
SQL Training

Course-End Project Problem Statement



Get Certified. Get Ahead.



ScienceQtech Employee Performance Mapping

Problem scenario:

ScienceQtech is a startup that works in the Data Science field. ScienceQtech has worked on fraud detection, market basket, self-driving cars, supply chain, algorithmic early detection of lung cancer, customer sentiment, and the drug discovery field. With the annual appraisal cycle around the corner, the HR department has asked you (Junior Database Administrator) to generate reports on employee details, their performance, and on the project that the employees have undertaken, to analyze the employee database and extract specific data based on different requirements.

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.

Note: You must download the dataset from the course resource section in LMS and create a table to perform the above objective

3 CSV files. Min data.

Dataset description:

emp_record_table: It contains the information of all the employees.

- EMP_ID – ID of the employee
- FIRST_NAME – First name of the employee
- LAST_NAME – Last name of the employee
- GENDER – Gender of the employee
- ROLE – Post of the employee
- DEPT – Field of the employee
- EXP – Years of experience the employee has
- COUNTRY – Country in which the employee is presently living
- CONTINENT – Continent in which the country is
- SALARY – Salary of the employee
- EMP_RATING – Performance rating of the employee
- MANAGER_ID – The manager under which the employee is assigned
- PROJ_ID – The project on which the employee is working or has worked on

Proj_table: It contains information about the projects.

- PROJECT_ID – ID for the project
- PROJ_Name – Name of the project
- DOMAIN – Field of the project

- START_DATE – Day the project began
- CLOSURE_DATE – Day the project was or will be completed
- DEV_QTR – Quarter in which the project was scheduled
- STATUS – Status of the project currently

Data_science_team: It contains information about all the employees in the Data Science team.

- EMP_ID – ID of the employee
- FIRST_NAME – First name of the employee
- LAST_NAME – Last name of the employee
- GENDER – Gender of the employee
- ROLE – Post of the employee
- DEPT – Field of the employee
- EXP – Years of experience the employee has
- COUNTRY – Country in which the employee is presently living
- CONTINENT – Continent in which the country is

Unclean data ✓
SQL only takes yyy-mm-dd
Used Excel to import
data

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.
2. Create an ER diagram for the given **employee** database.
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.
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

Pretty simple.
(not pres-)

5. 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.

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

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.

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.

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.

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

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.

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

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.

14. 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.

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'.

15. 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.
16. 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).
17. Write a query to calculate the average salary distribution based on the continent and country. Take data from the employee record table.

□

SQL SCRIPT FOR EMPLOYEE MAPPING: SQL

```
create database emp_performance;  
show schemas;  
use emp_performance;  
show tables;  
-- create the structure of the table in the schema
```

```
CREATE TABLE emp_record_table
```

```
(  
  EMP_ID VARCHAR(10) PRIMARY KEY,  
  FIRST_NAME VARCHAR(50),  
  LAST_NAME VARCHAR(50),  
  GENDER CHAR(1),  
  ROLE VARCHAR(50),  
  DEPT VARCHAR(50),  
  EXP INT,  
  COUNTRY VARCHAR(50),  
  CONTINENT VARCHAR(50),  
  SALARY INT,  
  EMP_RATING INT,  
  MANAGER_ID VARCHAR(10),  
  PROJ_ID VARCHAR(10)  
);
```

```
-- create structure of the proj tabel
```

```
CREATE TABLE Proj_table (
```

```
  PROJECT_ID varchar(10) PRIMARY KEY,  
  PROJ_Name VARCHAR(50) NOT NULL,  
  DOMAIN VARCHAR(50) NOT NULL,
```

*Study CSV
structure &
decide your
SQL data type*

```

START_DATE DATE NOT NULL,
CLOSURE_DATE DATE,
DEV_QTR CHAR(2) NOT NULL,
STATUS VARCHAR(20) NOT NULL
);
DROP TABLE Proj_table;
-- Create ctructure for table data science team

```

CREATE TABLE Data_science_team (

```

EMP_ID VARCHAR(10) PRIMARY KEY,
FIRST_NAME VARCHAR(50) NOT NULL,
LAST_NAME VARCHAR(50) NOT NULL,
GENDER char(1) NOT NULL,
ROLE VARCHAR(50) NOT NULL,
DEPT VARCHAR(50) NOT NULL,
EXP INT NOT NULL,
COUNTRY VARCHAR(50) NOT NULL,
CONTINENT VARCHAR(50) NOT NULL
);

```

-- check creation is correct

show tables;

select * FROM data_science_team;

SELECT * FROM Proj_table;

-- Now load the emp record csv file from computer to Mysql

LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server
8.3/Uploads/emp_record_table.csv' INTO TABLE emp_record_table

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

IGNORE 1 ROWS;

-- you can only upload csv from upload folder in MYSQL path

C:/ProgramData/MySQL/MySQL Server 8.3/Uploads/

SHOW variables like "secure_file_priv";

-- check if data is correctly loaded in emp table

SELECT * FROM emp_record_table;

-- import data from proj csv

LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.3/Uploads/proj_table.csv'
INTO TABLE Proj_table

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

IGNORE 1 ROWS;

-- error due to incoorect data format for input for Proj iD change it to varchar

-- error again due to data in incorrect date format. Clean data in excel before importing

-- finally import data from data science team table

LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server
8.3/Uploads/data_science_team.csv' INTO TABLE Data_science_team

FIELDS TERMINATED BY ','

LINES TERMINATED BY '\n'

IGNORE 1 ROWS;

-- check all tables are imported correctly

*Mind you:
Upload CSV sheet
only from this
path
else gives
error.*

```
SELECT * FROM Data_science_team;
SELECT * FROM Proj_table;
-- All Data successfully imported
```

-- Task no 3

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT
FROM emp_record_table;
```

-- Task 4 queries for different employee ratings

-- Less than 2

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING < 2;
```

-- Greater than 4

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING > 4;
```

-- Between 2 and 4

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING
FROM emp_record_table
WHERE EMP_RATING BETWEEN 2 AND 4;
```

-- Task 5

/* 5. 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.

*/

```
SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS NAME
FROM emp_record_table
WHERE DEPT = 'Finance';
```

-- Task No 6

/* 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 MANAGER_ID, COUNT(*) AS NUM_REPORTERS
FROM emp_record_table
GROUP BY MANAGER_ID;
```

Rudimentary

-- tASK 7

/*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 EMP_ID, FIRST_NAME, LAST_NAME, DEPT
FROM emp_record_table
WHERE DEPT = 'Healthcare')
UNION
(SELECT EMP_ID, FIRST_NAME, LAST_NAME, DEPT
FROM emp_record_table
WHERE DEPT = 'Finance');
```

*This is tricky
Use Windows
function.*

/* tASK NO 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 DEPT,
       EMP_ID,
       FIRST_NAME,
       LAST_NAME,
       ROLE,
       EMP_RATING,
       MAX(EMP_RATING) OVER(PARTITION BY DEPT) AS MAX_EMP_RATING
FROM emp_record_table
GROUP BY DEPT, EMP_ID, FIRST_NAME, LAST_NAME, ROLE;
```

Task 9

/* tASK NO 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.*/

```
SELECT ROLE,
       MIN(SALARY) AS MIN_SALARY,
       MAX(SALARY) AS MAX_SALARY
FROM emp_record_table
GROUP BY ROLE;
```


Task 10

/* Task no 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, GENDER, ROLE, DEPT, EXP, COUNTRY,
CONTINENT, SALARY, EMP_RATING, MANAGER_ID, PROJ_ID,
-- Use the RANK () function to assign ranks based on the experience column
RANK () OVER (ORDER BY EXP DESC) AS EXP_RANK
-- From the employee record table
FROM emp_record_table;

*Create a view then
use the view by
select statement*

Task 11

/*Task no 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 VIEW high_earners AS
SELECT FIRST_NAME, LAST_NAME, COUNTRY, SALARY
FROM emp_record_table
WHERE SALARY > 6000;
SELECT * FROM high_earners;

Task 12

/* Task no 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
);

Task 13

/* Task No. 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.*/
DELIMITER &&

```
CREATE PROCEDURE get_threeplusexperienced_employees()
BEGIN
  SELECT *
  FROM emp_record_table
  WHERE EXP >= 3;
END &&
Call get_threeplusexperienced_employees();
```

Need to understand
what Delimiter. Why only in
My SQL.
Note diff between
function & procedure

Task 14

/*

14. 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.

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'.

*/

```
SET GLOBAL log_bin_trust_function_creators = 1;
```

```
DELIMITER &&
```

```
CREATE FUNCTION assign_job_profile(exp INT)
```

```
RETURNS VARCHAR(50)
```

```
BEGIN
```

```
IF exp <= 2 THEN
```

```
  RETURN 'JUNIOR DATA SCIENTIST';
```

```
ELSEIF exp <= 5 THEN
```

```
  RETURN 'ASSOCIATE DATA SCIENTIST';
```

```
ELSEIF exp <= 10 THEN
```

The function
can throw error
w/o this.
Not clear. This
is some security
feature to protect
function usage.

```

        RETURN 'SENIOR DATA SCIENTIST';

ELSEIF exp <= 12 THEN

        RETURN 'LEAD DATA SCIENTIST';

ELSE

        RETURN 'MANAGER';

END IF;

END &&

DELIMITER ;

SELECT EMP_ID, LAST_NAME, ROLE, assign_job_profile(EXP) AS new_role
FROM emp_record_table;

```

Manually check.

Give table of all data role as per policy.

Task 15

/*

15. 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.

*/

-- Analyze speed after each query to identify suitable columns for indexing

-- First try on FIRST_NAME col

CREATE INDEX idx_firstname ON emp_record_table (FIRST_NAME);

Task 16

/*

16. 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 * EMP_RATING * 0.05 AS BONUS

FROM emp_record_table;

Task 17

/*

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

*/

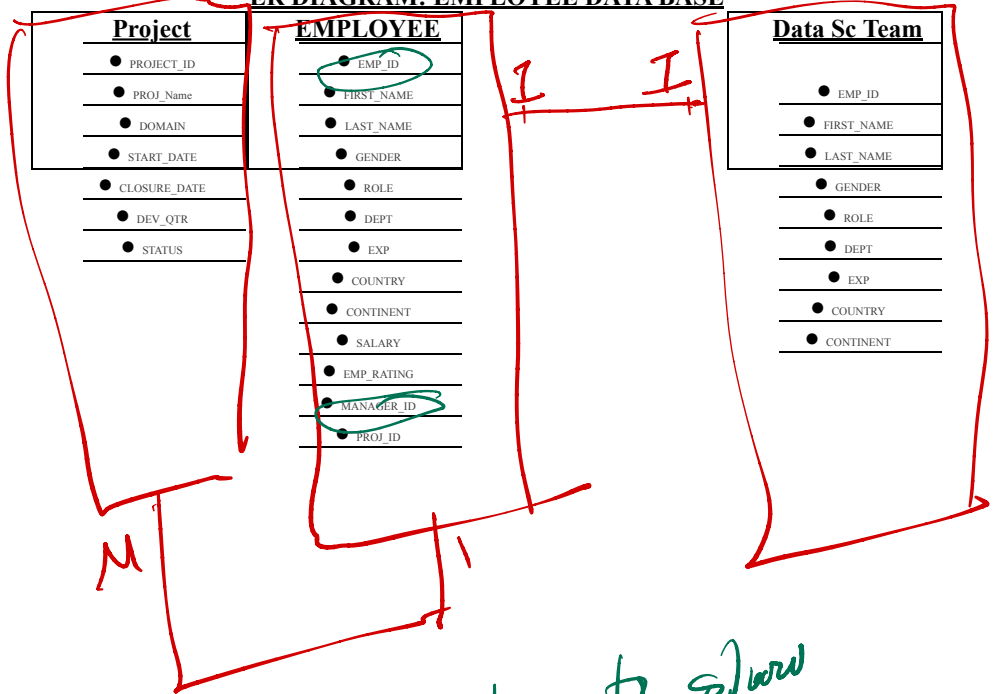
```
SELECT  
  CONTINENT, COUNTRY, AVG(SALARY) AS AVG_SALARY  
FROM  
  emp_record_table  
GROUP BY CONTINENT, COUNTRY;
```

*Test all straight fwd.
Simple straight fwd proj.
Explanation was clear & concise.
cheat sheet useful.*

ER diagram

PROJ 1- EMPLOYEE PERFORMANCE MAPPING

ER DIAGRAM: EMPLOYEE DATA BASE



I don't know how to show
relationship within table
like Emp ID & Manager ID

