**Chapter Seven Questions:**

1. Write a query to display the different years in which books have been published. Include each year only once and sort the results by year (Figure P7.60). (P60)

SELECT DISTINCT Book\_Year FROM BOOK ORDER BY Book\_Year;

文本

低可信度描述已自动生成

1. Write a query to display the different subjects on which FACT has books. Include each subject only once and sort the results by subject (Figure P7.61). (P61)

SELECT DISTINCT Book\_Subject FROM BOOK ORDER BY Book\_Subject;

文本

中度可信度描述已自动生成

1. Write a query to display the book number, title, and cost for all books that cost $59.95 sorted by book number (Figure P7.65). (P65)

SELECT Book\_Num, Book\_Title, Book\_Cost FROM BOOK

WHERE Book\_Cost = 59.95

ORDER BY Book\_Num;

表格

描述已自动生成

1. Write a query to display the patron ID, first and last name of all patrons who are students, sorted by patron ID (Figure P7.72). (44 rows) (P72)

SELECT PAT\_ID, PAT\_FNAME, PAT\_LNAME FROM PATRON

WHERE UPPER(PAT\_TYPE) = "STUDENT"

ORDER BY PAT\_ID;

表格

描述已自动生成

1. Write a query to display the patron ID, first and last name, and patron type for all patrons whose last name begins with the letter “C,” sorted by patron ID (Figure P7.73). (P73)

SELECT \* FROM PATRON

WHERE LEFT(PAT\_LNAME,1) = 'C'

ORDER BY PAT\_ID;

表格

描述已自动生成

1. Write a query to display the author ID, first and last name of all authors whose year of birth is unknown. Sort the results by author ID (Figure P7.74). (P74)

SELECT AU\_ID, AU\_FNAME, AU\_LNAME FROM AUTHOR

WHERE AU\_BIRTHYEAR IS NULL

ORDER BY AU\_ID;

图示, 表格

中度可信度描述已自动生成

1. Write a query to display the number of books in the FACT system (Figure P7.78). (P78)

SELECT COUNT(\*) AS 'Number of Books' FROM BOOK;

图片包含 文本

描述已自动生成

1. Write a query to display the number of different book subjects in the FACT system (Figure P7.79). (P79)

SELECT COUNT(DISTINCT Book\_Subject) AS 'Number of Subjects' FROM BOOK;

文本

描述已自动生成

1. Write a query to display the number of books that are available (not currently checked out) (Figure P7.80). (P80)

SELECT COUNT(DISTINCT A.BOOK\_NUM)

FROM BOOK AS A

LEFT JOIN CHECKOUT AS B ON A.BOOK\_NUM = B.BOOK\_NUM

WHERE CHECK\_IN\_DATE IS NOT NULL OR CHECK\_NUM IS NULL;

图片包含 表格

描述已自动生成

1. Write a query to display the highest book cost in the system (Figure P7.81). (P81)

SELECT MAX(BOOK\_COST) FROM BOOK;

文本

中度可信度描述已自动生成

1. Write a query to display the lowest book cost in the system (Figure P7.82). (P82)

SELECT MIN(BOOK\_COST) FROM BOOK;

文本

中度可信度描述已自动生成

1. Write a query to display the number of different patrons who have ever checked out a book (Figure P7.83). (P83)

SELECT COUNT(DISTINCT(A.PAT\_ID)) AS 'DIFFERENT PATRONS'

FROM PATRON AS A

JOIN CHECKOUT AS B ON A.PAT\_ID=B.PAT\_ID

JOIN BOOK AS C ON B.BOOK\_NUM=C.BOOK\_NUM;

文本

描述已自动生成

1. Write a query to display the author ID and the number of books written by that author. Sort the results in descending order by number of books, then in ascending order by author ID (Figure P7.85). (P85)

SELECT A.AU\_ID, COUNT(B.BOOK\_NUM) AS 'Books Written'

FROM AUTHOR AS A

JOIN WRITES AS B ON A.AU\_ID = B.AU\_ID

JOIN BOOK AS C ON B.BOOK\_NUM=C.BOOK\_NUM

GROUP BY AU\_ID

ORDER BY COUNT(B.BOOK\_NUM) DESC, A.AU\_ID;

图片包含 文本

描述已自动生成

1. Write a query to display the patron ID, first and last name of all patrons who have never checked out any book. Sort the result by patron last name and then first name (Figure P7.101). (P101)

SELECT A.PAT\_ID, A.PAT\_FNAME, PAT\_LNAME

FROM PATRON AS A

LEFT JOIN CHECKOUT AS B ON A.PAT\_ID = B.PAT\_ID

WHERE CHECK\_NUM IS NULL

ORDER BY A.PAT\_LNAME, A.PAT\_FNAME;

表格

描述已自动生成

**Rick’s Questions:**

1. Using the Ch07\_LargeCo ER Diagram, draw a Venn diagram (as seen in the Relational Sets lecture) of employees who have worked for the “Accounting” department and who have worked for the “Marketing” department.

Marketing employees

Accounting employees

1. Write a word statement describing the overlapping area in the diagram.

The overlapping area represents employees that worked in both accounting department and marketing department.

1. Write an SQL query that will return only the names of those employees who appear in the overlapping region of the diagram. Note that you will not be able to test it in MySQL.

(SELECT A.EMP\_FNAME, A.EMP\_LNAME

FROM LGEMPLOYEE AS A

JOIN LGDEPARTMENT AS B ON A.DEPT\_NUM = B.DEPT\_NUM

WHERE DEPT\_NAME = 'ACCOUNTING')

INTERSECT

(SELECT C.EMP\_FNAME, C.EMP\_LNAME

FROM LGEMPLOYEE AS C

JOIN LGDEPARTMENT AS D ON C.DEPT\_NUM = D.DEPT\_NUM

WHERE DEPT\_NAME = 'MARKETING');

1. Write an SQL query that will return the names of all employees who have purchased items (are Customers too). Note that you will not be able to test it in MySQL.

SELECT B.EMP\_FNAME, B.EMP\_LNAME

FROM LGCUSTOMER AS A

JOIN LGEMPLOYEE AS B ON A.CUST\_FNAME = B.EMP\_FNAME AND A.CUST\_LNAME = B.EMP\_LNAME;