Comp 1630
Relational Database Design & UML

EXERCISE 9 - SQL

1. Using **subqueries**, write a query to display the first name, last name, address, city, and state from the AUTHORS table, for authors who live in the state of 'CA', and have at least one book type of 'popular_comp' (found in the TITLES table via the TITLEAUTHOR table). Display the name of the author as last name, followed by a comma and a space, followed by the first name. Order the result set by the name of the author. The query should produce the result set listed below.

Name	Address	City	State
Carson, Cheryl	589 Darwin Ln.	Berkeley	CA
Dull, Ann	3410 Blonde St.	Palo Alto	CA
Hunter, Sheryl	3410 Blonde St.	Palo Alto	CA
Locksley, Charlene	18 Broadway Av.	San Francisco	CA

(4 row(s) affected)

2. Rewrite the query in question 1 without using subqueries. The query should produce the same result set as in question 1.

3. Create a view called **vw_sales_title_info** to display the store ID, order date, and quantity from the SALES table, the store name from the STORES table, and the title, price, advance, and publish date from the TITLES table.

4. Run the view **vw_sales_title_info** displaying the store ID, store name, title, and price where the price is equal to **$19.99**. Order the result set by the store ID. The view should produce the result set listed below.

StoreID	StoreName	Title	Price
6380	Eric the Read Books	The Busy Executive's Database Guide	19.99
7131	Doc-U-Mat: Quality Laundry and Books	Prolonged Data Deprivation: Four Case Studies	19.99
7896	Fricative Bookshop	Straight Talk About Computers	19.99
7896	Fricative Bookshop	Silicon Valley Gastronomic Treats	19.99
8042	Bookbeat	The Busy Executive's Database Guide	19.99

(5 row(s) affected)

5. Create a view called **vw_insert_stores** to display the store ID, store name, and state from the STORES table.

6. Using the view **vw_insert_stores**, insert a row into the STORES table with a store ID of **9999**, a store name of '**Peterson Books'**, and a state of '**UT'**. Check your results.

7. List the author ID, last name, city, state, and zip code from the AUTHORS table. The StateName column is generated using the values in the state column (for TN, IN, UT, and CA only). Check for zip codes that are less than 94300. Order the result set by the zip code. The query should produce the result set listed below. (Hint: Use the **CASE** command)

AuthorID	LastName	City	State	StateName	Zip
807-91-6654	Panteley	Rockville	MD	-	20853
527-72-3246	Greene	Nashville	TN	Tennessee	37215
722-51-5454	DeFrance	Gary	IN	Indiana	46403
712-45-1867	del Castillo	Ann Arbor	MI	-	48105
341-22-1782	Smith	Lawrence	KS	-	66044
899-46-2035	Ringer	Salt Lake City	UT	Utah	84152
998-72-3567	Ringer	Salt Lake City	UT	Utah	84152
172-32-1176	White	Menlo Park	CA	California	94025
486-29-1786	Locksley	San Francisco	CA	California	94130

(9 row(s) affected)

8. Using a **subquery**, list the publisher ID and name from the PUBLISHERS table, for those publishers who have published business books. The query should produce the result set listed below. (Hint: Use the **EXISTS** command)

PublisherID	Name
0736	New Moon Books
1389	Algodata Infosystems

(2 row(s) affected)

9. Write the command to determine the index for the EMPLOYEE table.

10. Write the command to create a new composite index called **empinx** on the EMPLOYEE table for the columns **emp_id** and **hire_date**.