Elle Nguyen – Final Exam – Question 1, 2, 4, 5

**Q1:**

A diagram of a process

Description automatically generated

**Q2:**

A diagram of a computer code

Description automatically generated with medium confidence

A white rectangular grid with black text

Description automatically generated

**Q4:** This table captures the data about authors, books, and publishers. In addition, it shows the per-book royalty amount of an individual author. Each author has a unique author id, last name, first name, and an institution to which he/she is affiliated. Each book has a unique book id, a title, and a price. Each publisher has a unique name and a city where it is headquartered. Each publisher has exactly one headquarter. Each author can write at least one book and could write multiple books. Each book is written by at least one author and could be written by multiple authors. Each book is published by exactly one publisher. Each publisher can publish at least one book, and could publish many books.

**a)** Using the table AUTHERBOOKMIX, describe an example of an insertion anomaly, a deletion anomaly, and a modification anomaly.

- **Insertion anomaly**: Cannot insert new BookID B7 without inserting an actual author’s information writing such book.

- **Deletion anomaly**: Cannot delete AuthorID A5 without also deleting all the data about their 2 books “Introduction to Algorithms” and “Algorithms Unlocked” as well as the publisher “The MIT Press.”

- **Modification anomaly**: To change the book publisher from “The MIT Press” to “Cambridge Academy”, three records have to be modified.

**b)** What are the types of the following functional dependencies (full, partial, and transitive)?

|  |  |
| --- | --- |
| **Functional Dependencies** | **Type** |
| AuthorID AuthorLastName, AuthorFirstName, AuthorInstitution | Partial |
| BookID BookTitle, BookPrice, BookPublisher, PublisherCity | Partial |
| BookPublisher PublisherCity | Transitive |
| AuthorID, BookID AuthorBookRoyalty | Full |

**c)** Is the table in 1NF? If not, normalize it to 1NF.

Yes, because each row is unique and no column in any row contains multiple values.

**d)** Show the result of normalizing the table to 2NF.

Remove partial function dependencies:

(P) AuthorID AuthorLastName, AuthorFirstName, AuthorInstitution

Create new table **AUTHORS**:

(F) **AuthorID (PK)** | AuthorLastName | AuthorFirstName |AuthorInstitution

Remove AuthorLastName, AuthorFirstName, AuthorInstitution from original table

(P) BookID BookTitle, BookPrice, BookPublisher, PublisherCity

Create new table **BOOKS**:

(F) **BookID (PK)** | BookTitle | BookPrice | BookPublisher | PublisherCity

Remove BookTitle, BookPrice, BookPublisher, PublisherCity from original table

Resulting table in 2NF:

(F) AUTHORS: **AuthorID (PK)** | AuthorLastName | AuthorFirstName |AuthorInstitution

(F) BOOKS: **BookID (PK)** | BookTitle | BookPrice | BookPublisher | PublisherCity

(F) AUTHOR\_BOOK\_MIX: **AuthorID (PK)** | **BookID** **(PK)** | AuthorBookRoyalty

**e)** Show the result of normalizing the table to 3NF.

Remove transitive function dependencies:

(T) BookPublisher PublisherCity

Create new table **BOOK\_PUBLISHER**:

(F) **BookPublisher (PK)** | PublisherCity

Remove PublisherCity from **BOOKS**: **BookID (PK)** | BookTitle | BookPrice | BookPublisher

Resulting table in 3NF:

(F) AUTHORS: **AuthorID (PK)** | AuthorLastName | AuthorFirstName |AuthorInstitution

(F) BOOKS: **BookID (PK)** | BookTitle | BookPrice | BookPublisher

(F) BOOK\_PUBLISHER: **BookPublisher (PK)** | PublisherCity

(F) AUTHOR\_BOOK\_MIX: **AuthorID (PK)** | **BookID** **(PK)** | AuthorBookRoyalty

**Q5:** Given the following transaction schedule, draw its precedence graph, decide if it is conflict serializable. If it is conflict serializable, list a serializability order of the transactions based on the precedence graph. If it’s not conflict serializable, explain why it’s not.

T1: W(D), T5: R(F), T4: W(E), T5: R(D), T3: W(F), T2: R(H), T4: R(D), T6: R(F), T3: R(E), T1: R(D), T6: W(H), T2: R(E)

A diagram of a diagram

Description automatically generated

A serializability order of transactions is T1, T4, T2, T6, T3, T5.