

CSC2001F - Assignment 4 - Part 1 of 2

Assignment 4 Part 1: SQL to submit to the Automarker

1. Instructions

In this assignment you will:

1. Create and load your own MySQL database using the SQL file 'classicModels2022.sql', available from the Vula assignment page and from the resources section of the site.
2. Write SQL queries to answer the questions below and run them against your database. **Your queries must be correct for any instance of the database schema, and not just for the given sample data.**

A diagram of the database schema may be found at the end of this document.

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

2. Marking

There are 20 questions. Each correct question earns 5 marks, for a total of 100.

3. Submission

Submit one zip file containing separate files for each query to the automatic marker.

Call the file with your answer to question 1 'query1.sql', the one with your answer to question 2 'query2.sql', and so forth.

To ensure successful automatic marking, make sure that you use lowercase and uppercase letters exactly as in the assignment question - the output of each of your files will be compared with the expected output.

NOTE:

- Your ZIP file must only contain your answer files. It should not contain a folder containing your answer files.
- You can include single line comments in your answer files. A comment must begin with '--' and may not contain any semicolons, ";", hyphens. "-", or single quotes. "'". Avoid pasting from PDFs or Word documents.
- 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 automatic marker 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.

4. Questions

Queries

1. Show all information in the **offices** relation.
2. Show any 1 tuple in the **products** relation (just one).
3. Give the **phone** numbers of **customers** whose **salesRepEmployeeNumber** is missing.

4. Give the average payment **amount**, and the total of all **payments** in the database. Call the first value **mean** and the second value **total**.
5. Get the **city** and **phone** number of the Paris and London **offices**.
6. What cities are our **offices** in?
7. If we were to make a call to every **phone** number in the **customers** relation, how many calls would we have to make? Call the answer **numCalls**.
8. Give the VAT amount of the costliest product i.e., the product with highest **buyPrice**. As VAT is 15%, if a **buyPrice** is say 100, then the VAT is 15. Call the answer **biggest**.
9. For each customer, give their **customerNumber** and the total of all **payments** by that customer. Call the second column **total**.
10. For each customer, give their **customerName** and the total of all **payments** by that customer. Call the second column **total**.
11. For each customer, give their **customerName** and the total of all **payments** by that customer, but only for Paris customers who have made more than 4 payments. Call the second column **total**.
12. Which customers have never made any **payments**? Give their **customerName** and **phone** number.
13. Give the **productName**, **quantityInStock** and **productLine** description for all products that have a quantity in stock below 100.
14. What product(s) have been ordered the most? Give the **productCode(s)** and the total quantity ordered. Call the second column **total**.
15. Give the emails of **employees** who are sales rep for fewer **customers** than employee 1166. Only show **employees** whose job title is 'Sales Rep' in your answer.
16. For customer 121, give the total cost (price * quantity) of all their orders. Call the answer **totalCost**.
17. Give the total **payments** obtained from the **customers** of each sales rep in office 7. Call the answer columns **firstName**, **lastName** and **total** (in that order i.e., **lastName** is middle column).

Database manipulation

18. Add a new **office** to the database, giving it **officeCode** 999 (meaning planned for later). This office will be in Cape Town, but no other details are known yet. Make **state** 'Western Province'.
19. Employee 1625 is superstitious. Change their employee number in the database, giving them the employee number 1 greater than the largest employee number in the database.
20. The order with **OrderNumber** 10101 was never signed by the customer. Remove it from the database.

5. Appendices

