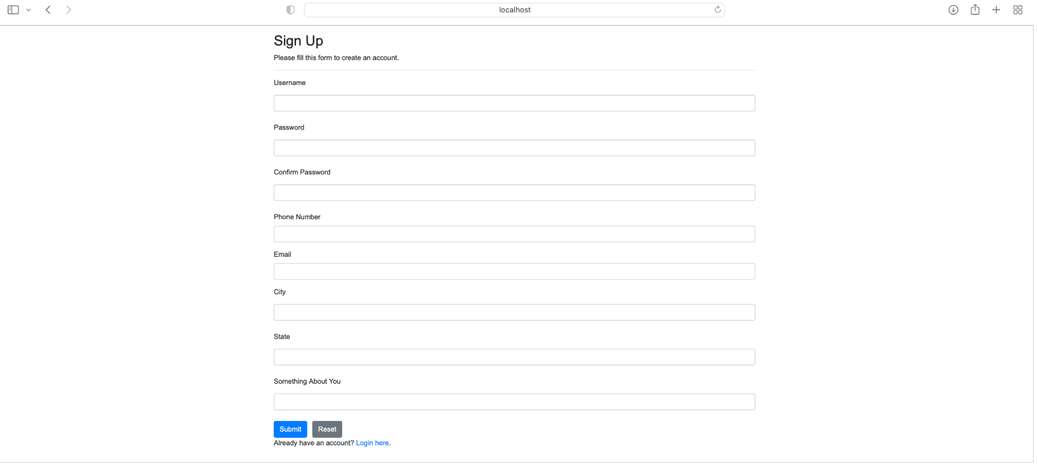
- Username
- Password
- Phone Number
- Email
- City
- State
- Profile

Account creation validation and checks:

- Username has to be Unique
- Both password and re-enter password should match
- Phone number has to be a 10-digit number
- Email has to be valid

On successful registration we are able to redirect to Login otherwise the Respective validation error corresponding to the form field is shown on the same page.

A Reset button is present to clear the form if there are pre-filled values or multiple changes to be made in the form.



Sign Up

Please fill this form to create an account.

Username

Password

Confirm Password

Phone Number

Email

City

State

Something About You

Already have an account? Login here.

Login :

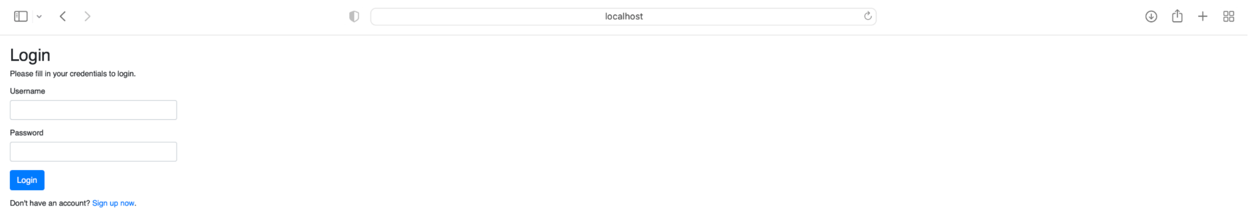
After a user account is created. Users can login using the login page passing username and password.

Checks and Validation:

- If the given username is present or not.
- If the password matches corresponding to the username

After a successful login a session is created for the user.

In case a user doesn't have an account a signup link is provided at the bottom.



Login

Please fill in your credentials to login.

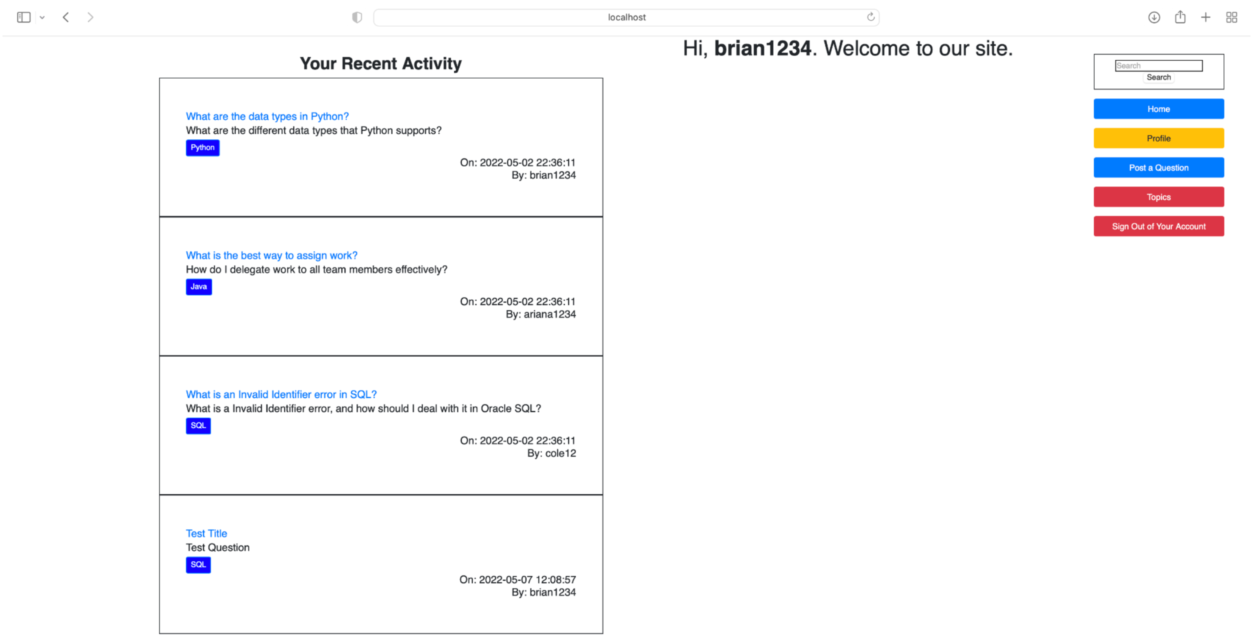
Don't have an account? Sign up now.

Main Page:

All the User activity is presented on this page in reverse chronological order.

User Activities such as
- Questions posted by user
- Answers provided by user
- Any new comments on the answer provided by user

There is also a menu bar at the right side.
- Search
- Topics
- User Profile
- Post a Question
- Signout



Hi, **brian1234**. Welcome to our site.

Your Recent Activity

What are the data types in Python?
What are the different data types that Python supports?
Python
On: 2022-05-02 22:36:11
By: brian1234

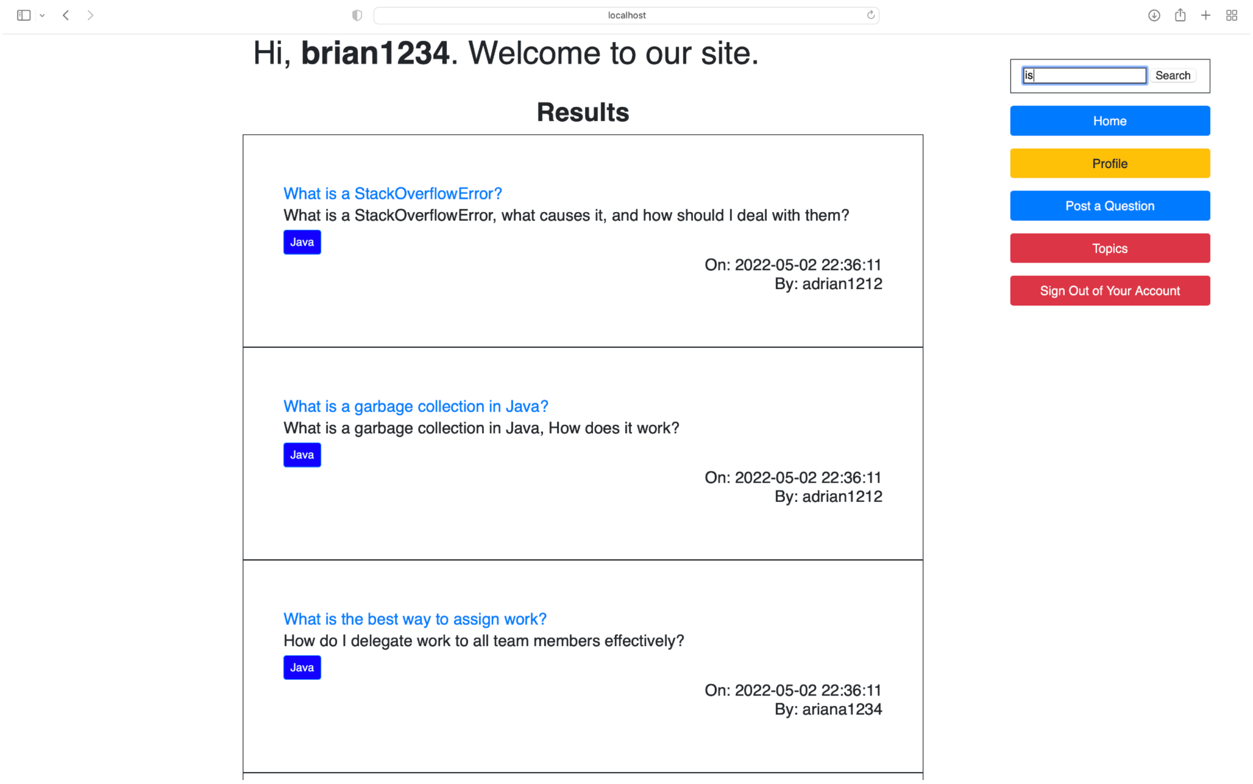
What is the best way to assign work?
How do I delegate work to all team members effectively?
Java
On: 2022-05-02 22:36:11
By: ariana1234

What is an Invalid Identifier error in SQL?
What is a Invalid Identifier error, and how should I deal with it in Oracle SQL?
SQL
On: 2022-05-02 22:36:11
By: cole12

Test Title
Test Question
SQL
On: 2022-05-07 12:08:57
By: brian1234

Search :

Filter and result the questions which have the search keyword either in Question Title or Question Text.



Hi, **brian1234**. Welcome to our site.

Results

What is a StackOverflowError?
What is a StackOverflowError, what causes it, and how should I deal with them?
Java
On: 2022-05-02 22:36:11
By: adrian1212

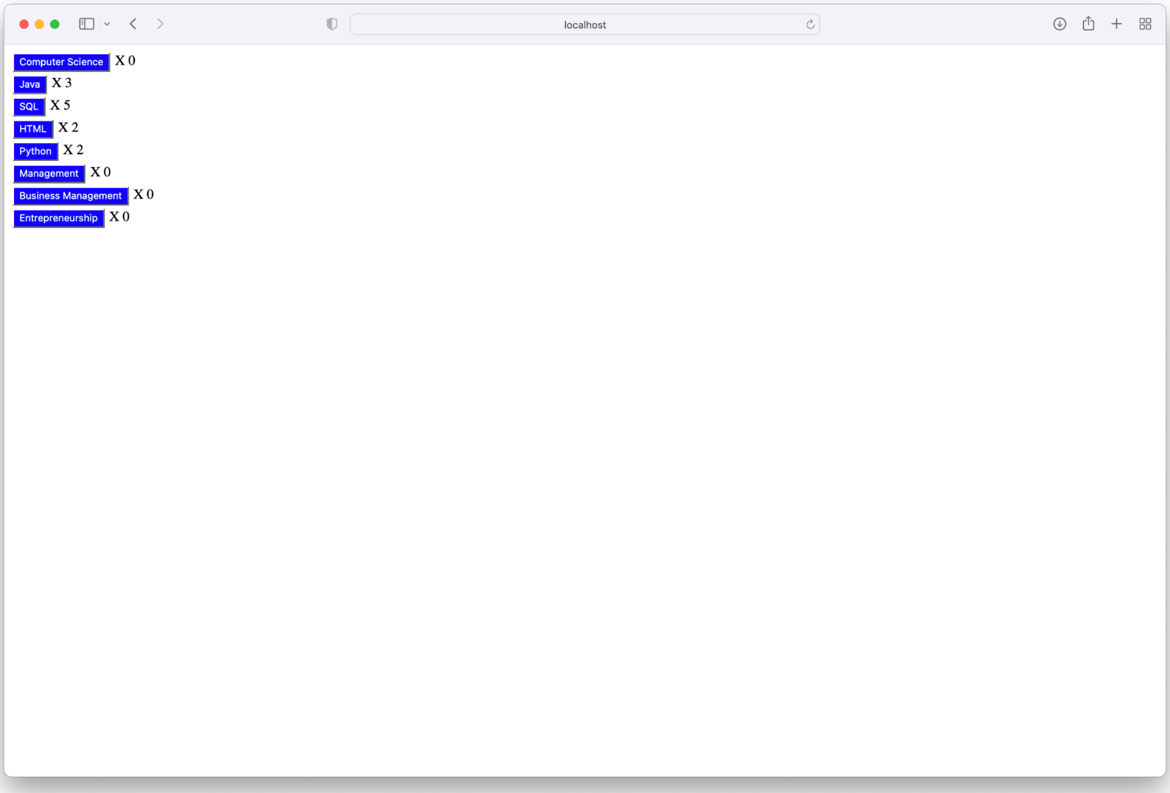
What is a garbage collection in Java?
What is a garbage collection in Java, How does it work?
Java
On: 2022-05-02 22:36:11
By: adrian1212

What is the best way to assign work?
How do I delegate work to all team members effectively?
Java
On: 2022-05-02 22:36:11
By: ariana1234

Topics :

Redirects to a new page where there is a List of all the topics and number of questions related to each topic that have been asked so far.

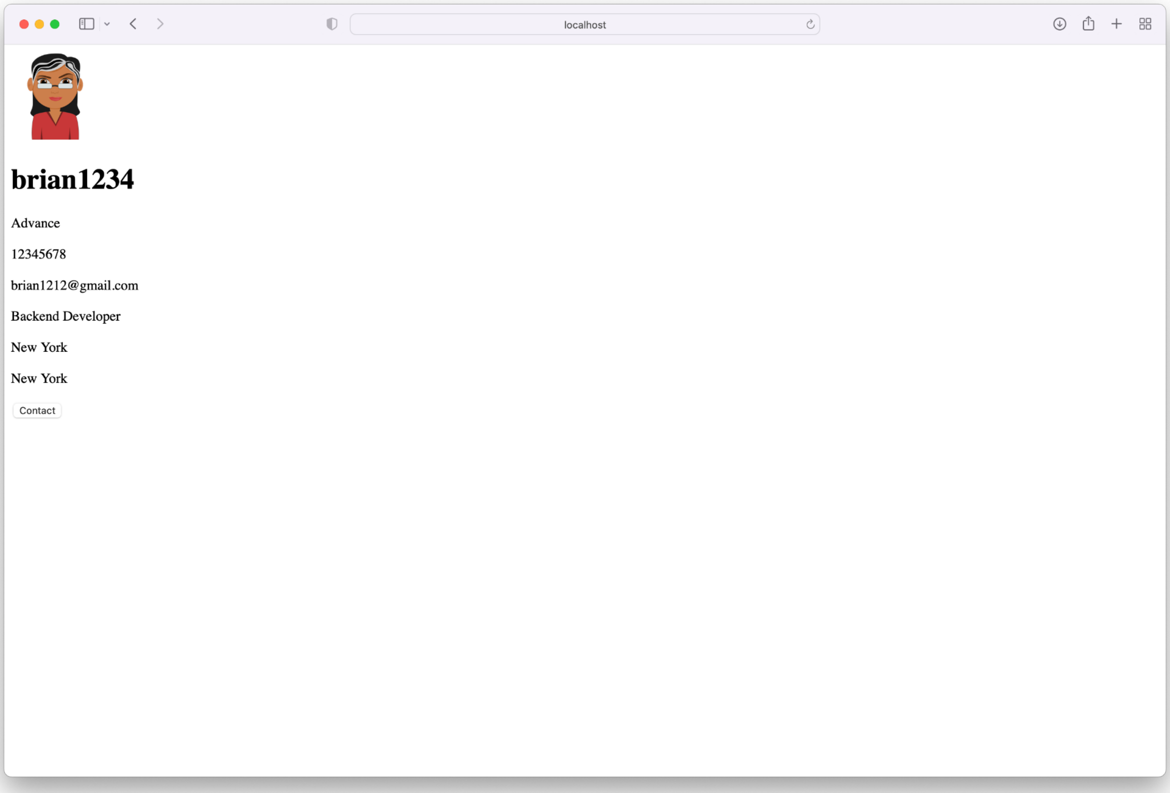
Clicking on each topic redirects to all the questions asked related to that topic.



Computer Science X 0
Java X 3
SQL X 5
HTML X 2
Python X 2
Management X 0
Business Management X 0
Entrepreneurship X 0

User Profile:

All the information regarding the user is shown on this page. Including the Status of the user i.e, beginner, Expert or Advance.



brian1234

Advance

12345678

brian1212@gmail.com

Backend Developer

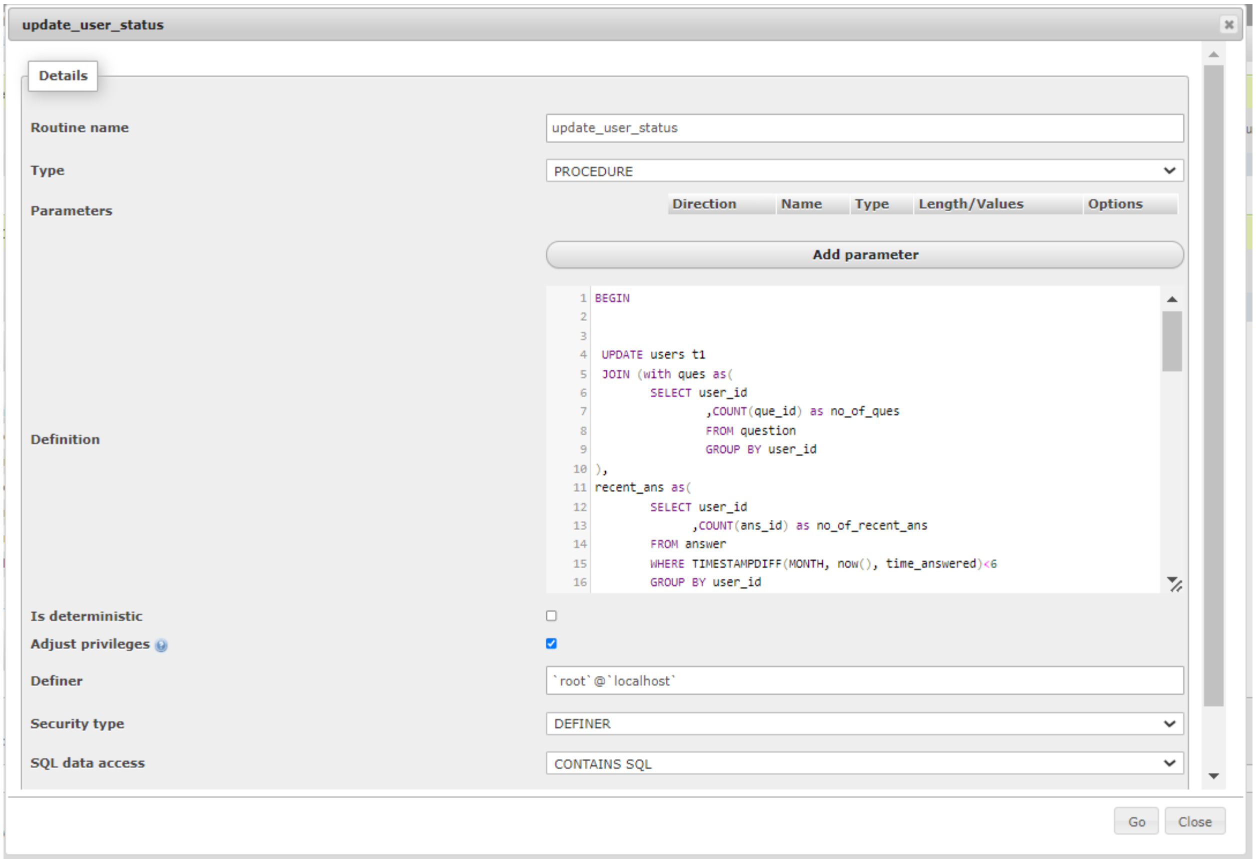
New York

New York

Updating User Status :

- We have created a Stored procedure which is called every 15 seconds to update the user status based on the criteria discussed above.

  Procedure :



Routine name: update_user_status

Type: PROCEDURE

Parameters:

Direction	Name	Type	Length/Values	Options

Definition:

```
BEGIN


 UPDATE users t1
 JOIN (with ques as(
        SELECT user_id
                ,COUNT(que_id) as no_of_ques
                FROM question
                GROUP BY user_id
),
recent_ans as(
        SELECT user_id
              ,COUNT(ans_id) as no_of_recent_ans
        FROM answer
        WHERE TIMESTAMPDIFF(MONTH, now(), time_answered)<6
        GROUP BY user_id
```

Is deterministic ☐

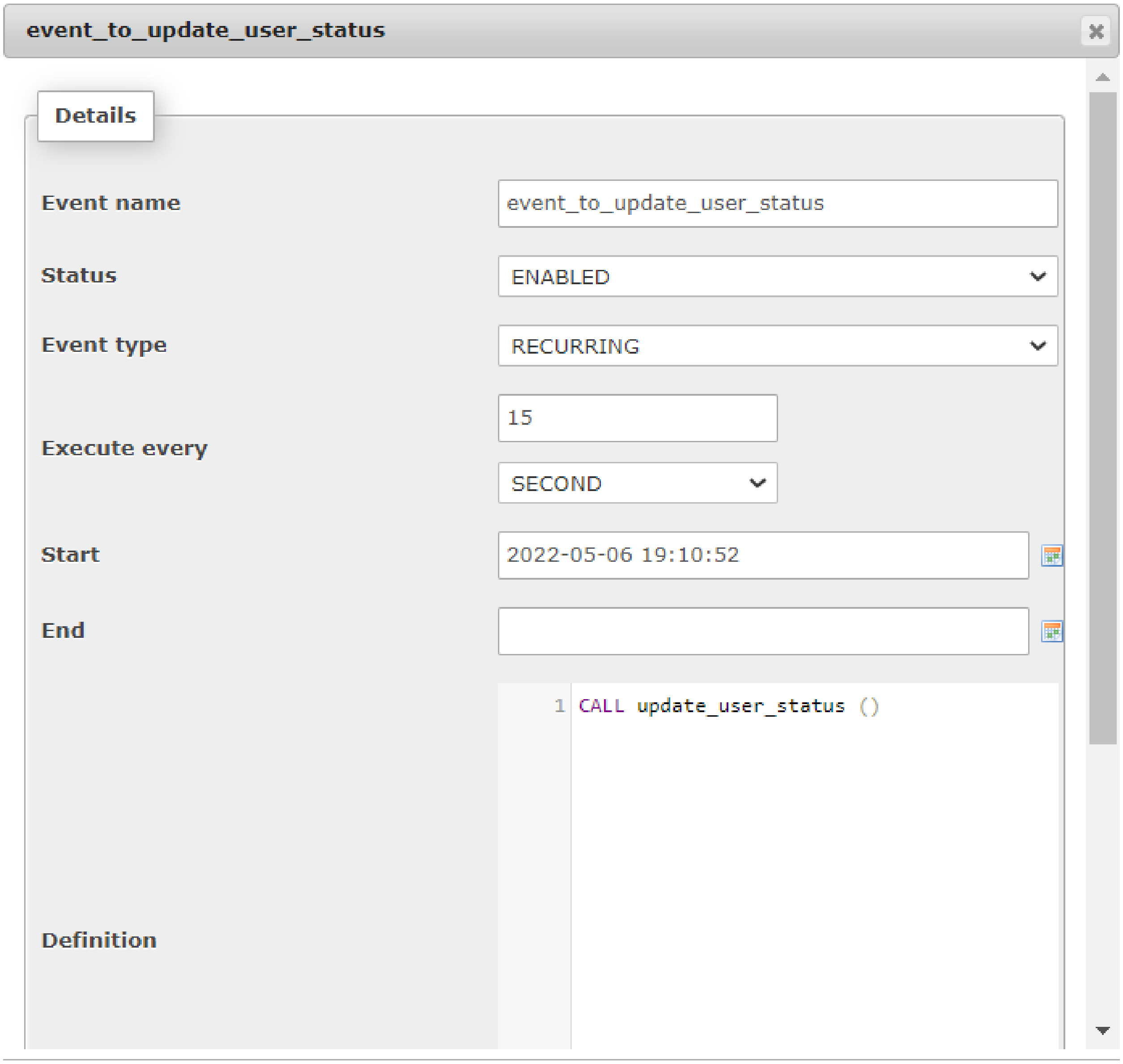
Adjust privileges ☑

Definer: `root`@`localhost`

Security type: DEFINER

SQL data access: CONTAINS SQL

Event Scheduler :



event_to_update_user_status

Details

Event name	event_to_update_user_status
Status	ENABLED
Event type	RECURRING
Execute every	15 SECOND
Start	2022-05-06 19:10:52
End	
Definition	CALL update_user_status ()

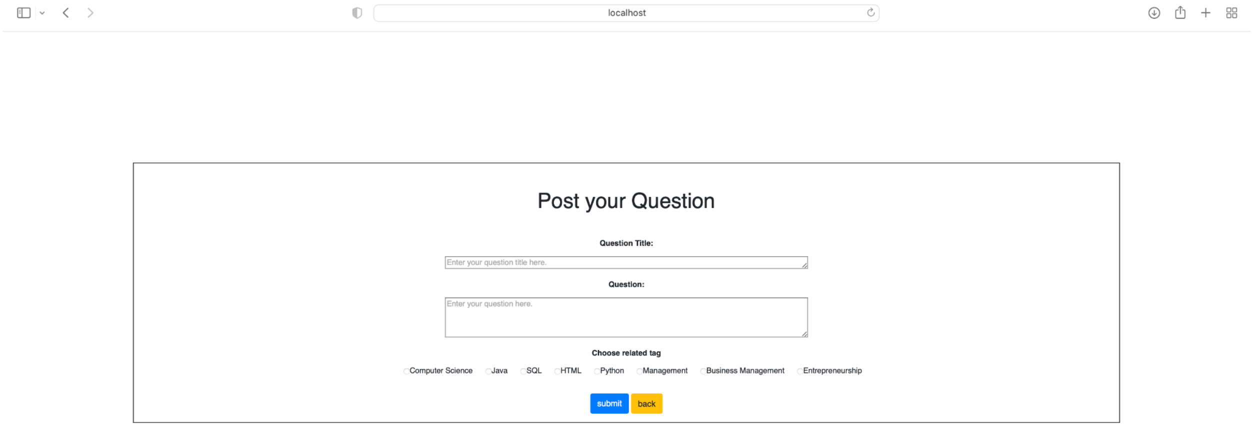
Signout :

- Logout the user and Ends the session and session data of the user logged in.

Post a Question:

A user will be able to post questions, They will be able to add a question title, question text (Information for users to be able to answer the question) and choose a tag related to the question.

All three fields are mandatory to post a question, once the user clicks the submit button, the fields are validated to make sure they have been filled and then the question gets added to the database.



Post your Question

Question Title:

Enter your question title here.

Question:

Enter your question here.

Choose related tag

○ Computer Science ○ Java ○ SQL ○ HTML ○ Python ○ Management ○ Business Management ○ Entrepreneurship

View Selected Question:

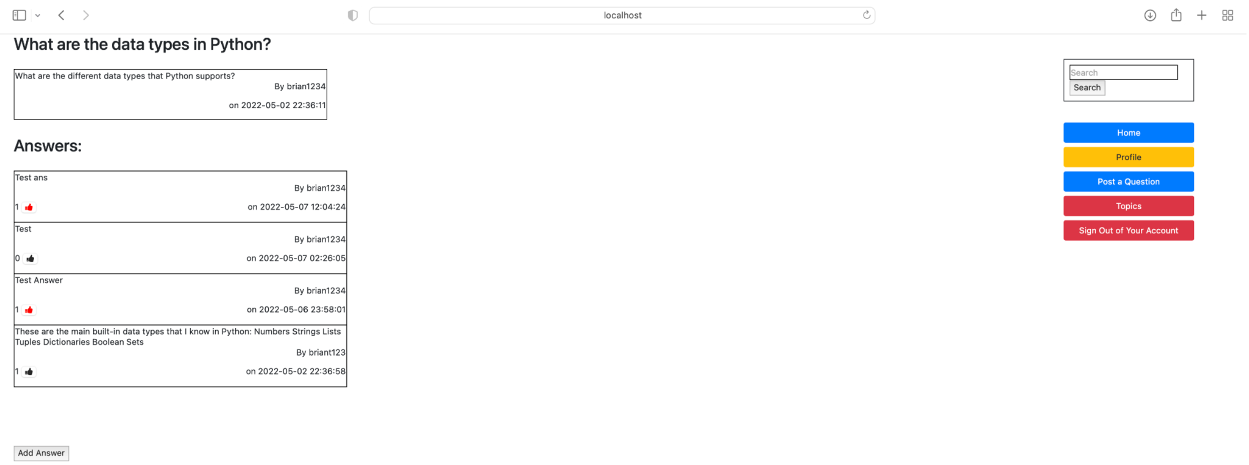
The user will be able to select a question to view its answers. The user will also be able to post answers to the question and like/unlike answers.

Once the users likes an answer the color of the like button will change to red and when they click on the button again, they will unlike the answer and color will change to gray.

All the answers that the user has already liked will have the red like button.

The user can select the add answer button to post an answer to the question.

All answers to the question will appear in reverse chronological order.



What are the data types in Python?

What are the different data types that Python supports?
By brian1234
on 2022-05-02 22:36:11

Answers:

Test ans
By brian1234
1 👍 on 2022-05-07 12:04:24

Test
By brian1234
0 👍 on 2022-05-07 02:26:05

Test Answer
By brian1234
1 👍 on 2022-05-06 23:58:01

These are the main built-in data types that I know in Python: Numbers Strings Lists Tuples Dictionaries Boolean Sets
By briant123
1 👍 on 2022-05-02 22:36:58