Assignment 2

In this assignment, you will practice working with Pure SQL. Your solutions, containing PostgreSQL statements for solving the problems, should be submitted to IUCanvas in a file Assignment2.sql. It is advised that you also include comments in this file to elaborate on your solutions.

Restrictions on SQL code: You can use views but you can not use the `GROUP BY` clause and aggregate functions. You can also not use the `INNER JOIN` (or other joins) operators. In other words, only statements in Pure SQL are permitted. Solutions with SQL statements that do not obey these requirements will not receive credit.

Use the files student.txt, majors.txt, book.txt, cites.txt, and buys.txt that are provided for this assignment.

Consider the following relation schemas about students and books.

Student($\underline{Sid}, Sname$)
Major($\underline{Sid, Major}$)
Book($\underline{BookNo}, Title, Price$)
Cites($\underline{BookNo, CitedBookNo}$)
Buys($\underline{Sid, BookNo}$)

The relation Major stores students and their majors. A student can have multiple majors but we also allow that a student has no major. major. A tuple (b, c) in the relation Cites indicates that the book with book number b cites the book with book number c. Note that a book may cite multiple other books. Also, a book does not have to cited.

The primary keys of the relations are the underlined attributes and we assume the following foreign keys:

Attribute in Relation	References Primary Key of Relation
Sid in Major	Sid in Student
BookNo in Cites	BookNo in Book
CitedBookNo in Cites	BookNo in Book
Sid in Buys	Sid in Student
BookNo in Buys	BookNo in Book

Furthermore, assume the following domains for the attributes:

Attribute	Domain
Sid	INTEGER
Sname	VARCHAR(15)
Major	VARCHAR(15)
BookNo	INTEGER
Title	VARCHAR(30)
Price	INTEGER
CitedBookNo	INTEGER

To do this assignment, you will have to create the above relations, including the primary and foreign keys, using the given domain types. You also need to insert the data given in the .txt files.

Write the following queries in SQL.

1. Find the sid and major of each student who bought a book that cost less than $20.
2. Find the bookno and title of each book that cost between $20 and $40 and that is cited by another book.
3. Find the sid and name of each student who majors in 'CS' and who bought a book that is cited by a lower priced book.
4. Find the bookno and title of each cited book that is itself cited by another book.
5. Use the SQL `ALL` predicate to find the booknos of the cheapest books.
6. Without using the SQL `ALL` or `SOME` predicates, find the booknos and titles of the most expensive books.
7. Find the booknos and titles of the second most expensive books.
8. Find the bookno and price of each book which, if it is cited by another book, cites a book that cost more than $20. (Note that each book which is not cited also satisfies the condition of the query and must therefore also be included in the answer.)
9. Find the bookno and title of each book that is bought by a student who majors in 'Biology' or in 'Psychology'.
10. Find the bookno and title of each book that is not bought by all students who major in 'CS'.
11. Find the bookno of each book that was only bought by students who major in 'Biology'.
12. Find the bookno and title of each book that is bought by all students who major in both 'CS' and in 'Math'.
13. Find the sid and name of each student who not only bought books that were bought by at least two 'CS' students.
14. Find the sid and name of each student who bought at most one book that cost more than $20.
15. Find each (s, b) pair where s is the sid of a student and where b is the bookno of a book whose price is the cheapest among the books bought by that student.