Final Project Report

Team 18

Members: Tae Yoon Choi, Jae Kwan Kim, Ji Hoon Ban, Youngjae Shin

Date: December 11, 2019

<u>Project Summary</u>

This project is to manage the library data focusing on the relationship of different information: book, issue/return, and student. To manage the data, it requires multiple relational programs to implement operations using aggregates, functions, triggers, views, etc. The main point of this project is to create tables and functions so that librarian can manage the data much easier, using specific targeted functions.

<u>Project progress</u>

First, we sketched the tables that would be used for this project including categorizing data, tables format, information that will be handled, setting constraints, and primary and foreign keys.

Second, we created functions, aggregates, views, and triggers to manage data created on the previous step. At this point, we had to think about what functions are needed to manage different types of data and what information would be needed for librarian. What we think is the most important functions was inserting and replacing data in tables.

Lastly, we updated more functions, triggers to develop programs that we built on previous step. On this progress, not only adding more functions for easier user-interactive programs but also simplifying functions for less time complexity was considered.

Function Accomplishments and Explanation

Relational Functions and Triggers

Insertbook function is when you insert book data into bookinfo table, new book id would be inserted in bookid table with newly inserted ISBN in bookinfo table. For example, when you insert into bookinfo values(13524, 'Database Concept',1,'Sabry',1,5), it will make this trigger to insert 5 book_ids with isbn=13524 in bookid table. That's why we use loop in the function.

Checkshelf function is to check what shelf is the book located given book title. With relevant booktitle, it will automatically find the bookshelf number and returns what shelf is the book located at.

Availiable function is to check how many books are left in the library. Input is ISBN so the function return bigint.

Burrowedby function put ISBN and return id, name, issued_date, and return date with given ISBN number. This function is to check who borrowed such book with associated ISBN.

Issueabook function is to insert data of issuing books into bookissue table. Only book_id and student_id are required. Other fields will be automatically updated. Also, if the book is already borrowed by someone and has not been returned, this function will throw an error message.

Returnabook function updates return_status to true when it runs along the associated data. This function is to update bookissue table return_status to be true when a student returns a book. Once a function is used, it will automatically update the table that is relevant to data input or if nothing matches or is already true, it will not change or update anything.

updateoverdue function is to call once a day; therefore, user should call this function before library close in order to update the balance of the people who did not return the book. Update overdue function will not always update balance in case return date is not yet passed. However, once the return date is passed, it will update the data and add the balance 0.25 to student who did not return a book in due date.

longtermoverdue function is to see people who still need to return a book and their balance. Therefore, this function returns student_id, student_name, book_id, book_title, issued_date, return_date, and balance to look at the information from the students who requires to return a book and those who still have balance to pay.

payoverdue is a function to decrease balance of each student who paid the overdue charges, with given student_id and amount of charges they paid. We set the restriction on this function not to let students pay more than their charges. If student does not have overdue or trying to pay more than overdue in the balance, it raises errors. For example, when you select the function pay overdue (3,0.25) which is Id called 3, the balance from the Id 3 is decreased 0.25

balance. That is why we used if status to figure out of the id as that is paid the amount of the balance, then decreased the balance which ID paid by.

Returnrecord function is somewhat similar to longtermoverdue function but different on the point that this function does not care about the balance of students. This function only cares about student who still need to return a book regardless of the due date. It will returns the record with student name, bookid, booktitle, and return date to check if the due date has been passed.

Overduerecord function is to check who still need to pay their overdue charges. It will simply show the student information with the balance they need to charges. It will let the student and user to check the balance easily and be aware of the charges. This function is required before payoverdue function runs to avoid student pay more than what they have in their balance.

Project Issues and Solutions

Before making the update overdue, long term overdue, payoverdue and overdue record functions, we actually tried to create triggers that runs altogether to update the overdue balance automatically. The trigger was supposed to update balance by 0.25 once the due date has passed. However, we had problems on creating these triggers because if the exact date is not established, the triggers would not work correctly to increase the balance. Also because of complicated and multiple if statements we used in triggers, it does not run and throws multiple errors. The if statement we wanted to use was to check overdue and update balance to increase by 0.25 automatically.Since there were multiple errors thrown by cramming many functions into one trigger, we decided to separate each functions to be more specific on the purpose of use. Therefore, we divided one trigger to 4 different functions for easier use of the system and of the library to check the students' status everyday.

Resources

Garcia-Molina, H., Ullman, J. D., & Widom, J. (2014). *Database systems: the complete book* (2nd ed.). Harlow: Pearson Education Limited.