Tutorial: Data Warehousing

Students at the National University of Ngendipura (NUN) buy books for their studies. They also lend and borrow books to and from other students. Your company, **Apasaja Private Limited**, is commissioned by NUN Students Association (NUNStA) to implement an online book exchange system that records information about students, books that they own and books that they lend and borrow.

The database records the name, faculty, and department of each student. Each student is identified in the system by his/her email. The database also records the date at which the student joined the university. If a student has graduated, the database record the date of graduation. A department in NUN must belong to exactly one faculty.

The database records the title, authors, publisher, language, year as well as the ISBN-10 and ISBN-13 for each book. A book can have several authors but it must have at least one author. The database also records author that currently has no book. It should also record the format of the book (i.e., if the book is hardcover or softcover). The International Standard Book Number, ISBN-10 or -13, is an industry standard for the unique identification of books. It is possible that the database records books that are not owned by any students (e.g., because the owners of a copy graduated or because the book was advised by a lecturer for a course but not yet purchased by any student).

A student may own multiple copy of the same book. We differentiate the copy by its copy number. For instance, John may own two copies of the book Database Systems with ISBN-13 number of 9780131873254. The first copy has a copy number of 1 while the second copy has a copy number of 2. The copy number should be a consecutive number starting from 1.

The database also records the date at which a book copy is borrowed and the date at which it is returned. We refer to this information as a loan record. Obviously, a student can only borrow or lend book after he/she is enrolled.

For auditing purposes the database records information about the books, the copies and the owners of the copies as long as the owners are students or as there are loan records concerning the copies. For auditing purposes the database records information about graduated students as long as there are loan records concerning books that they owned.

Questions

Not all questions will be discussed during tutorial. You are expected to attempt them before coming to the tutorial. You are encouraged to discuss them on Canvas Discussion.

1. **OLTP**.

   Before we begin, we need to load the data for the NUNStA database. We define "loan duration" as one of the following.

   - If the item has been returned, then it is the "borrowed date" subtracted from the "returned date" (i.e., `rd_date - bd_date`).
   - If the item has not been returned, then it is the "borrowed date" subtracted from the "current date" (i.e., `CURRENT_DATE - bd_date`).

   (a) For each owner faculty and borrower faculty pair, find the average duration, and the longest duration such that the owner is from the different faculty than the borrower.

   Output the owner faculty, borrower faculty, average duration, and longest duration. Round the average duration to 2 decimal places using `ROUND(val, dp)` . Sort the output in descending order of average duration. See the sample run below.

owner_faculty	borrower_faculty	average	longest
Faculty of Arts and Social Science	Faculty of Engineering	74.89	869
School of Computing	Faculty of Engineering	68.74	923
Faculty of Science	School of Computing	66.23	853
:	:	:	:
Faculty of Science	Faculty of Engineering	50.09	848
Faculty of Arts and Social Science	School of Computing	49.60	966
Faculty of Engineering	Faculty of Science	41.86	700

   Comment on the complexity of the query.

   (b) For each book, find the student(s) that borrow the book for the longest duration. Output the book title and the name of the student. Sort the result in ascending order of book ISBN13 and descending order of student email.

   Multiple students may borrow a book for the longest time if they have the same longest time. In such cases, output all such students with the longest duration for the book. See the sample run below.

title	name
How to Score 100 in Database	PRABAWA ADI
A Beginner s Guide to Constructing the Universe: ...	ZENG YIHUI
Krakatoa: The Day the World Exploded: ...	TAN CHENG HAN
:	:
Indian Heroes and Great Chieftains (Large Print)	QIN YUWEI
The Law	GE DUO
Introduction to Nuclear And Particle Physics: ...	ZHANG YUZHAO

   Comment on the speed of the query.

2. **Transform**.

   Now, we want to transform our OLTP database into an OLAP database. We want to support the two queries from Q1. Transform the NUNStA database into a star schema following dimensional modelling.

   Note that you can insert into a table using query. For instance, if we want to create a copy of a department, we can use the following statement.

Code: INSERT with Query

```
CREATE TABLE wh_department (
  d_dept VARCHAR(32) PRIMARY KEY NOT NULL,
  d_faculty VARCHAR(62) NOT NULL
);
INSERT INTO wh_department (
  SELECT d_name, d_faculty
  FROM departments
);
```

3. **OLAP**.

   Now that we have the star schema, we can redo Q1.

   (a) For each owner faculty and borrower faculty pair, find the average duration, and the longest duration such that the owner is from the different faculty than the borrower.

   Output the owner faculty, borrower faculty, average duration, and longest duration. Round the average duration to 2 decimal places using `ROUND(val, dp)`. Sort the output in descending order of average duration.

   Comment on the complexity of the query.

   (b) For each book, find the student(s) that borrow the book for the longest duration. Output the book title and the name of the student. Sort the result in ascending order of book ISBN13 and descending order of student email.

   Multiple students may borrow a book for the longest time if they have the same longest time. In such cases, output all such students with the longest duration for the book.

   Comment on the speed of the query.

References

[1] S. Bressan and B. Catania. *Introduction to Database Systems*. McGraw-Hill Education, 2006. ISBN: 9780071246507.

[2] Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom. *Database Systems: The Complete Book*. 2nd ed. Prentice Hall Press, 2008. ISBN: 9780131873254.

[3] R. Kimball. *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*. Wiley, 1996. ISBN: 9780471153375.

[4] Raghu Ramakrishnan and Johannes Gehrke. *Database Management Systems*. 2nd. USA: McGraw-Hill, Inc., 2000. ISBN: 0072440422.

[5] *W3schools Online Web Tutorials*. https://www.w3schools.com/. [Online; last accessed 2025].