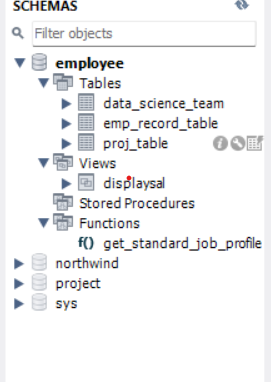
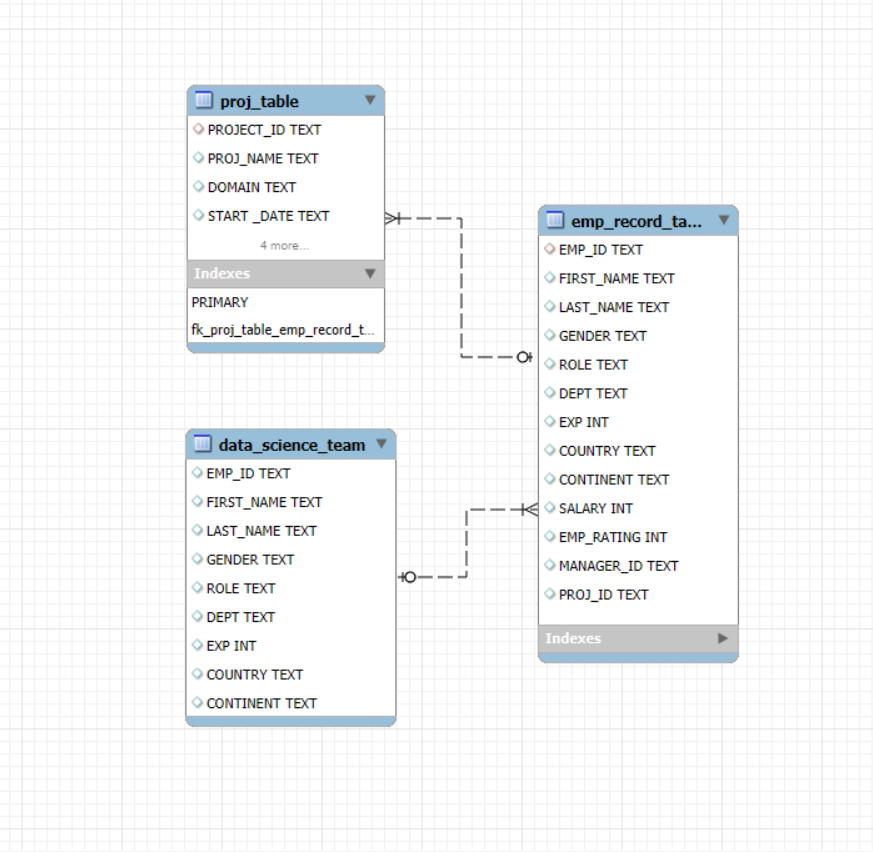
-- Q-1 – Create a database named employee, then import data\_science\_team.csv proj\_table.csv and emp\_record\_table.csv into the employee database from the given resources.



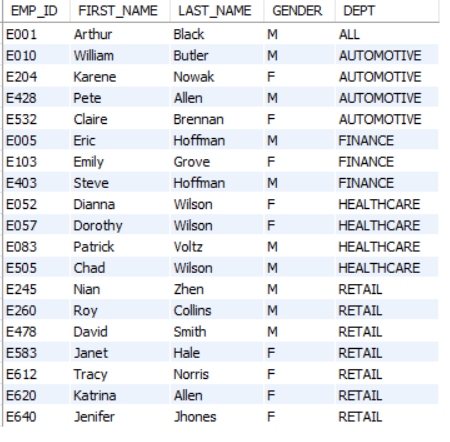
-- Q-2 – Create an ER diagram for the given employee database.



-- Q-3 – Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, and DEPARTMENT from the employee record table, and make a list of employees and details of their department.

select emp\_record\_table.EMP\_ID,emp\_record\_table.FIRST\_NAME,emp\_record\_table.LAST\_NAME,

emp\_record\_table.GENDER,emp\_record\_table.DEPT from emp\_record\_table order by emp\_record\_table.DEPT;



-- Q-4 -- Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPARTMENT, and EMP\_RATING if the EMP\_RATING is: ● less than two ● greater than four ● between two and four

select emp\_record\_table.EMP\_ID,emp\_record\_table.FIRST\_NAME,emp\_record\_table.LAST\_NAME,

emp\_record\_table.GENDER,emp\_record\_table.DEPT,emp\_record\_table.EMP\_RATING,

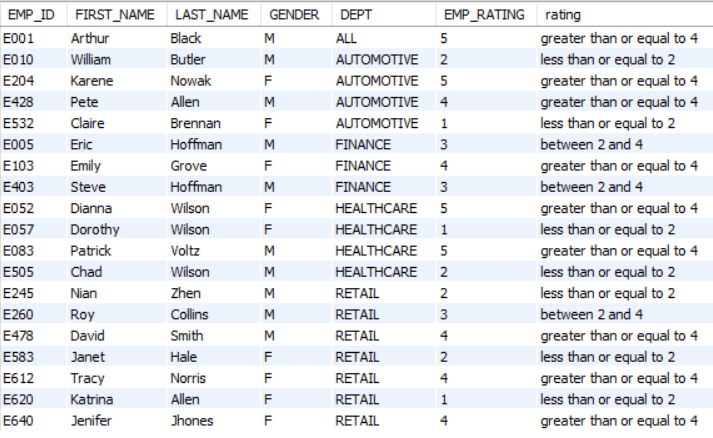
case

when EMP\_RATING <= 2 then 'less than or equal to 2'

when EMP\_RATING > 2 and EMP\_RATING < 4 then 'between 2 and 4'

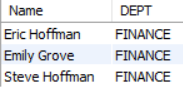
else 'greater than or equal to 4'

end as rating from emp\_record\_table order by DEPT ;



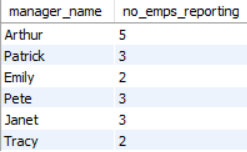
-- Q-5 -- Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPARTMENT, and EMP\_RATING if the EMP\_RATING is: ● less than two ● greater than four ● between two and four

select concat(first\_name,' ',last\_name) as Name, DEPT from emp\_record\_table where DEPT='finance';



-- Q-6 -- Write a query to list only those employees who have someone reporting to them. Also, show the number of reporters (including the President).

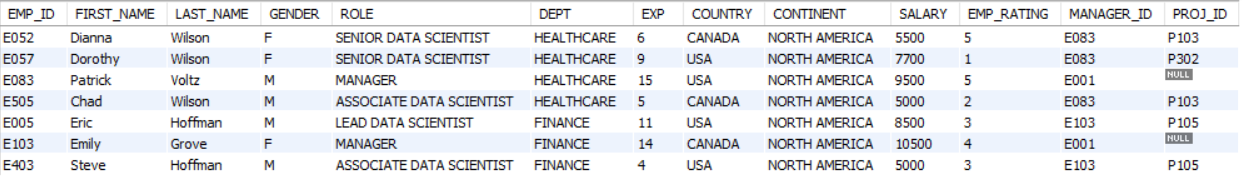
select m.first\_name as manager\_name,count(\*) no\_emps\_reporting from emp\_record\_table as e join emp\_record\_table as m on m.EMP\_ID=e.MANAGER\_ID group by manager\_name having no\_emps\_reporting>0;



-- Q-7 -- Write a query to list down all the employees from the healthcare and finance departments using union. Take data from the employee record table.

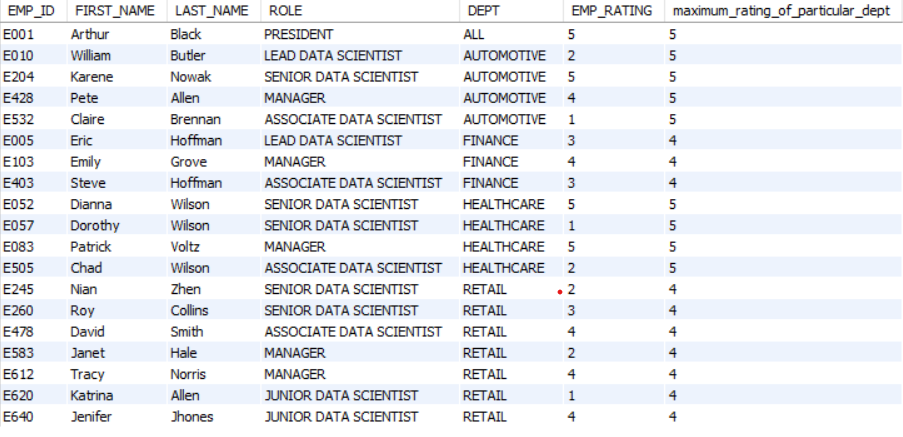
select \* from emp\_record\_table where dept="healthcare" union all

select \* from emp\_record\_table where DEPT='finance';



-- Q-8 -- Write a query to list down employee details such as EMP\_ID, FIRST\_NAME, LAST\_NAME, ROLE, DEPARTMENT, and EMP\_RATING grouped by dept. Also include the respective employee rating along with the max emp rating for the department.

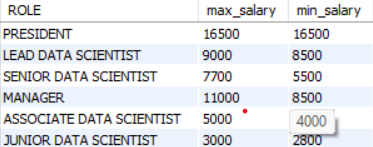
select EMP\_ID,FIRST\_NAME,LAST\_NAME,ROLE,DEPT,EMP\_RATING, max(EMP\_RATING) over(partition by dept) as maximum\_rating\_of\_particular\_dept from emp\_record\_table;



-- Q-9 -- Write a query to calculate the minimum and the maximum salary of the employees in each role. Take data from the employee record table.

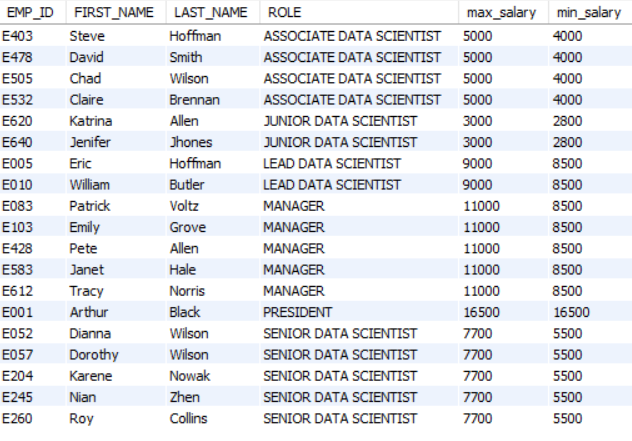
* With group by clause

select ROLE,max(SALARY) as max\_salary,min(SALARY) as min\_salary from emp\_record\_table group by ROLE;



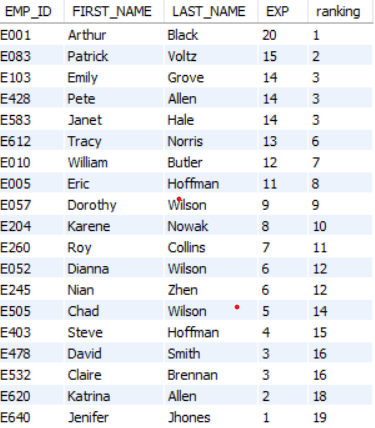
* Without group by clause

select EMP\_ID,FIRST\_NAME,LAST\_NAME,ROLE,max(SALARY) over(partition by role) as max\_salary,min(SALARY) over(partition by role) as min\_salary from emp\_record\_table;



-- Q-10 -- Write a query to assign ranks to each employee based on their experience. Take data from the employee record table.

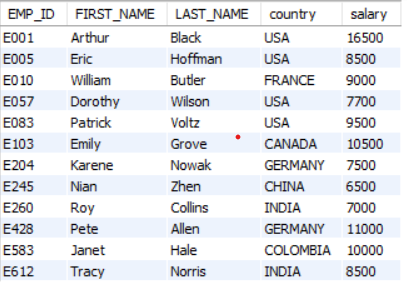
select EMP\_ID,FIRST\_NAME,LAST\_NAME,EXP,rank() over(order by exp desc) as ranking from emp\_record\_table;



-- Q-11 -- Write a query to create a view that displays employees in various countries whose salary is more than six thousand. Take data from the employee record table

create or replace view displaysal as select EMP\_ID,FIRST\_NAME,LAST\_NAME,country,salary from emp\_record\_table where salary>6000;

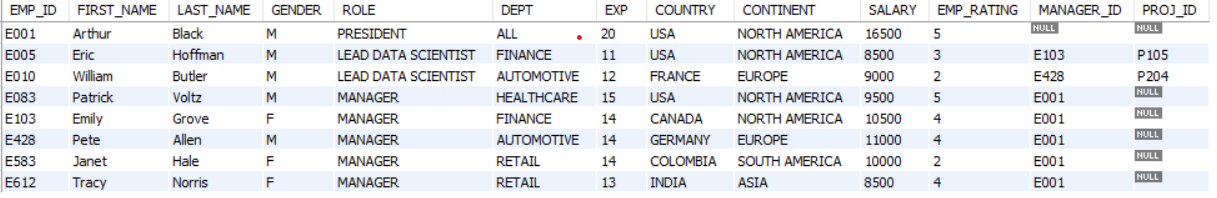
select \* from displaysal;



-- Q-12 -- Write a nested query to find employees with experience of more than ten years. Take data from the employee record table.

select \* from emp\_record\_table where EMP\_ID in

(select EMP\_ID from emp\_record\_table where EXP>10);



-- Q-13 -- Write a query to create a stored procedure to retrieve the details of the employees whose experience is more than three years. Take data from the employee record table.

USE `employee`;

DROP procedure IF EXISTS `empexpabove3`;

USE `employee`;

DROP procedure IF EXISTS `employee`.`empexpabove3`;

;

DELIMITER $$

USE `employee`$$

CREATE PROCEDURE `empexpabove3`()

BEGIN

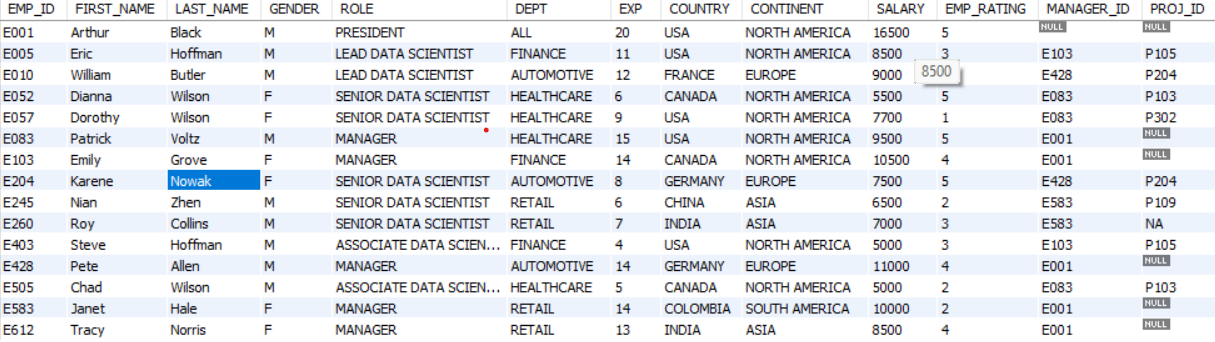
select \* from emp\_record\_table where exp>3;

END$$

DELIMITER ;

;

call empexpabove3;

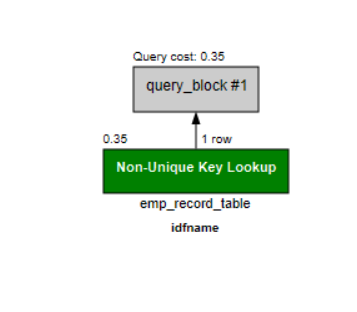


-- Q-14 – Create an index to improve the cost and performance of the query to find the employee whose FIRST\_NAME is ‘Eric’ in the employee table after checking the execution plan.

create index idfname on emp\_record\_table(first\_NAME(50));

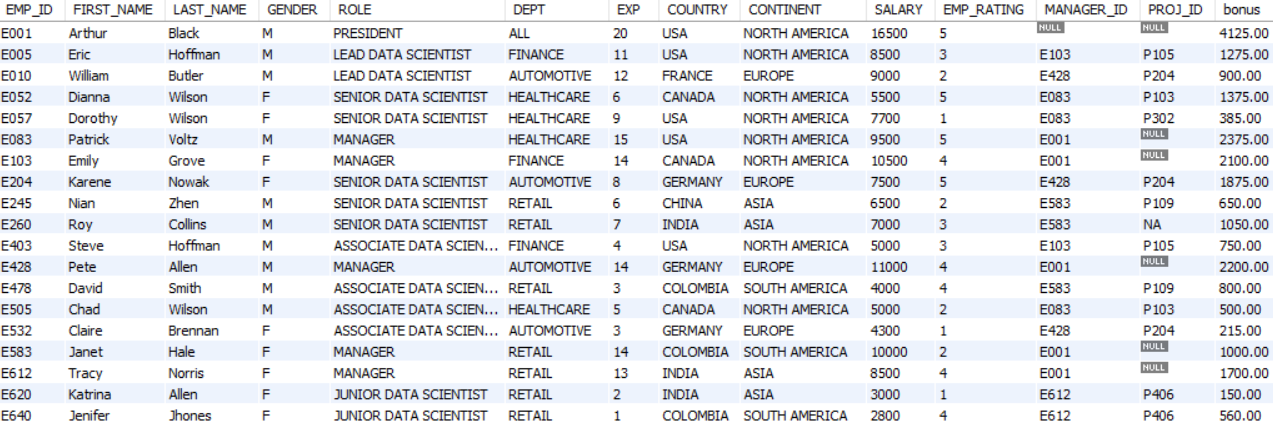
select \* from emp\_record\_table where FIRST\_NAME='eric';





-- Q-15 -- .Write a query to calculate the bonus for all the employees, based on their ratings and salaries (Use the formula: 5% of salary \* employee rating).

select \*, SALARY \* 0.05\* EMP\_RATING as bonus from emp\_record\_table;



-- Q-16 -- Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table.

select CONTINENT,COUNTRY,avg(SALARY) as average\_salary from emp\_record\_table group by continent,country with rollup;

