Instructions: The homework is due on Fri, Apr 27th, 11:59pm. Submission is done in electronic format: create a folder HW5 in your course directory and place the source code there. Name your source files Q1.sql, Q2.sql and Q3.sql. Make sure you set the proper permissions for files!

Consider a database schema with three relations:

- Parts (pid:integer, pname:string, year:integer, price:integer)
- Suppliers (sid:integer, sname: string, state:string, zipcode:string)
- Orders (pid:integer, sid:integer, quantity:integer)

The description is as follows: a factory keeps a database with parts that it uses, the suppliers of those parts, and purchase orders. Each part is uniquely identified by pid. Each part has a string description pname, year of fabrication and price per unit. Parts are provided by suppliers, and each supplier is uniquely identified by sid. Each supplier has a name sname, and it operates at the given state and zipcode. The company has a number of orders for parts, and each order contains the pid of the part ordered, the sid of the supplier from which the part is ordered, and the quantity ordered. You are allowed flexibility on the exact attribute types you use for your schema, as long as they reasonably match the specification above (e.g., in terms of number types, string types). Include the schema definition in your submission in a file called schema.sql.

Question 1 (25 points for 430, 15 points for 630)

Write a PL/SQL function that takes a price as argument and returns the pid of the part that has the third-nearest price to the one given. Note that nearest can be either higher or lower. In case there are multiple parts with that property, choose the one with higher year. If there is still a tie, choose the one with highest pid.

Question 2 (25 points for 430, 15 points for 630)

Write a PL/SQL procedure that receives as arguments pid, sid and quantity of a prospective order. First, you need to determine if the value (i.e., dollar amount) of that order will be lower or equal than 75% of the average previous order value for that part. If the answer is yes, go ahead and input the new order into the database.

Otherwise, compute the price value that would make the prospective order value be exactly at the 75% limit above, and then insert a NEW part with that price, and the same attributes as the part given in the pid parameter (except for the pid of course, for which you need to determine a unique value). Then, input in the database an order with the sid and quantity given, but for the new pid.

Question 3 [630 only] (0 points for 430, 20 points for 630)

Write a PL/SQL module that takes as input a zipcode and returns the mean and the variance of order values (i.e., dollar amount) for all orders corresponding to suppliers in that zipcode. You are not allowed to use any pre-defined Oracle functions for computing statistics.