**TOPICS: Allowed Time:** 90 Minutes

**Instructions: Total Marks:** 35

1. Gossips are not allowed.
2. Teacher assistants are for your help, so be nice with them. Respect them as they are teaching you. Raise your hands if you have some problem and need help from TA. Avoid calling them by raising your voice and disturbing the environment of Lab.
3. TA may deduct your marks for any kind of ill-discipline or misconduct from your side.
4. Evaluation will be considered final and you cannot debate for the marks. So, focus on performing the tasks when the time is given to you.
5. Paste the query as well as result table screenshot as a result of each task

**Task 01: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ (10 Marks)**

| **PlaybackID** | **UserID** | **UserName** | **UserEmail** | **PlaylistID** | **PlaylistName** | **SongID** | **SongTitle** | **Artist** | **Album** | **PlayDateTime** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | U001 | John Doe | john.doe@mail.com | PL001 | Chill Vibes | S001 | Ocean Eyes | Billie Eilish | Don't Smile At Me | 2024-12-01 10:30:00 |
| 2 | U001 | John Doe | john.doe@mail.com | PL001 | Chill Vibes | S002 | Blinding Lights | The Weeknd | After Hours | 2024-12-01 11:00:00 |

A music streaming platform tracks its users, playlists, songs, and playback history in a single table. Each time a user listens to a song, a record is created in the system. Over time, this table has grown large, and the platform management wants to organize the data to reduce redundancy, improve data integrity, and scale effectively for millions of users and songs.

Normalize the given table up to **3NF** while fulfilling the following requirements

Determine the relationships between the table columns, identifying the primary key and alternative keys (if any).

1. **Eliminate Redundancy**:
   * Avoid repeating user, playlist, and song details across multiple records.
2. **Preserve Relationships**:
   * Maintain relationships between users, playlists, songs, and playback history.
3. **Ensure Data Integrity**:
   * Ensure updates to any information (e.g., user email, playlist name, song title) do not lead to inconsistencies.

Normalize this relation upto 3NF

Playback Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PBID | UID | PLID | SID | PDTime |
| 1 | U001 | PL001 | S001 | 2024-12-01 10:30:00 |
| 2 | U001 | PL001 | S002 | 2024-12-01 11:00:00 |

User Table

|  |  |  |
| --- | --- | --- |
| UID | UName | UEmail |
| U001 | John Doe | john.doe@mail.com |

Playlist Table

|  |  |
| --- | --- |
| PLID | PLName |
| PL001 | Chill Vibes |

Song Table

|  |  |  |  |
| --- | --- | --- | --- |
| SID | STitle | Artist | Album |
| S001 | Ocean Eyes | Billie | Don’t smile at me |
| S002 | Blinding Lights | Weeknd | After Hours |

**Task 02: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ (10 Marks)**

An online movie rental platform keeps track of movie rentals using a single table. The platform provides details about movies, customers, and rental transactions. However, the current table is poorly designed, with multiple values in some columns, making it challenging to maintain data integrity.

**Scenario Context**

1. Each movie has a unique MovieID, title, and genre.
2. Customers are identified by a unique CustomerID and have details like name and email.
3. A customer can rent multiple movies in a single transaction, leading to multiple MovieIDs being recorded in the same column as a comma-separated list.
4. The rental record includes the rental date and the due date for returning the movies.

The management wants to:

* Organize the database to avoid confusion caused by multiple values in columns.
* Ensure scalability for millions of customers and movies.
* Maintain accurate relationships between customers, movies, and rental transactions.

 **Identify Issues**:

* Identify all problems caused by the current table structure, including hints of repeating groups and dependencies.

 **Normalize the Table**:

* Break down the table into separate tables and normalize up to **3NF** to eliminate redundancy, ensure data integrity, and improve scalability.

 **Define Relationships**:

* Create appropriate relationships between customers, movies, and rentals while ensuring no data is lost.

Normalize this relation upto 3NF

| **RentalID** | **CustomerID** | **CustomerName** | **CustomerEmail** | **MovieID** | **MovieTitle** | **Genre** | **RentalDate** | **DueDate** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | C001 | John Doe | john.doe@mail.com | M001 | Inception | Sci-Fi | 2024-12-01 | 2024-12-05 |
| 1 | C001 | John Doe | john.doe@mail.com | M002 | Interstellar | Sci-Fi | 2024-12-01 | 2024-12-05 |
| 2 | C002 | Jane Smith | jane.smith@mail.com | M003 | The Godfather | Crime | 2024-12-02 | 2024-12-06 |
| 3 | C003 | Sarah Johnson | sarah.johnson@mail.com | M001 | Inception | Sci-Fi | 2024-12-03 | 2024-12-07 |
| 3 | C003 | Sarah Johnson | sarah.johnson@mail.com | M004 | Titanic | Romance | 2024-12-03 | 2024-12-07 |
| 3 | C003 | Sarah Johnson | sarah.johnson@mail.com | M005 | Avatar | Sci-Fi | 2024-12-03 | 2024-12-07 |

Rental Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RENTALID | CID | MID | RDate | DDate |
| 1 | C001 | M001 | 2024-12-01 | 2024-12-05 |
| 1 | C001 | M002 | 2024-12-01 | 2024-12-05 |
| 2 | C002 | M003 | 2024-12-02 | 2024-12-06 |
| 3 | C003 | M001 | 2024-12-03 | 2024-12-07 |
| 3 | C003 | M004 | 2024-12-03 | 2024-12-07 |
| 3 | C003 | M005 | 2024-12-03 | 2024-12-07 |

Customer Table

|  |  |  |
| --- | --- | --- |
| CID | CName | CEmail |
| C001 | John Doe | John.doe@mail.com |
| C002 | Jane Smith | Jane.smith@mail.com |
| C003 | Sarah Johnson | Sarah.johnson@mail.com |

Movie Table

|  |  |  |
| --- | --- | --- |
| MID | Title | Genre |
| M001 | Inception | Sci-Fi |
| M002 | Interstellar | Sci-Fi |
| M003 | Godfather | Crime |
| M004 | Titanic | Romance |
| M005 | Avatar | Sci-Fi |