Solution to Practice Exercises

Please send an email for any modifications if needed at: faisal.alvi@sse.habib.edu.pk

8.18. Consider the LIBRARY relational database schema shown in Figure 8.14, which is used to keep track of books, borrowers, and book loans. Referential integrity constraints are shown as directed arcs in Figure 8.14, as in the notation of Figure 5.7. Write down relational expressions for the following queries:

a. How many copies of the book titled *The Lost Tribe* are owned by the library branch whose name is 'Sharpstown'?

b. How many copies of the book titled *The Lost Tribe* are owned by each library branch?

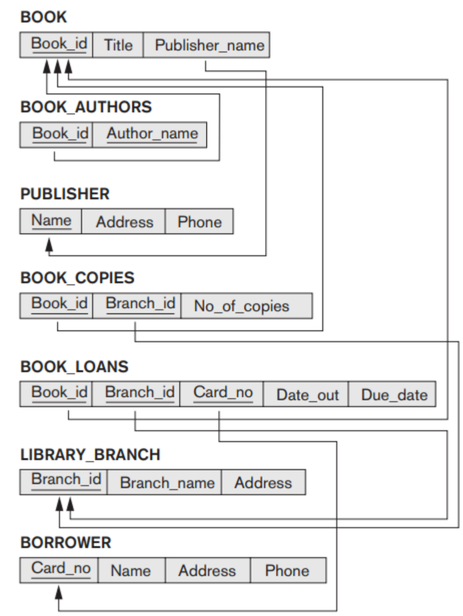
c. Retrieve the names of all borrowers who do not have any books checked out.

d. For each book that is loaned out from the Sharpstown branch and whose Due_date is today, retrieve the book title, the borrower's name, and the borrower's address.

e. For each library branch, retrieve the branch name and the total number of books loaned out from that branch.

f. Retrieve the names, addresses, and number of books checked out for all borrowers who have more than five books checked out.

g. For each book authored (or coauthored) by Stephen King, retrieve the title and the number of copies owned by the library branch whose name is Central.



(a) How many copies of the book titled *The Lost Tribe* are owned by the library branch whose name is 'Sharpstown'?
[Note: There can be multiple correct answers]

$$\Pi_{No\_of\_copies}\,(\sigma_{Title = \text{"The Lost Tribe"} \wedge Branch\_name = \text{"Sharpstown"}}\,(\,(Book) \bowtie_{Book.Book\_id = Book\_Copies.Book\_id} (Book\_Copies) \bowtie_{Book\_Copies.Branch\_id = Library\_Branch.Branch\_id} (Library\_Branch)\,)$$

(b) How many copies of the book titled *The Lost Tribe* are owned by each library branch?

$$\Pi_{No\_of\_copies,\ Branch\_name}\,(\sigma_{Title = \text{"The Lost Tribe"}}\,(\,(Book) \bowtie_{Book.Book\_id = Book\_Copies.Book\_id} (Book\_Copies) \bowtie_{Book\_Copies.Branch\_id = Library\_Branch.Branch\_id} (Library\_Branch)\,)$$

(c) Retrieve the names of all borrowers who do not have any books checked out.

$$\Pi_{Name}\,(\sigma_{Date\_out = NULL \wedge Due\_Date = NULL}\,(\,(Borrower) \mathbin{\text{⟕}}_{Borrower.Card\_No = Book\_Loans.Card\_No} (Book\ Loans)\,)\,)$$

(d) For each book that is borrowed from the Sharpstown branch, and whose Due_Date is today, retrieve the book title, the borrower's name, and the borrower's address.

Assume that the variable TODAY represents today's date, i.e. TODAY ← 06/10/2022

TODAY ← 06/10/2022,

$$\text{Cards\_Due\_Today} \leftarrow \sigma_{\text{Due\_Date = TODAY}} \left((\text{Borrower}) \bowtie_{\text{Borrower.Card\_No = Book\_Loans.Card\_No}} (\text{Book\_Loans}) \right)$$

$$\text{Books\_Sharpstown} \leftarrow \sigma_{\text{Branch\_name = ``Sharpstown''}} \left((\text{Book}) \bowtie_{\text{Book.Book\_id = Book\_Copies.Book\_id}} (\text{Book\_Copies}) \bowtie_{\text{Book\_Copies.Branch\_id = Library\_Branch.Branch\_id}} (\text{Library\_Branch}) \right)$$

$$\Pi_{\text{Title, Name, Address}} (\text{Cards\_Due\_Today} \bowtie_{\text{Card\_Due\_Today.Book\_id = Book\_Sharpstown.Book\_id}} \text{Books\_Sharpstown})$$

(e) For each library branch, retrieve the branch name and the total number of books loaned out from that branch.

This requires the use of a function such as ‘AGGREGATE’ in SQL, which we did not cover in Relational Algebra.

(f) Retrieve the names, addresses and the number of books checked out for all borrowers who have more than 5 books checked out.

This requires the use of a function such as ‘COUNT’ in SQL, which we did not cover in Relational Algebra.

(g) For each book authored or coauthored by Stephen King, retrieve the title and the number of copies owned by the library branch whose name is ‘Central’. [Slightly different solution]

$$\Pi_{\text{Title, No\_Of\_Copies}} (\sigma_{\text{Author\_Name = ``Stephen King''} \wedge \text{Branch\_Name = ``Central''}}$$

$$(\text{Book\_Authors} \bowtie_{\text{Book\_Authors.Book\_id = Books.Book\_id}} \text{Book}$$

$$\bowtie_{\text{Book.Book\_id = Book\_Copies.Book\_id}} \text{Book\_Copies}$$

$$\bowtie_{\text{Book\_Copies.Branch\_id = Library\_Branch.Branch\_id}} \text{Library\_Branch})$$

14.29. Consider the following relations for an order-processing application database at ABC, Inc.

ORDER (<u>O#</u>, Odate, Cust#, Total_amount)
ORDER_ITEM(<u>O#</u>, <u>I#</u>, Qty_ordered, Total_price, Discount%)

Assume that each item has a different discount. The Total_price refers to one item, Odate is the date on which the order was placed, and the Total_amount is the amount of the order. If we apply a natural join on the relations ORDER_ITEM and ORDER in this database, what does the resulting relation schema RES look like? What will be its key? Show the FDs in this resulting relation. Is RES in 2NF? Is it in 3NF? Why or why not? (State assumptions, if you make any.)

Solution:

A natural join on the relations ORDER and ORDER_ITEM would make the resulting schema RES look as follows:

RES (<u>O#</u>, <u>I#</u>, Odate, Cust#, Qty_ordered, Total_price, Discount%, Total_amount)

The key will be <u>O#</u>, <u>I#</u>.

The following functional dependencies are there in this relation:

O# → Odate [Reason: Each order has a date]

O# → Cust# [Reason: Each order is by one customer]

O# → Total_amount [Reason: Each order has a total amount]

I# → Discount% [Assumption: The store has a particular discount% on each item. Given that the Item# is known, the discount% can be extracted]

I#, Qty_Ordered, Discount% → Total_price [Assumption: If the Item# and its Qty_Ordered and the Discount% is known then the Total_price can be calculated for the item]

Due to dependencies such as O# → Odate, RES is not in 2NF. Clearly, it is not in 3NF too.