**Objective:**

To facilitate a better understanding, managers have provided ratings for each employee which will help the HR department to finalize the employee performance mapping. As a DBA, you should find the maximum salary of the employees and ensure that all jobs are meeting the organization's profile standard. You also need to calculate bonuses to find extra cost for expenses. This will raise the overall performance of the organization by ensuring that all required employees receive training.

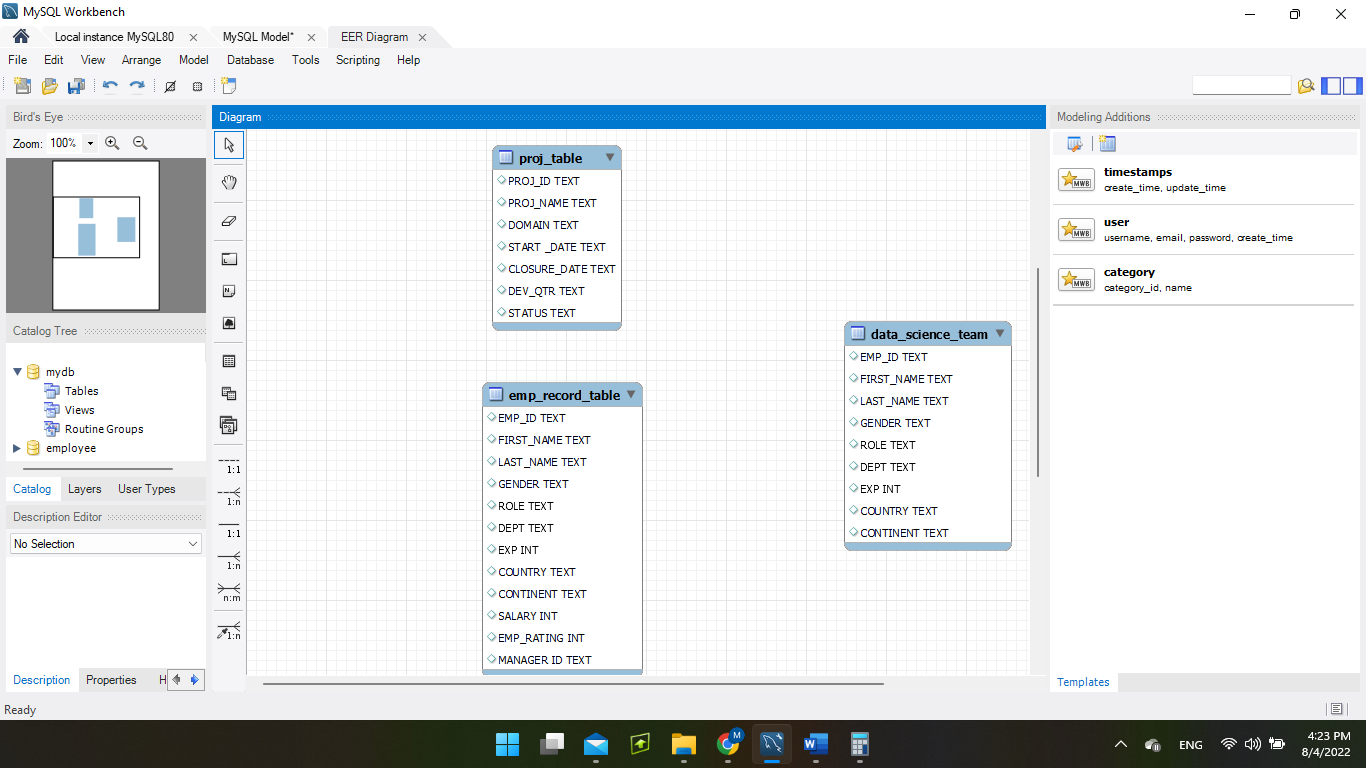
**The task to be performed:**

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. (LESSON 2)

Graphical user interface, text, application

Description automatically generated

1. Create an ER diagram for the given **employee**database. (LESSON 3)



1. 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. (LESSON 5-6)

  SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT FROM emp\_record\_table;

A screenshot of a computer

Description automatically generated

1. Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPARTMENT, and EMP\_RATING if the EMP\_RATING is:  (LESSON 5-6)

* less than two
* greater than four
* between two and four

 SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING FROM emp\_record\_table WHERE EMP\_RATING<2;

A screenshot of a computer

Description automatically generated

 SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING FROM emp\_record\_table WHERE EMP\_RATING>4;

A screenshot of a computer

Description automatically generated

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPT, EMP\_RATING FROM emp\_record\_table WHERE EMP\_RATING BETWEEN 2 AND 4;

Graphical user interface, text, application

Description automatically generated

1. Write a query to concatenate the FIRST\_NAME and the LAST\_NAME of employees in the Finance department from the employee table and then give the resultant column alias as NAME. (LESSON 5-6)

SELECT FIRST\_NAME, LAST\_NAME,CONCAT(FIRST\_NAME, LAST\_NAME) AS full\_name FROM emp\_record\_table WHERE DEPT=”FINANCE”;

Graphical user interface, text, application

Description automatically generated

1. Write a query to list only those employees who have someone reporting to them. Also, show the number of reporters (including the President). (LESSON 5-6)

SELECT \* FROM emp\_record\_table WHERE ROLE = "MANAGER" OR ROLE = "PRESIDENT";Graphical user interface, application

Description automatically generated with medium confidence

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. (LESSON 5-6)

SELECT \* FROM emp\_record\_table WHERE DEPT = "HEALTHCARE"

UNION

SELECT \* FROM emp\_record\_table WHERE DEPT = "FINANCE";

Graphical user interface, application

Description automatically generated

1. 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. (LESSON 5-6)

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, ROLE, DEPT, EMP\_RATING, max(emp\_rating) as max\_dept\_rating FROM emp\_record\_table GROUP BY DEPT;

A screenshot of a computer

Description automatically generated

1. Write a query to calculate the minimum and the maximum salary of the employees in each role. Take data from the employee record table. (LESSON 5-6)

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, MIN(SALARY), ROLE FROM emp\_record\_table GROUP BY ROLE

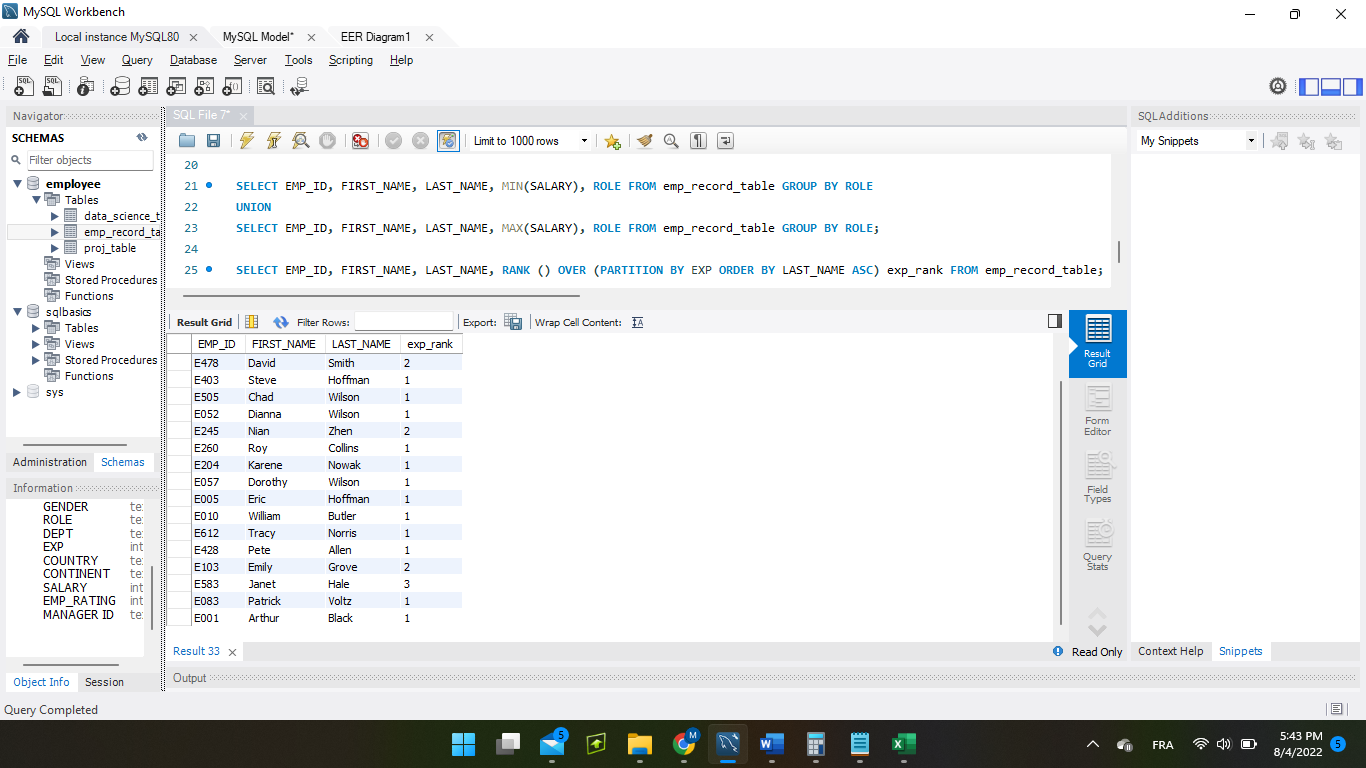
UNION

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, MAX(SALARY), ROLE FROM emp\_record\_table GROUP BY ROLE;Graphical user interface, application

Description automatically generated

1. Write a query to assign ranks to each employee based on their experience. Take data from the employee record table. (LESSON 5-6)

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, RANK () OVER (PARTITION BY EXP ORDER BY LAST\_NAME ASC) exp\_rank FROM emp\_record\_table;

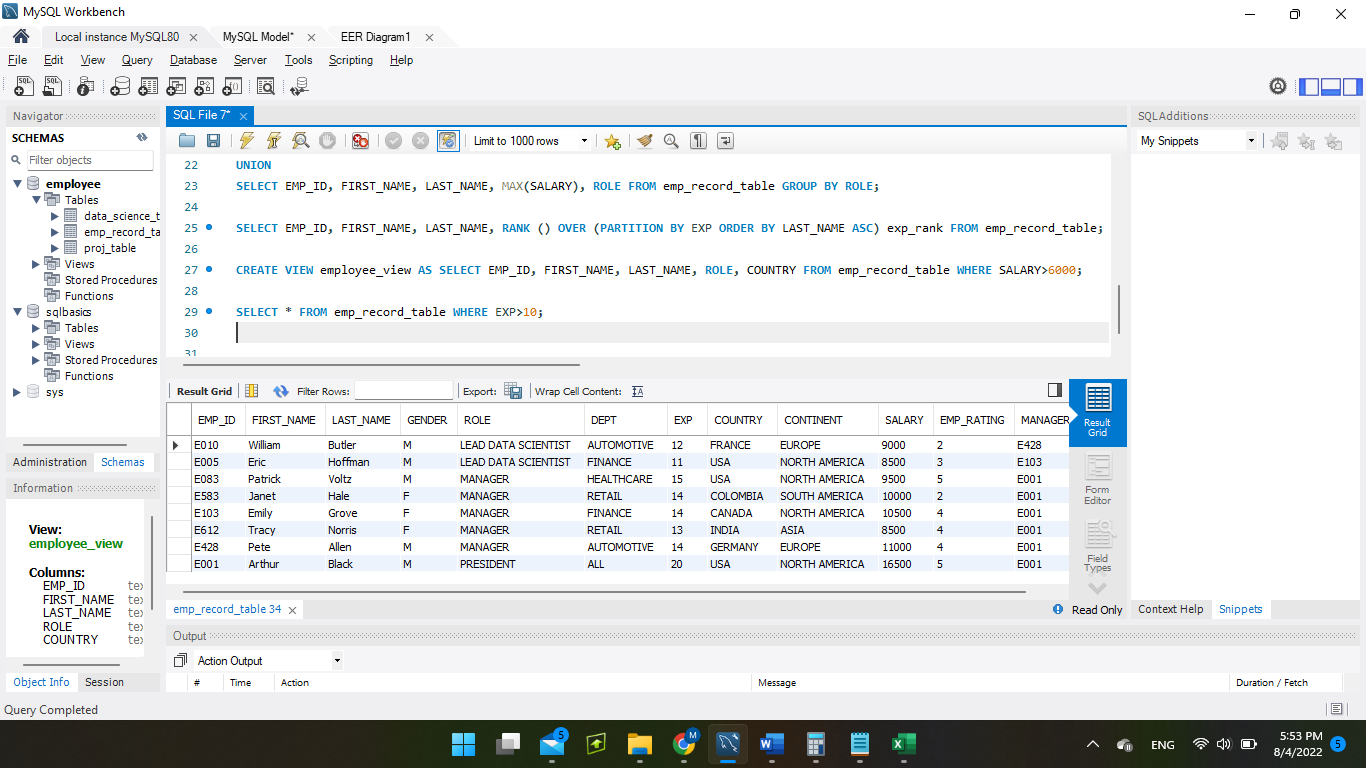


1. Write a query to create a view that displays employees in various countries whose salary is more than six thousand. Take data from the employee record table. (LESSON 5-6)

CREATE VIEW employee\_view AS SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, ROLE, COUNTRY FROM emp\_record\_table WHERE SALARY>6000;

1. Write a nested query to find employees with experience of more than ten years. Take data from the employee record table. (LESSON 8)

SELECT \* FROM emp\_record\_table WHERE EXP>10;



1. 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. (LESSON 5-6)

SELECT \* from emp\_record\_table;

delimiter &&

CREATE PROCEDURE morethan3yrsexp()

BEGIN

SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME, EXP

FROM emp\_record\_table where EXP>3;

END &&

delimiter ;;

call morethan3yrsexp();

Graphical user interface, text, application

Description automatically generated

1. Write a query using stored functions in the project table to check whether the job profile assigned to each employee in the data science team matches the organization’s set standard. (LESSON 5-6)

DELIMITER //

CREATE FUNCTION correct\_standard (EXP int)

RETURNS VARCHAR(80)

BEGIN

DECLARE correct\_standard VARCHAR(80);

IF EXP<=2 THEN SET correct\_standard='JUNIOR DATA SCIENTIST';

ELSEIF EXP BETWEEN 2 AND 5 THEN SET correct\_standard='ASSOCIATE DATA SCIENTIST';

ELSEIF EXP BETWEEN 5 AND 10 THEN SET correct\_standard='SENIOR DATA SCIENTIST';

ELSEIF EXP BETWEEN 10 AND 12 THEN SET correct\_standard='LEAD DATA SCIENTIST';

ELSE SET correct\_standard='MANAGER';

END IF;

RETURN correct\_standard;

END; //

DELIMITER ;

**WASN’T ABLE TO RUN THIS PART OF THE CODE**

The standard being:

For an employee with experience less than or equal to 2 years assign 'JUNIOR DATA SCIENTIST',

For an employee with the experience of 2 to 5 years assign 'ASSOCIATE DATA SCIENTIST',

For an employee with the experience of 5 to 10 years assign 'SENIOR DATA SCIENTIST',

For an employee with the experience of 10 to 12 years assign 'LEAD DATA SCIENTIST',

For an employee with the experience of 12 to 16 years assign 'MANAGER'.

1. 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. (LESSON 12)

CREATE INDEX indx ON emp\_record\_table (FIRST\_NAME);

EXPLAIN SELECT EMP\_ID, FIRST\_NAME, LAST\_NAME FROM emp\_record\_table WHERE FIRST\_NAME=”Eric”;

Graphical user interface, text, application, email

Description automatically generated

**I GOT ERROR CODE 1170 HERE, KEY LENGTH SPECIFICATION???**

1. 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). (LESSON 5-6)

SELECT \* , 0.05\*salary\*emp\_rating as bonus FROM emp\_record\_table;

Graphical user interface, text, application, email

Description automatically generated

1. Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table. (LESSON 5-6)

SELECT COUNTRY, AVG(SALARY) AS AVG\_SALARY FROM emp\_record\_table GROUP BY COUNTRY WITH ROLLUP;

Graphical user interface, text, application

Description automatically generated