CSCD 327 Lab #5 (14 points)
Due: 11:59pm on November 9, 2020

Write the following queries in SQL using *BooksDB*. Please include both the queries and the query results in your submission.

1. List the book tittle and retail price for all books with a retail price lower than the average retail price of all books in the bookstore.

title	retail
HOW TO GET FASTER PIZZA	29.95
THE WOK WAY TO COOK	28.75
REVENGE OF MICKEY	22.00
BODYBUILD IN 10 MINUTES A DAY	30.95
HANDCRANKED COMPUTERS	25.00
SHORTEST POEMS	39.95
COOKING WITH MUSHROOMS	19.95
BIG BEAR AND LITTLE DOVE	8.95
HOW TO MANAGE THE MANAGER	31.95

2. Determine which books cost less than the average cost of books in the same category. List the title, the category, and the cost of these books.

title	category	cost
REVENGE OF MICKEY	FAMILY LIFE	14.20
HANDCRANKED COMPUTERS	COMPUTER	21.80
COOKING WITH MUSHROOMS	COOKING	12.50
BIG BEAR AND LITTLE DOVE	CHILDREN	5.32
DATABASE IMPLEMENTATION	COMPUTER	31.40

3. Determine which orders were shipped to the same state as order 1014 [Note: it is okay to include order 1014 in the result].

order_num
1007
1014

4. Determine which orders had a higher total order amount than order 1002. [Note: total order amount = sum(Quantity * PaidEach)].

order_num	SUM(oi.quantity*oi.paideach)
1001	117.40
1004	170.90
1007	335.85
1012	166.40

5. List the title of all books in the same categories as books previously purchased by customer 1007. Don't include books this customer has already purchased.

title
HANDCRANKED COMPUTERS
PAINLESS CHILD-REARING
BUILDING A CAR WITH TOOTHPICKS

6. List the shipping city and state for the order that had the longest shipping delay. [Note: shipping delay = **datediff**(shipdate, orderdate)].

shipcity	shipstate
TRENTON	NJ

7. Determine which customers placed orders for the least expensive book (in terms of regular retail price) carried by the bookstore.

customer_num
1007
1017
1015

8. Determine the number of different customers who have placed an order for books written or co-written by James Austin.

COUNT(DISTINCT customer_num)
5

Write the following queries in SQL using *ProductsDB*. Please include the queries and the query results in your submission.

9. Determine which products have a list price that's greater than the average list price for all products? Return the product_name and list_price columns for each product. Sort the results by the list_price column in descending sequence.

product_name	list_price
Gibson SG	2517.00
Gibson Les Paul	1199.00

10. Write a SELECT statement that returns the category_name column from the Categories table. Return one row for each category that has never been assigned to any product in the Products table. To do that, use a subquery introduced with the NOT EXISTS operator.

category_name
Keyboards

11. Use a **correlated subquery** to return one row per customer, representing the customer's oldest order (the one with the earliest date). Each row should include these three columns: email_address, order_id, and order_date.

email_address	order_id	order_date
allan.sherwood@yahoo.com	1	2012-03-28 09:40:28
barryz@gmail.com	2	2012-03-28 11:23:20
christineb@solarone.com	4	2012-03-30 15:22:31
david.goldstein@hotmail.com	5	2012-03-31 05:43:11
erinv@gmail.com	6	2012-03-31 18:37:22
frankwilson@sbcglobal.net	7	2012-04-01 23:11:12
gary_hernandez@yahoo.com	8	2012-04-02 11:26:38

Write the following queries in SQL using *UniversityDB*. Please include the queries and the query results in your submission.

12. Find the instructor earning the **highest** salary. (Don't use ORDER BY and LIMIT in your solution.)

id	name	salary
22222	Einstein	95000.00

13. Find the instructor earning the **second-highest** salary. (Don't use ORDER BY and LIMIT in your solution.)

id	name	salary
83821	Brandt	92000.00

14. Find the instructor who has taught the most number of distinct courses. (Don't use ORDER BY and LIMIT in your solution.)

id	name	cnt
10101	Srinivasan	3