DBMS Lab 6,7

Maniya Yash Rajeshbhai
B20CS033

BOOK

Author_ID	Book_ID	Author_Name	Book
An_Ch_0103	Aest_AC_0103	Anjan Chatterjee	The Aesthetic Brain
An_Da_0104	Self_AD_0104	Antonio Damasio	Self Comes to Mind
Ca_Sa_0319	Beyo_CS_0319	Carl Safina	Beyond Words: What Animals Think and Feel
Ca_Sa_0319	Song_CS_0319	Carl Safina	Song for the Blue Ocean
Jo_Ro_1018	Deat_JR_1018	Joanne K. Rowling	Deathly Hallows_Harry Potter
Jo_Ro_1018	Fant_JR_1018	Joanne K. Rowling	Fantastic Beasts and Where to Find Them
Jo_Ro_1018	Gobl_JR_1018	Joanne K. Rowling	Goblet of Fire_Harry Potter
Jo_Ro_1018	Phil_JR_1018	Joanne K. Rowling	Philosopher's Stone_Harry Potter
Jo_Ro_1018	Pris_JR_1018	Joanne K. Rowling	Prisoner of Azkaban_Harry Potter
La_Ch_1203	Mind_LC_1203	Lars Chittka	The Mind of a Bee
Ma_Mi_1313	Emot_MM_1313	Marvin Minsky	Emotion Machine
Ma_Mi_1313	Soci_MM_1313	Marvin Minsky	Society of Mind
Pe_Wo_1623	Aunt_PW_1623	Pelham G. Wodehouse	Aunts Aren't Gentlemen
Pe_Wo_1623	Wode_PW_1623	Pelham G. Wodehouse	Wodehouse at the Wicket
Vi_Ra_2218	Emer_VR_2218	Vilayanur Ramachandran	The Emerging Mind
Vi_Ra_2218	Phan_VR_2218	Vilayanur Ramachandran	Phantoms in the Brain

AUTHOR

Author_ID	Author_Name
An_Ch_0103	Anjan Chatterjee

An_Da_0104	Antonio Damasio
Ca_Sa_0319	Carl Safina
Jo_Ro_1018	Joanne K. Rowling
La_Ch_1203	Lars Chittka
Ma_Mi_1313	Marvin Minsky
Pe_Wo_1623	Pelham G. Wodehouse
Vi_Ra_2218	Vilayanur Ramachandran

BOOK PURCHASE

Book_ID	Book	Purchase_Date	Quantity
Aest_AC_0103	The Aesthetic Brain	Sep 5, 2022	1
Self_AD_0104	Self Comes to Mind	Sep 5, 2022	1
Beyo_CS_0319	Beyond Words: What Animals Think and Feel	Sep 5, 2022	2
Song_CS_0319	Song for the Blue Ocean	Sep 6, 2022	2
Deat_JR_1018	Deathly Hallows_Harry Potter	Sep 7, 2022	5
Fant_JR_1018	Fantastic Beasts and Where to Find Them	Sep 6, 2022	5
Gobl_JR_1018	Goblet of Fire_Harry Potter	Sep 5, 2022	5
Phil_JR_1018	Philosopher's Stone_Harry Potter	Sep 5, 2022	5
Pris_JR_1018	Prisoner of Azkaban_Harry Potter	Sep 5, 2022	5
Mind_LC_1203	The Mind of a Bee	Sep 6, 2022	2
Emot_MM_1313	Emotion Machine	Sep 5, 2022	1
Soci_MM_1313	Society of Mind	Sep 6, 2022	1
Aunt_PW_1623	Aunts Aren't Gentlemen	Sep 7, 2022	4
Wode_PW_1623	Wodehouse at the Wicket	Sep 5, 2022	4
Emer_VR_2218	The Emerging Mind	Sep 5, 2022	1
Phan_VR_2218	Phantoms in the Brain	Sep 6, 2022	3

Q1.

a) Is the given database normalized? Would you want to normalize the given database any further?

No, the database is not normalized as it is in 2NF and we can normalize it to BCNF.

The functional dependencies of the tables are:
1.Book : Book_Id->Book, Author_Id, Author_Name, Book->Author_Id
2.Author : Author_ID->Author_Name
3.Book Purchase : Book_Id->Purchase Date, Book, Quantity

- Primary keys:

Book : Book_Id
Author : Author_Id
Book Purchase : Book_Id

- Normalisation:

The given tables are in the following normal forms:
1NF : As no multivalued attributes are present
2NF : As no partial dependencies are there

The database is not in 3NF as there is a transitive dependency in the Book table.
Book_Id -> Book -> Author

Here are the Normalized tables :

BOOK

Book_ID	Book
Aest_AC_0103	The Aesthetic Brain
Self_AD_0104	Self Comes to Mind
Beyo_CS_0319	Beyond Words: What Animals Think and Feel
Song_CS_0319	Song for the Blue Ocean
Deat_JR_1018	Deathly Hallows_Harry Potter
Fant_JR_1018	Fantastic Beasts and Where to Find Them

AUTHOR

Author_ID	Author_Name
An_Ch_0103	Anjan Chatterjee
An_Da_0104	Antonio Damasio
Ca_Sa_0319	Carl Safina
Jo_Ro_1018	Joanne K. Rowling
La_Ch_1203	Lars Chittka
Ma_Mi_1313	Marvin Minsky
Pe_Wo_1623	Pelham G. Wodehouse
Vi_Ra_2218	Vilayanur Ramachandran

PURCHASE

Book_ID	Purchase_Date	Quantity
Aest_AC_0103	Sep 5, 2022	1
Self_AD_0104	Sep 5, 2022	1
Beyo_CS_0319	Sep 5, 2022	2
Song_CS_0319	Sep 6, 2022	2

BOOK-AUTHOR

Book_ID	Author_ID
Aest_AC_010	An_Ch_0103
Self_AD_0104	An_Da_0104
Beyo_CS_031	Ca_Sa_0319
Song_CS_031	Ca_Sa_0319
Deat_JR_1018	Jo_Ro_1018
Fant_JR_1018	Jo_Ro_1018

The given tables are in the following normal forms:

1. 1NF--As no multivalued attributes are present
2. 2NF--As no partial dependencies are there
3. 3NF--No transitive dependencies are present
4. BCNF-- As LHS of all dependencies is a candidate key
5. 4NF-- As no multivalued dependencies are present
5. 5NF--As no lossless decompositions are there(we can check by decomposing book_detaills and purchase_details)

b) Identify the different indexes that you would use for the tables in the database. Justify your answer?

1. In the Book table, since Book_ID is the primary key there, it is chosen for hashing and obtaining the address. The main reason behind it was the uniqueness of the Primary key. Since it can be used to identify unique tuples, if it is used for indexing, the chances of collision will be minimal.
2. In the Author table also, since Author_Id can be used to uniquely determine the tuples, it is chosen as an index for the hashing purpose.
3. For the Book Purchase table, Book_Id is chosen for hashing for the same reason as it can be used to uniquely identify the tuples so choosing it will minimize the collision hence improve the insertion and fetching time.

Q2, 3.

Asked 5-bucket extendible/linear hash + doubly linked lists are already submitted in a zip file.

a) **Retrieve names of books written by 'Carl Safina' :**
Relational Algebra Expressions :

a. $\Pi_{Book}\left(\sigma_{Author\_Name = 'Carl\ Safina'}(Book)\right)$

b) **Retrieve book names and author details of all books written by authors with names beginning with 'A' or 'P' :**
Relational Algebra Expressions :

b. $Auth \leftarrow \sigma_{Author\_Name[0]='A' \text{ or } Author\_Name[0]='P'}(Author)$

Ans : $\Pi_{Book, Author\_Id, Author\_Name}\left(Book \bowtie Auth\right)$

c) Retrieve all books with >= 5 copies :

Relational Algebra Expressions :

c. $X \leftarrow \sigma_{quantity \geq 5}(Book\ Purchase)$

$Book\_Ids \leftarrow \Pi_{Book\_Id}(X)$

$Ans : \Pi_{Book}(Book \bowtie Book\_Ids)$

d) Retrieve author_names whose books have been purchased across all dates available on the purchase table :

Relational Algebra Expressions :

d. $Dates \leftarrow \Pi_{Purchase\_Date}(Book\ Purchase)$

$Req\_Books \leftarrow (Book\ Purchase) \div (Dates)$

$Ans : \Pi_{AuthorName}(Author \bowtie (Book \bowtie Req\_Books))$

e) For the books that have just a single copy in the database, retrieve the author_names :

Relational Algebra Expressions :

e. $Singles \leftarrow \Pi_{Book\_id}(\sigma_{Quantity=1}(Book\ Purchase))$

$Ans : \Pi_{Author\_Name}(Singles \bowtie Author)$

Q4.

Compare between execution times for all queries in Q.2 and Q.3
[Comment on the role of the data-structures used, and complexity of the different query plans.]

For 5-bucket extendible hash + doubly linked lists :

Execution time for different query plan for all the queries :

Subpart	Execution Time
a.	17 microseconds
b.	18 microseconds
c.	30 microseconds
d.	52 microseconds
e.	23 microseconds

For 5-bucket linear hash + doubly linked lists :

Execution time for different query plan for all the queries :

Subpart	Execution Time
a.	15 microseconds
b.	14 microseconds
c.	27 microseconds
d.	46 microseconds
e.	19 microseconds

Observations :

1. The first thing to observe here is since there's no spike in execution time, that concludes there was no bucket split that has taken place in accordance to the hash functions made for linear and extendible hashing.
2. As observed from the query tree, all the optimized queries where join operation took place after filtering the result, the time of execution is observed less than that where join operation was performed beforehand.

Time and Space complexities

1. For all the subparts , for the naive query (query (i)) , at first the join operation was performed which took O(m*n) space where m and n are the sizes of the two tables on which join operation was performed.
2. In terms of time also, the naive way took O(m*n) to perform join and since the new table has size of m*n, the linear searching time was also quadratic , i.e O(m*n).
3. For optimized queries where at first the searching was performed and then join operation, in worst case still the complexity will be quadratic where all the tuples on one table will be selected for the join operation with the other table, but on average, since the searching is now reduced to linear which incase also reduces the size of one table with only specific results to join with other, the average retrieval time will be much optimized and this is also valid for space complexity as well.