Roll No. _____ **Name** _____ **Section** ___

National University of Computer and Emerging Sciences, Lahore Campus

Course:	**Database Systems**	**Course Code:**	**CS2005**
Program:	**BS(CS, DS, SE)**	**Semester:**	**Spring 2024**
Duration:	**60 Minutes**	**Total Marks:**	**35**
Paper Date:	**28-Feb-2024**	**Weight**	**15%**
Section:	**ALL**	**Page(s):**	**5**
Exam:	**Midterm-I - SOLUTION**		

Instruction/Notes: All the questions should be solved on the question paper. You will not get any credit if you do not show proper working, reasoning, and steps as asked in the question statements.

Consider the following simplified database schema for a forum post system like *Stack Overflow*. A forum post system is an online platform where users can discuss by posting messages. Users can create new posts or reply to existing posts.

In the DB schema given below:

- The Post table stores information about forum posts. The AuthorID is the ID of the User who created the post.
- The User table stores information about users.
- The Reply table stores replies to forum posts. A user can write a reply to an existing reply, meaning that replies are hierarchical and can be organized into parent-child relationships. Each reply can have zero or more replies, and each reply is associated with a parent reply. The column ParentReplyID indicates the parent reply to which the current reply is a response. If a reply is a direct response to the main post, then the "ParentReplyID" is NULL.

```
CREATE TABLE Post (
    PostID INT PRIMARY KEY,
    Title VARCHAR(255),
    Content TEXT,
    AuthorID INT,
    CreatedAt DATETIME
);
```

```
CREATE TABLE Reply (
    ReplyID INT PRIMARY KEY,
    PostID INT,
    ReplyText TEXT,
    AuthorID INT,
    ParentReplyID INT,
    CreatedAt DATETIME
);
```

```
CREATE TABLE User (
    UserID INT PRIMARY KEY,
    Username VARCHAR(50),
    Email VARCHAR(100),
    Gender CHAR(1)
);
```

Q.1(4 points) Add **referential integrity constraints:** "On Delete cascade" and "On Update Cascade" to the foreign keys in the POST, REPLY, and USER tables.

ALTER TABLE post ADD CONSTRAINT fk_authorP_id FOREIGN KEY (authorId) REFERENCES user (userid) ON DELETE CASCADE on UPDATE CASCADE;

The column 'UserId' in user table is a foreign key in Reply table with name author Id and referential integrity constraint is on Delete cascade.

ALTER TABLE reply ADD CONSTRAINT fk_authorP_id FOREIGN KEY (authorId) REFERENCES user (userid) ON DELETE CASCADE on UPDATE CASCADE;

Roll No. _____ Name _____ Section ___

The column 'PostId' in post table is a foreign key in Reply table with name postId and referential integrity constraint is on Delete cascade.

ALTER TABLE reply ADD CONSTRAINT fk_authorP_id FOREIGN KEY (postId) REFERENCES post (postId) ON DELETE CASCADE on UPDATE CASCADE;

`ParentReplyID is a foreign key (that references replyID from same table. SELF reference….`

ALTER TABLE reply ADD CONSTRAINT fk_parentr_id FOREIGN KEY (`ParentReplyID`) REFERENCES Reply (ReplyId) ON DELETE CASCADE on UPDATE CASCADE;

Roll No. _____ Name _____ Section ___

Q.2 *(15 points)* Specify the following queries in ***SQL***

a. List the IDs and Names of the Female **Users who have not created any posts**.

a) SELECT u.UserID, u.Username
FROM User u LEFT JOIN Post p ON u.UserID = p.AuthorID
WHERE p.PostID IS NULL AND u.Gender = 'F';

Alternate Method:

```
SELECT UserID, Username
FROM User
WHERE Gender = 'F' AND userID NOT IN (SELECT authorID FROM Post);
```

b. Print IDs of the **Replies** that have received two or more replies.

```
SELECT ParentReplyID
FROM Reply
GROUP BY ParentReplyID
HAVING COUNT(ReplyID) >= 2;
```

c. List the usernames of users who have replied to their posts.

```
SELECT u.Username
FROM User u JOIN Reply r ON u.UserID = r.AuthorID JOIN Post p ON r.PostID = p.PostID
WHERE u.UserID = p.AuthorID;
```

Alternate Method:

```
SELECT Username
FROM User
WHERE UserID IN (SELECT P.authorID FROM Reply R JOIN Post P ON R.postID=P.postID AND
R.authodID=P.authorID);
```

Roll No. _____ Name _____ Section ___

Consider the following database state.

User table:

UserID	UserName	Gender	Email
1	Alice	Female	alice@example.com
2	Bob	Male	bob@example.com
3	Charlie	Male	charlie@example.com

Post table:

PostID	Title	CreatedAt	AuthorID	Content
1	Introduction	2024-02-20	1	Welcome to our platform!
2	Tips and Tricks	2024-02-21	2	Here are some tips for you.
3	Question about AI	2024-02-22	3	I have a question about AI.
4	Programming Question	2024-02-23	1	I need help with programming.
5	Data Science	2024-02-24	2	Let's discuss data science.

Reply table:

ReplyID	PostID	AuthorID	ParentReplyID	ReplyText	CreatedAt
1	1	2	NULL	Welcome, Alice!	2024-02-20
2	1	1	NULL	Thanks, Bob!	2024-02-21
3	1	3	1	Hello, everyone!	2024-02-22
4	1	2	2	Hi, Alice!	2024-02-22
5	2	1	NULL	Great tips, Bob!	2024-02-21
6	2	3	5	I agree!	2024-02-22
7	3	2	NULL	Can someone help me?	2024-02-22
8	3	1	NULL	Sure, what's up?	2024-02-22
9	3	3	7	What do you need help with?	2024-02-22

Q3. (9 points) Consider the above database state and give the output tuples generated after running the following queries. Also, explain in ONE sentence what these queries do

a) SELECT u.UserName AS name, p.Title AS title, r.ReplyText AS text
FROM (User u JOIN Post p ON u.UserID = p.AuthorID) JOIN Reply r ON p.PostID = r.PostID WHERE p.Title <> 'Introduction' AND r.ParentReplyID IS NULL;

Answer: This query will return the users who have posted more than once, sorted by the number of posts they have made.

Name	Title	Text
Bob	Tips and Tricks	Great tips, Bob!
Charlie	Question about AI	Can someone help me?
Charlie	Question about AI	Sure, what's up?

b)
```
SELECT p.PostID AS ID,  p.Title AS title, u.UserName AS name, COUNT(r.ReplyID) as CountR
FROM  (Post p LEFT JOIN  User u ON p.AuthorID = u.UserID ) LEFT JOIN  Reply r ON
p.PostID = r.PostID GROUP BY  p.PostID, p.Title, u.UserName
ORDER BY CountR DESC;
```

Answer: showing each post along with its author's name and the number of replies it has received.

ID	title	name	CountR
1	Introduction	Alice	4
3	Question about AI	Charlie	3
2	Tips and Tricks	Bob	2
4	Programming Question	Alice	1
5	Data Science	Bob	1

c)
```
SELECT UserID, UserName, COUNT(*) as
Countp FROM User u JOIN  Post p  ON u.user_id
= p.author_id GROUP BY UserID, UserName
HAVING COUNT(*) > 1
ORDER BY UserID desc, UserName desc, Countp DESC;
```

Answer:This query retrieves the usernames, post titles, and top-level reply texts for posts that are not titled 'Introduction'.

UserID	UserName	CountP
2	Bob	2
1	Alice	2

Roll No. _____ Name _____ Section ___

Q.4 (4 points) Considering the constraints applied on the schema in (Q1) and **database state** given in (Q3).

Apply the following operations on the above database. State if the operation would be carried out successfully or not. In case of successful operation, indicate the changes that will be made to the above database, clearly specify the name of the table and change. Also, state all the integrity constraints violated by each operation, if any. Please note that **all operations are independent**.

1. DELETE FROM User WHERE username='Bob';

Successful, ten rows deleted (one from user, 2 from post and 7 from reply table).

2. INSERT INTO reply VALUES (10,6,' hello',3,NULL,' 2024-02-13');

Failed, reference integrity issue, parent value 6 does not exist in post table.

3. UPDATE post SET postId=7 WHERE title='introduction';

Successful, five rows updated (one from post table and four from reply table).

4. DELETE FROM post WHERE postId=4;

Successful, one row deleted from post table.