National University of Computer & Emerging Sciences

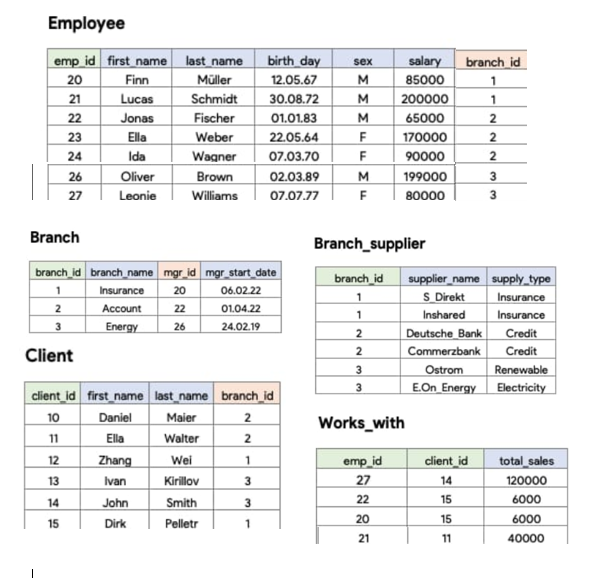
Department of Computer Science

Database Systems

Assignment # 02

|  |
| --- |
| Instructions:   1. **Plagiarism is strictly prohibited; assignment will be marked zero if plagiarized.** 2. **Late submission is not allowed.** 3. Assignment should be submitted online with the naming scheme as “22F-XXXX\_DB\_A#” 4. Total Marks 130 |
| * **Output Requirement: Each screenshot must display the output of the executed SQL query.** * **Modified Tables: In cases where the output of a query is not directly visible (such as in update or delete queries), explicitly include a modified table using a SELECT query. This will show the current state of the table after the operation.** * **Documentation Format: All SQL queries must be typed directly into a Word document. Do not include screenshots of the queries themselves; only the results should be captured as screenshots.** * **Clear Separation: Clearly separate each section of the document by using headings for different queries or tasks. Include a brief description of what each query does.** * **Formatting: Ensure that the SQL code is properly formatted in the Word document for readability. Use a monospaced font (like Courier New) to distinguish code from regular text.** |

**NOT FOLLOWING THE INSTRUCTIONS WILL RESULT IN MARKS DEDUCTION**



**PART A:** You are tasked with creating SQL queries based on the provided relational schema (Employee, Branch, Client, Branch\_supplier, and Works\_with) with specific constraints. Here’s a step-by-step breakdown of what you need to do:

1. **Create the 5 tables** with appropriate data types. **(5\*4=20 Marks)**
   * Ensure no **NULL values**.
   * Define **Primary keys** and **foreign keys**.
   * Set **constraints** such as salary >= 65,000.
2. **Insert data** exactly as seen in the provided image into each table.

**PART B: WRITE SQL QUERIES FOR THE FOLLOWING but DON’T USE JOINS. (46 Marks)**

1. Write an SQL query to insert a new employee working in branch 2 manages by employee. **(2)**
2. Formulate an SQL query to insert a new client associated with a branch. **(2)**
3. Write an SQL query to update the total\_sales of employee with total\_sales more than 10,000. **(2)**
4. List all suppliers who provide supplies on credit, along with the branches they supply to, sorted by supplier name. **(2)**
5. List all employee where the employee’s birth date is between January 1, 2020, and December 31, 2022. **(2)**
6. Select employees whose first name starts with the letter 'L' and whose salary is between 70,000 and 150,000. **(2)**
7. Find the total number of clients served by each branch, grouped by branch\_id. **(2)**
8. List all employees who do not work in branches that supply materials on credit. **(4)**
9. Select all employees whose salary is either below 75,000 or above 200,000, but they must not work in the 'Research' branch. **(4)**
10. Find the total sales amount for each employee, but only for employees who have more than three clients assigned to them. **(4)**
11. Select all clients who are associated with branches other than the 'Finance' and 'Insurance' branches, sorted by their last name. **(4)**
12. Retrieve the name of the employee who do not work with any client, sorted ascending. **(4)**
13. List Employees and Their Corresponding Manager's Name. **(4)**
14. Delete employees from the Employee table where their salary is less than 70,000, but only if they are not in the branch 'Marketing'. **(4)**
15. Update the salaries of employees who work in the branch 'Sales' to increase by 15%, but only if their current salary is less than 120,000. **(4)**

**PART C: WRITE THE FOLLOWING QUERIES USING JOINS**. **(32 Marks)**

1. Retrieve the first name and last name of all employees along with their corresponding branch names. **(2)**
2. Retrieve the names of all branches and their corresponding managers, ensuring that branches are displayed even if they lack assigned managers. **(2)**
3. Display all branches and the employees assigned to them, ensuring that all branches are included in the results. **(2)**
4. Retrieve all employees along with their salary and the name of the branch they work in, even if some employees do not belong to any branch. **(2)**
5. List all clients and the employees who work with them, including the total sales for each client. **(4)**
6. Find all suppliers and the branches they supply to, including branches that may not have any associated suppliers. **(4)**
7. Show all employees and their clients, including cases where an employee may not have any clients assigned. **(4)**
8. List all employees and their respective managers, providing details on both the employees and their managers. **(4)**
9. Display the names of clients and the employees they work with, ensuring that clients are listed even if they have no employees assigned. **(4)**
10. Show a list of all employees alongside any clients they may work with, ensuring that all employees are represented in the output. **(4)**

**PART D: SOLVE THE FOLLOWING USING SUB QUERIES. (32 Marks)**

1. **Find all employees who have a higher total sale than the average total sales of their respective branches. (2)**
2. Retrieve the names of all clients who have total sales exceeding the average total sales across all clients. **(2)**
3. Write a query to find all employees whose salaries are greater than the average salary of employees in the same branch. **(4)**
4. Retrieve clients who have purchased from any branch that has a total sales amount greater than 100,000. **(4)**
5. Find the branches where the total sales from clients associated with that branch exceed the total sales from clients of any other branch. **(5)**
6. Write a query to display all employees for whom there exists a client with total sales greater than 50,000. **(5)**
7. List employees whose birth date is earlier than the earliest birth date of employees in the 'Finance' branch. **(5)**
8. Show suppliers who provide supplies to branches where the number of employees is greater than 10. **(5)**