Database Systems Lab

BS(SE) Morning - Fall 2022

LAB – 09

Course & Lab Instructor: Dr. Sanam Ahmed

Task 01 (14 Marks)

1. Display the manager name, manager number and salary of the lowest paid employee for the manager. Exclude anyone whose manager is not known. Sort the output in descending order of the salary.
2. Retrieve a list of employees who have the same salary grade as their managers
3. List employee names (ENAMES) and the count of employees working under each manager.
4. List EMPNO, ENAME, JOB, SAL, DEPTNO, DNAME, RANK, TAX, NET_SAL. Tax is a derived attribute computed on the basis of GRADE. All the employees rank 1 are exempted from tax. For the employees with GRADE 2,3, the tax rate is 10% of the salary. For the employees with GRADE above 3, the tax rate is 12% of the salary. NET_SAL is computed as SAL-TAX.
5. List all employees along with their managers, managers' salary and their managers' salary grades.
6. List all employees, their managers, managers department, no of employees in managers department and average of the grades of all employees in that department.
7. From PUCIT table, list all students who has the matching first name to the middle name of any other student from the table. Display the fullnames of both matching students. (Insert student name 'Hassan Muhammad Malik' into the table first)
8. You need to combine the names of students and employees such that we get first and last name from Pucit table and middle name will be the employee name. For Example: for first row: Muhammad KING Khan.
9. List all employees who earn Grade 2 only by the commission they earn.
10. List employees who earn 2000 more than the average salary in their department.

Tables Structure:

Student Table

Attributes: Contains details about students such as student_id, first_name, last_name, department_id, and class_id.

References:

department_id references the **Department** table, indicating which department the student belongs to.

class_id references the **Class** table, showing which class the student is enrolled in.

Teacher Table

Attributes: Contains details about teachers such as teacher_id, first_name, last_name, and department_id.

References:

department_id references the **Department** table, indicating which department the teacher belongs to.

Course Table

Attributes: Contains details about courses such as course_id, course_name, teacher_id, and department_id.

References:

teacher_id references the **Teacher** table, indicating who teaches the course.

department_id references the **Department** table, indicating which department offers the course.

Department Table

Attributes: Contains department details such as department_id and department_name.

No references as this is the parent table for both **Student**, **Teacher**, and **Course**.

Class Table

Attributes: Contains details about classes such as class_id, class_name, course_id, and teacher_id.

References:

course_id references the **Course** table, indicating which course the class covers.

teacher_id references the **Teacher** table, showing which teacher conducts the class.

Tasks:

11. Join the Student and Department tables to show each student's name and department, along with the total number of students in that department.
12. Join the Course and Department tables to list courses that a specific department offers, along with the number of classes and total enrollment in each course.
13. Join the Teacher and Course tables to display each teacher's name along with the courses they teach, and rank the courses by the number of students enrolled in each course.
14. Join the Class, Course, and Teacher tables to show which course a class is related to and which teacher is assigned to that class, and include the average grade for students in that class.
15. Use a GROUP BY on the Student table's class_id to count how many students are enrolled in each class, and filter for classes with more than 10 students.
16. Join the Student and Department tables and group by department_id to calculate the average number of students in each department, including only departments with more than 3 courses.
17. Join the Department and Course tables and group by department_id to find how many courses each department offers, along with the total number of students enrolled in those courses.
18. Use a GROUP BY on the Course table's teacher_id and apply a HAVING clause to find teachers assigned to more than one course, then rank these teachers by the total number of students across all courses they teach.

19. Join the Student, Class, and Teacher tables to list each student's name along with the class they're in, the teacher responsible for their class, and the total number of students the teacher is responsible for.
20. Join the Teacher and Class tables and use GROUP BY on the teacher's ID to calculate the total number of classes each teacher is conducting, and also include the average student enrollment in those classes.

Task 03 VIVA (5 Marks)