**Homework#1**

**ACS575 Database Systems, Spring 2024**

**Q6. Translate the (E)ER-diagram from Q5 (1) into a set of relations suitable for the database of “Musicmatic”.**

**Ans :**

**User Relation:**

**Represented as 𝑅(UserID)**

**Primary Key: UserID**

**BusinessUser Relation:**

**Represented as 𝑅(UserID, VATNumber, upload, *BusinessUserID*)**

**RegularUser Relation:**

**Represented as 𝑅(UserID, createSuggestion, purchase, *RegularUserID*)**

**Song Relation:**

**Represented as 𝑅(Title, Year, Length, Genre, *UploadUserID, CreateSuggestionUserID, PurchaseUserID, CreateSongArtistID, CreateSongAlbumNumber*)**

**Artist Relation:**

**Represented as 𝑅(ArtistID, DateOfBirth, ArtName, URL, *CreateSongArtistID*)**

**Album Relation:**

**Represented as 𝑅(AlbumNumber)**

**Hit Relation:**

**Represented as 𝑅(HitID, TrackNumber, *SongTitle, SongYear, SongLength, SongGenre*)**

**Suggestion Relation:**

**Represented as 𝑅(SuggestionID, *CreateSuggestionUserID*)**

**Q7. Review the case study on the “Local Library Management System”. Based on the details provided, answer the following:**

**(1) Determine the normal form in which the “library” table currently resides.**

**Ans :**

The "Library" table is currently in the first normal form (1NF) due to its atomic values and absence of repeating groups. Despite this, further normalization is required to eliminate redundancy and enhance data integrity. Progressing towards higher normal forms, like second normal form (2NF) and third normal form (3NF), involves organizing data into separate tables and establishing relationships for optimal database structure.

**(2) Enumerate all functional dependencies found in the “library” table. Note: In this scenario, a book has only one author.**

**Ans :**

The functional dependencies found in the "library" table can be enumerated as follows:

* **MemberID -> MemberName, Address**
* **BookID -> BookTitle, Author,LibrarySection**
* **BorrowDate, ReturnDate -> Librarian**
* **BookID -> LibrarySection**
* **LibrarySection-> Librarian**

**(3) From the functional dependencies, identify any partial dependencies**

**Ans :**

The candidate key is (MemberID, BookID) lets identify partial dependencies using this candidate keys :

* **MemberID -> MemberName, Address**: If MemberID = 123 corresponds to a specific library member, the functional dependency indicates that the MemberName and Address for that member can be determined solely based on their MemberID.
* **BookID -> BookTitle, Author, Librarian, LibrarySection**: if BookID = CHMOD corresponds to a specific book in the library, the functional dependency suggests that the BookTitle, Author, Librarian, and LibrarySection for that book can be determined solely based on its BookID.

These partial dependencies arise because certain attributes are functionally dependent on only a part of the candidate key, not the entire key. In a well-designed database, such dependencies should be minimized through normalization to reduce redundancy and potential update anomalies.

**(4) From the functional dependencies, identify any transitive dependencies.**

**Ans :**

Here, below are transitive dependencies because both side attributes are non-prime in the above function dependency.

BookTitle -> Author, LibrarySection

Librarian -> LibrarySection

The determination of Author and LibrarySection is not directly dependent on BookTitle or Librarian, but rather through the intermediary attribute LibrarySection. This indirect dependency classifies them as transitive dependencies.

**(5) Transform the current “library” relation into relations in the third normal form (3NF). Clearly indicate the primary key for each new relation. Additionally, specify any foreign keys and their referencing table.**

**Ans :**

* To transform the current "library" relation into relations in the third normal form (3NF), we need to analyze the given data and organize it into separate tables to reduce redundancies and inconsistencies. The primary key for each new relation should be clearly indicated, and any foreign keys and their referencing tables should be specified.
* **Members Table:**
  + Primary Key: MemberID
  + Attributes: MemberName, Address
* **Books Table:**
  + Primary Key: BookID
  + Attributes: BookTitle, Author, LibrarySection
  + Foreign Key: MemberID (referencing Members Table)
* **Borrowings Table:**
  + Composite Primary Key: (MemberID, BookID)
  + Attributes: BorrowDate, ReturnDate, Librarian
* **Authors Table:**
  + Primary Key: AuthorID
  + Attributes: AuthorName
* **BooksAuthors Table** (Junction Table for Many-to-Many Relationship between Books and Authors):
  + Composite Primary Key: (BookID, AuthorID)
  + Foreign Keys: BookID (referencing Books Table), AuthorID (referencing Authors Table)

In this transformation, the "library" relation has been normalized into separate tables to adhere to the third normal form. Each table has a clearly defined primary key, and foreign keys have been specified to establish relationships between the tables.