CSC2001F - SQL ASSIGNMENT Part 1 of 2 parts

Instructions:

In this assignment you will create and load your own MySQL database using the given **.sql** file, write SQL queries to answer the questions below, and run them against your database. A diagram, showing the schema of the Classic Cars database that this .sql file creates, is provided. **MSRP** is Maufacturers Recommended Selling Price, **buyPrice** is the price the company has to pay to buy the item from the manufacturer, and **checkNumber** is the cheque number of the payment. **Your queries must be correct for *any* instance of the database schema, and not just for the given sample data.**

Please use the Amathuba Discussions forum for all questions on this assignment, so all information is in 1 place accessible to all. Emails will not be answered. **Check Amathuba Discussions** frequently for messages on this assignment.

Submission:

Submit a zip file (with your student number as its name, e.g. STDNUM01.zip) containing separate files for each query. Call the file with your answer for query 1 '`query1.sql`', the one with your answer for query 2 a file '`query2.sql`', and so forth. The questions will be Automarked, so carefully **check your spelling**, and make sure that you use **lowercase and uppercase letters exactly as in the assignment question**, as the output of each of your files will be compared with the expected output for **exact match**.

NOTE:

- Your ZIP file must only contain your answer files. It **should not contain a folder** containing your answer files.
- Do NOT include comments in your answer files and **avoid pasting from PDFs or Word documents;** certain characters result in errors on the Automarker.
- You do not need to complete all questions before trying out your answers on the automatic marker – it will just report that it can't find some.
- The automarker is used by all CS students and at times can be under heavy load. If there is not an immediate response when submitting your work, please be patient.

Queries:

1. Show all information in the **offices** relation.
2. Show the **city** and **phone** of all offices, in alphabetical order of **city** – if there is more than one office in a city then give their phone numbers in decreasing order**.**
3. Show all information in the **offices** table for **offices** located in countries 'UK' and 'SA'.
4. Show all information in the **orderdetails** relation for large orderlines. A large orderline is one where the cost is more than 10 000 (cost is **quantityOrdered** times **priceEach**).
5. We have doubled our **quantityInStock** of every product from **productVendor** 'Exoto Designs'. Show **productVendor, productCode,** and new (doubled) **quantityInStock** for all 'Exoto Designs' products - I'm unsure how "Exoto" is spelt, but I know it starts "Ex" and has a "to" somewhere. Call the last column newStock.
6. In what cities do we have **offices**? Call the answer column **city**.
7. Show all data for **offices** where there is an **addressLine2** value but the **state** is missing (NULL).
8. How many tuples (rows) are there in **employees**? Call the result column numEmps.
9. What is the average **buyPrice** in the database? Call the result avPrice and show 2 decimal places.
10. How many different **creditLimit** values are there in our database? Call the result numLimits.
11. Show **orderNumber, status, priceEach, quantityOrdered** and **productName** for all **products** from **productVendor** 'Exoto Designs'.
12. Show the **OrderNumber, Comments** and **customerName** of all orders that have a **status** of 'Disputed'.
13. Show the **productCode** of all **products** that have never been ordered.
14. Show how many **employees** there are in each office. Call the 1st column **officeCode** and the 2nd column numEmps
15. Show how many Sales Reps there are in each office. Call the 1st column **officeCode** and the 2nd column numReps
16. Show how many customers each employee is associated with (as **salesRepEmployeeNumber**), but only for employees who are the **salesRepEmployeeNumber** for at least 10 customers and who report to employee 1143. Call the 1st column **salesRepEmployeeNumber** and the 2nd column numCustomers.
17. Which pairs of offices are in the same country? Show the country and the 2 **officeCodes**, making sure that the 2nd column's **officeCode** is less than the 3rd column's **officeCode** so information is not repeated. Call the 2nd column oneOffice and the 3rd column otherOffice.
18. What percentage of the **offices** are located in the 'USA' **country**? Call the result percentUSA.
19. Show the **customerNumber** of the customer/s with the largest single check (cheque) payment **amount.**
20. Give a SQL statement to output "YES" if any attribute storing a price contains a value that is zero – i.e. if **priceEach** in any **orderdetails** row**,** or **buyPrice** in any **products** row, is zero. If there is no zero in those columns then it should output an empty table. Call the result column anyProblems.