California State University, East Bay

Database Architecture

Libraries DB project2

Author:
Erwan Guyader

Supervisor:
Prof. James Daley

November 22, 2011

1. For each category, list the category name, the number of books owned in that category, and the first alphabetical title of books owned in that category. (Hint: use the min function.) Use table headings: Category, NumBooksOwned, FirstBook.

```
SELECT DISTINCT b1.category, (

SELECT count( * )
FROM books
WHERE category = b1.category
) AS NumBooksOwned, (

SELECT min( title )
FROM books
WHERE category = b1.category
) AS FirstBook
FROM books b1
```

category	NumBooksOwned	FirstBook
Business	4	Plastics
Consumer	4	"No Doc" Loans
Science	3	Raccoons
Travel	1	Hayward Auto Row

2. For each branch, list the branch name, the number of books owned at that branch in the Consumer category, the first alphabetical title of books owned at that branch in the Consumer category, and the length of the longest name of cardholders who have checked out a book owned at that branch in the Consumer category. Use column headings: Branch, NumBooksOwned, FirstBook, LongestCardholderNameLength.

```
SELECT DISTINCT b1.branch, (

SELECT count( * )
FROM books
WHERE branch = b1.branch
AND category = 'Consumer'
) AS NumBooksOwned, (

SELECT min( title )
FROM books
WHERE branch = b1.branch
AND category = 'Consumer'
) AS FirstBook, (

SELECT max( length( cardholder ) )
FROM books
WHERE branch = b1.branch
AND category = 'Consumer'
) AS LongestCardholderNameLength
FROM books b1
```

branch	NumBooksOwned	FirstBook	LongestCardholderNameLength
Bayfair	1	Same as Cash	NULL
Cal State	1	Living in State Parks	10
Downtown	2	"No Doc" Loans	15

3. For each branch, list the branch name, the number of books checked out at that branch, the first alphabetical title of books checked out at that branch, and the length of the longest name of cardholders who have checked out a book at that branch. Use column headings: Branch, NumBooksCheckedOut, FirstBook, LongestCardholderNameLength.

```
SELECT DISTINCT b1.branch, (

SELECT count( * )
FROM books
WHERE branch = b1.branch
AND cardholder IS NOT NULL
) AS NumBooksCheckedOut, (

SELECT min( title )
FROM books
WHERE branch = b1.branch
AND cardholder IS NOT NULL
) AS FirstBook, (

SELECT max( length( cardholder ) )
FROM books
WHERE branch = b1.branch
AND cardholder IS NOT NULL
) AS LongestCardholderNameLength
FROM books b1
```

branch	NumBooksCheckedOut	FirstBook	LongestCardholderNameLength
Bayfair	2	Hayward Auto Row	10
Cal State	4	Living in State Parks	15
Downtown	3	"No Doc" Loans	15

4. For each category, list the category name, the number of books owned in that category, and the average length of titles of the books owned in that category. List only categories for which more than one book is owned. Use category headings Category, NumBooksOwned, AvgLengthTitle.

```sql
SELECT DISTINCT b1.category, (

SELECT count( * )
FROM books
WHERE category = b1.category
) AS NumBooksOwned, (

SELECT avg( length( title ) )
FROM books
WHERE category = b1.category
) AS AvgLengthTitle
FROM books b1
```

category	NumBooksOwned	AvgLengthTitle
Business	4	11.2500
Consumer	4	17.0000
Science	3	6.0000
Travel	1	16.0000

5. Make a table of the number of books checked out by each cardholder. Include a row showing the number of books not currently checked out by anyone. Use column headings: Cardholder, NumCheckedOutBy.

```
(
SELECT DISTINCT b1.cardholder, (

SELECT count( * )
FROM books
WHERE cardholder = b1.cardholder
) AS NumCheckedOutBy
FROM books b1
WHERE b1.cardholder IS NOT NULL
)
UNION (

SELECT cardholder, (

SELECT count( * )
FROM books
WHERE cardholder IS NULL
)
FROM books
WHERE cardholder IS NULL
)
```

cardholder	NumCheckedOutBy
Bev Costco	3
Raul Felix	1
Sashoon Nusrat	2
Swanny Pottkins	2
Trin Prawn	1
NULL	3

6. Repeat the previous problem, but order the table in order of decreasing number of books. Use column headings: Cardholder, NumCheckedOutBy.

```sql
(
SELECT DISTINCT b1.cardholder , (

SELECT count( * )
FROM books
WHERE cardholder = b1.cardholder
) AS NumCheckedOutBy
FROM books b1
WHERE b1.cardholder IS NOT NULL
)
UNION (

SELECT cardholder , (

SELECT count( * )
FROM books
WHERE cardholder IS NULL
) AS NumCheckedOutBy
FROM books
WHERE cardholder IS NULL
)
ORDER BY 2 DESC
```

cardholder	NumCheckedOutBy
NULL	3
Bev Costco	3
Sashoon Nusrat	2
Swanny Pottkins	2
Raul Felix	1
Trin Prawn	1

7. For each branch and category, list the name of the branch, name of the category, number of books that are checked out by that branch in that category, and alphabetically first such book. Use column headings: Branch, Category, NumBooks, FirstBook. In the answer, you can omit rows where the number of books checked out is 0.

```sql
SELECT DISTINCT branch, category, (

SELECT count( cardholder )
FROM books
WHERE branch = b1.branch
AND category = b1.category
) AS NumBooks, (

SELECT min( title )
FROM books
WHERE branch = b1.branch
AND category = b1.category
AND cardholder IS NOT NULL
) AS FirstBook
FROM books b1
```

branch	category	NumBooks	FirstBook
Downtown	Consumer	2	"No Doc" Loans
Bayfair	Travel	1	Hayward Auto Row
Cal State	Consumer	1	Living in State Parks
Cal State	Business	2	Plastics
Bayfair	Science	1	Raccoons
Bayfair	Consumer	0	NULL
Downtown	Science	1	WD-40
Cal State	Science	1	WD-40
Downtown	Business	0	NULL
Bayfair	Business	0	NULL

8. For each category, list the category name, expert for that category, number of books owned in that category, and average title length of books owned in that category. Use column headings: Category, Expert, NumOwned, AvgLength.

```
SELECT categoryname, expert, (

SELECT count( * )
FROM books
WHERE category = categoryname
) AS NumOwned, (

SELECT avg( length( title ) )
FROM books
WHERE category = categoryname
) AS AvgLength
FROM categories
```

categoryname	expert	NumOwned	AvgLength
Business	Myrtle Grimes	4	11.2500
Consumer	Ed Fezner	4	17.0000
Science	Meg Molly Unruh	3	6.0000
Travel	Meg Molly Unruh	1	16.0000

9. List the titles of books checked out whose titles are longer than the average for books owned in their category. Use column headings: Title.

```
SELECT title
FROM books b1
WHERE length( title ) > (
SELECT avg( length( title ) )
FROM books
WHERE category = b1.category )
```

			title
☐	🖊	✗	You're Fired!
☐	🖊	✗	You're Fired!
☐	🖊	✗	Living in State Parks
☐	🖊	✗	Living in State Parks
☐	🖊	✗	Raccoons

10. For each branch and category, list the name of the branch, name of the category, number of books owned by the branch in that category, and the number of books checked out from the branch in that category. Use column headings: Branch, Category, NumOwned, NumCheckedOut. In the answer, you can omit rows where the number of books owned is 0.

```sql
SELECT DISTINCT branch, category, (

SELECT count( * )
FROM books
WHERE branch = b1.branch
AND category = b1.category
) AS NumOwned, (

SELECT count( cardholder )
FROM books
WHERE branch = b1.branch
AND category = b1.category
) AS NumCheckedOut
FROM books b1
```

branch	category	NumOwned	NumCheckedOut
Downtown	Consumer	2	2
Bayfair	Travel	1	1
Cal State	Consumer	1	1
Cal State	Business	2	2
Bayfair	Science	1	1
Bayfair	Consumer	1	0
Downtown	Science	1	1
Cal State	Science	1	1
Downtown	Business	1	0
Bayfair	Business	1	0

11. For each category, list the number of different titles (not books) owned in that category. Use column headings: Category, NumTitles.

```
SELECT DISTINCT category , (

SELECT count( DISTINCT title )
FROM books
WHERE category = b1.category
) AS NumTitles
FROM books b1
```

category	NumTitles
Business	3
Consumer	3
Science	2
Travel	1

12. List the categories from which 3 or more books are checked out. Use column headings: Category.

```
SELECT DISTINCT category
FROM books b1
WHERE (

SELECT count( cardholder )
FROM books
WHERE category = b1.category
) >=3
```

			category
☐	✏	✗	Consumer
☐	✏	✗	Science