**Assignment 2b**

Please work on this assignment INDIVIDUALLY or WITH YOUR PARTNER if you are working in a pair. Your submitted work must be "ENTIRELY" your work. If your submitted work is found out to have been re-produced from any human or non-human resource (for example, the web), I will strictly follow the Academic Misconduct procedure as per college policies.

**How to submit:** You need to submit this assignment **twice**. Please submit the .sql file ONLY using the link ‘Assignment 2b’ in ‘Assignment 2 folder’ in Assignments. Name this file Assignment2B\_*yourname*(s).sql**. This is the one I will be marking so do not forget to submit via this link.** Now just change the extension of the file form .sql to .txt and submit it with the ‘Assignment 2b Turnitin’ option in the same folder. **I will be marking the Assignment 2b only if it has been submitted in BOTH the above locations. Please submit ONLY ONCE in the ‘**Assignment 2b Turnitin’ once you are sure you do not want to make any changes to your assignment submission. The submission will be compared to assignments of all other students and web sources for Academic Integrity implementation purposes.

**If the script does not run successfully due to any error(s), I will only mark the results I can see. So please ensure that there are no errors in the script and it runs completely and successfully. I will be running the complete script at one go, not each solution independently.**

**The output AND the code needs to be completely correct, following the question requirements, to be awarded marks for a solution. Please label your answer to match the question number else will not be marked. Just copy the question into the script and comment it – quick and easy.**

I have provided a sql script premiere\_products.sql. Download it, run it and you have the required database and tables to work with for this assignment. Put your name(s) and date in comments at the head. I will only mark your script if I can identify it by your name(s) in comments.

ADD the solutions of the queries below in this file. The queries should be answered in sequence and numbered correctly in comments.

**Please write non-redundant code. You will lose marks for redundant/useless code, such as joining tables not required, grouping by where not required, displaying columns not asked for, using DISTINCT if not required etc. Please provide grammatically correct and meaningful relevant alias names for required fields in all select statements. Non-numeric literals must appear in quotes and numeric data without quotes and date with proper CAST.**

**All requirements carry marks and failure in following them will result in losing marks.**

You need to follow the instructions below COMPLETELY and CORRECTLY. PLEASE ANSWER ALL QUERIES FROM THE INFORMATION GIVEN IN THE QUESTION - NO PEEKING IN THE TABLE DATA AND USING EXTRA INFORMATION FROM THERE.

1. Use ONLY an OUTER JOIN to display the order number and order date for those orders that have exactly two different parts on the order. Sort the result in descending order number. For example, order number 21617 has two parts BV06 and CD52 on order.
2. Use ONLY an OUTER JOIN to list the order number, part number, part description, price, quoted price and the price discount(= price – quoted\_price) for those parts that are on order and the price discount > 0. Order the result by part number. For example, part number AT94 with order number 21608 was given a discount of 3 dollars.
3. Use ONLY an INNER JOIN to list the rep number and full address (concatenate the street, city, state and zip **each separated by a comma**) of the rep, with the name of the customer he/she represents, if the rep number's street contains Jackson. YOU NEED TO USE STRING FUNCTION(S) to check if the street contains Jackson and LIKE is not a function fyi. Sort the output by rep number. For example, rep 25 lives on street 124 Jackson.
4. Use ONLY an INNER JOIN to display the customer number and his/her order total, only for all his/her orders, if the order total >= 1000. Sort the output by customer number. The total of any order is (quantity\_ordered\* quoted\_price). For example, the order total for customer 408, who has two orders, is 4\*329.95 + 1\*399.99 = 1719.79
5. Use ONLY a JOIN to display the rep number and full name (concatenate the first and last name, **separated by a blank**) of the rep(s) who does not represent any customer currently. For example, rep number 99 (Anju Chawla) does not represent any customer.
6. Using a CORRELATED SUBQUERY ONLY (with exists/not exists), find the rep number and rep name (first name and last name concatenated) of each rep who represents a customer living in the zip 33321. Order the result by rep number. For example, rep number 25, Greg Smith has a customer living in that zip.
7. Using a CORRELATED SUBQUERY ONLY (with exists/not exists), find the customer number and name of each customer whose rep does not live in city Grove. Order the result by customer number. For example, the rep of customer number 282 (Brookings Direct) does not live in Grove city.
8. Use ONLY a subquery to find the number and name of each customer that currently has an order on file for a Washer (description of the part). No other information is available so code the solution only based on information given. For example, customer number 356 (Ferguson's) has an order for Washer.

1. Use ONLY a subquery and the ANY(not ALL) keyword (NO INBUILT FUNCTION ALLOWED) to list the part number, part description, unit price and class for each part that has a unit price greater than the unit price of every part in the Appliances class, AP. Order the results by part number. For example, price of part number BV06 (Home Gym) is greater than the price of all parts in the AP class.