Science Q-Tech Employee Performance Mapping.

DESCRIPTION

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.

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

data_science_team

EMP_ID	FIRST_NAME	LAST_NAME	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINENT
E260	Roy	Collins	M	SENIOR DATA SCIENTIST	RETAIL	7	INDIA	ASIA
E245	Nian	Zhen	M	SENIOR DATA SCIENTIST	RETAIL	6	CHINA	ASIA
E620	Katrina	Allen	F	JUNIOR DATA SCIENTIST	RETAIL	2	INDIA	ASIA
E640	Jenifer	Jhones	F	JUNIOR DATA SCIENTIST	RETAIL	1	COLOMBIA	SOUTH AMERICA
E403	Steve	Hoffman	M	ASSOCIATE DATA SCIENTIST	FINANCE	4	USA	NORTH AMERICA
E204	Karene	Nowak	F	SENIOR DATA SCIENTIST	AUTOMOTIVE	8	GERMANY	EUROPE
E057	Dorothy	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	9	USA	NORTH AMERICA
E010	William	Butler	M	LEAD DATA SCIENTIST	AUTOMOTIVE	12	FRANCE	EUROPE
E478	David	Smith	M	ASSOCIATE DATA SCIENTIST	RETAIL	3	COLOMBIA	SOUTH AMERICA
E005	Eric	Hoffman	M	LEAD DATA SCIENTIST	FINANCE	11	USA	NORTH AMERICA
E052	Dianna	Wilson	F	SENIOR DATA SCIENTIST	HEALTHCARE	6	CANADA	NORTH AMERICA
E505	Chad	Wilson	M	ASSOCIATE DATA SCIENTIST	HEALTHCARE	5	CANADA	NORTH AMERICA
E532	Claire	Brennan	F	ASSOCIATE DATA SCIENTIST	AUTOMOTIVE	3	GERMANY	EUROPE

emp_record_table

EMP_ID	FIRST_NAM	LAST_NAM	GENDER	ROLE	DEPT	EXP	COUNTRY	CONTINEN	SALARY	EMP_RA	MANAGER
E260	Roy	Collins	M	SENIOR DA	RETAIL	7	INDIA	ASIA	7000	3	E583
E245	Nian	Zhen	M	SENIOR DA	RETAIL	6	CHINA	ASIA	6500	2	E583
E620	Katrina	Allen	F	JUNIOR DA	RETAIL	2	INDIA	ASIA	3000	1	E612
E640	Jenifer	Jhones	F	JUNIOR DA	RETAIL	1	COLOMBIA	SOUTH AM	2800	4	E612
E403	Steve	Hoffman	M	ASSOCIATE	FINANCE	4	USA	NORTH AM	5000	3	E103
E204	Karene	Nowak	F	SENIOR DA	AUTOMOT	8	GERMANY	EUROPE	7500	5	E428
E057	Dorothy	Wilson	F	SENIOR DA	HEALTHCA	9	USA	NORTH AM	7700	1	E083
E010	William	Butler	M	LEAD DATA	AUTOMOT	12	FRANCE	EUROPE	9000	2	E428
E478	David	Smith	M	ASSOCIATE	RETAIL	3	COLOMBIA	SOUTH AM	4000	4	E583
E005	Eric	Hoffman	M	LEAD DATA	FINANCE	11	USA	NORTH AM	8500	3	E103
E052	Dianna	Wilson	F	SENIOR DA	HEALTHCA	6	CANADA	NORTH AM	5500	5	E083
E505	Chad	Wilson	M	ASSOCIATE	HEALTHCA	5	CANADA	NORTH AM	5000	2	E083
E532	Claire	Brennan	F	ASSOCIATE	AUTOMOT	3	GERMANY	EUROPE	4300	1	E428
E083	Patrick	Voltz	M	MANAGER	HEALTHCA	15	USA	NORTH AM	9500	5	E001
E583	Janet	Hale	F	MANAGER	RETAIL	14	COLOMBIA	SOUTH AM	10000	2	E001
E103	Emily	Grove	F	MANAGER	FINANCE	14	CANADA	NORTH AM	10500	4	E001
E612	Tracy	Norris	F	MANAGER	RETAIL	13	INDIA	ASIA	8500	4	E001
E428	Pete	Allen	M	MANAGER	AUTOMOT	14	GERMANY	EUROPE	11000	4	E001
E001	Arthur	Black	M	PRESIDENT	ALL	20	USA	NORTH AM	16500	5	E001

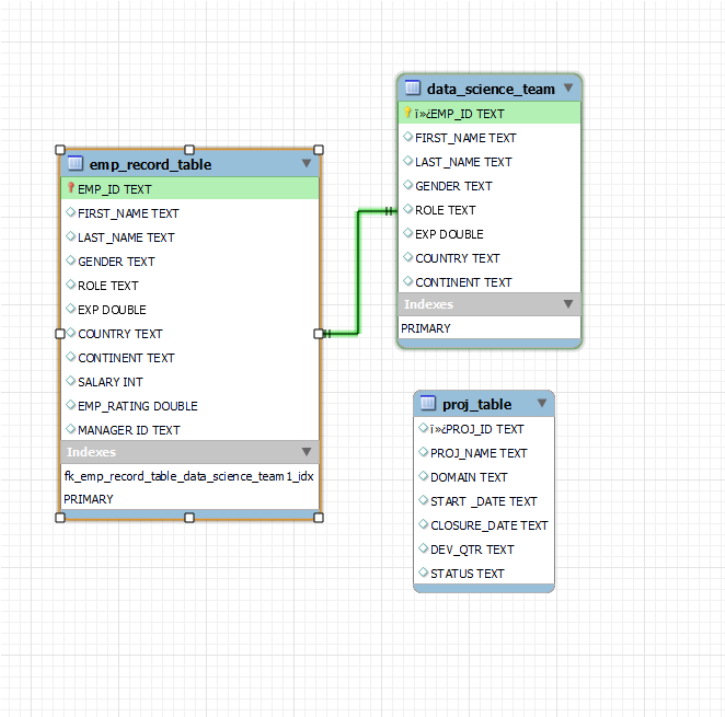
proj_table

PROJ_ID	PROJ_NAM	DOMAIN	START _DATE	CLOSURE_DATE	DEV_QTR	STATUS
P103	Drug Disco	HEALTHCA	4/6/2021	6/20/2021	Q1	DONE
P105	Fraud Dete	FINANCE	4/11/2021	6/25/2021	Q1	DONE
P208	Algorithmi	FINANCE	1/16/2022	3/27/2022	Q4	YTS
P109	Market Bas	RETAIL	4/12/2021	6/30/2021	Q1	DELAYED
P204	Supply Cha	AUTOMOT	7/15/2021	9/28/2021	Q2	WIP
P406	Customer S	RETAIL	7/9/2021	9/24/2021	Q2	WIP
P302	Early Detec	HEALTHCA	10/8/2021	12/18/2021	Q3	YTS
P201	Self Driving	AUTOMOT	1/12/2022	3/30/2022	Q4	YTS

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.

```
CREATE DATABASE employee;
USE employee;
```

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.

```
SELECT * FROM emp_record_table;
ALTER TABLE employee.emp_record_table RENAME COLUMN ï»¿EMP_ID TO EMP_ID;
SELECT * FROM emp_record_table;

SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT FROM employee.emp_record_table
ORDER BY DEPT,FIRST_NAME,LAST_NAME;
```

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_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM
employee.emp_record_table WHERE EMP_RATING <2;

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

SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING FROM
employee.emp_record_table WHERE EMP_RATING >=2 AND EMP_RATING <=4 ORDER BY
EMP_RATING;
```

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 employee.emp_record_table WHERE
DEPT = 'FINANCE';
```

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 A.`MANAGER ID`,count(A.`MANAGER ID`)AS REPORTERS, concat(b.FIRST_NAME, '
',B.LAST_NAME) AS NAME  FROM employee.emp_record_table AS A JOIN
employee.emp_record_table AS B
ON A.`MANAGER ID` = B.EMP_ID
GROUP BY A.`MANAGER ID` ORDER BY A.`MANAGER ID` ;
```

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 employee.emp_record_table AS FIN_DATA WHERE DEPT = 'FINANCE'
UNION All
SELECT * FROM employee.emp_record_table AS HC_DATA WHERE DEPT = 'HEALTHCARE'ORDER BY
EMP_ID;
```

```
#SELECT * FROM employee.emp_record_table AS FIN_DATA WHERE DEPT IN ( 'FINANCE','HEALTHCARE')
```

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 MAX_RATING,CONCAT(EMP_RATING , "/",   MAX(EMP_RATING) OVER
(PARTITION BY DEPT) )AS EFF,REPEAT("*", EMP_RATING) AS STARS
FROM
```

```
employee.emp_record_table
ORDER BY  EMP_RATING DESC;
```

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), MAX(SALARY)
FROM employee.emp_record_table
WHERE `ROLE`!= 'PRESIDENT'
GROUP BY `ROLE`;
```

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, DEPT, EXP,
    RANK() OVER(ORDER BY EXP DESC) AS 'Rank'
FROM employee.emp_record_table
ORDER BY 'Rank';
```

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 V_COUNTRY_SAL AS
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EXP,COUNTRY,SALARY
FROM employee.emp_record_table
WHERE SALARY>6000;
```

```
SELECT * FROM V_COUNTRY_SAL;
```

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

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, EXP, DEPT, EMP_RATING FROM
employee.emp_record_table WHERE EMP_ID IN (SELECT EMP_ID FROM  employee.emp_record_table
WHERE EXP >10);
```

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 EMP_DETAILS()
BEGIN
        SELECT * FROM employee.emp_record_table WHERE EXP>3;
END //

DELIMITER //;enecccbettbglghiuukeegdcbhvdrrenbjfgcnrlngfi

CALL EMP_DETAILS();
```

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'.

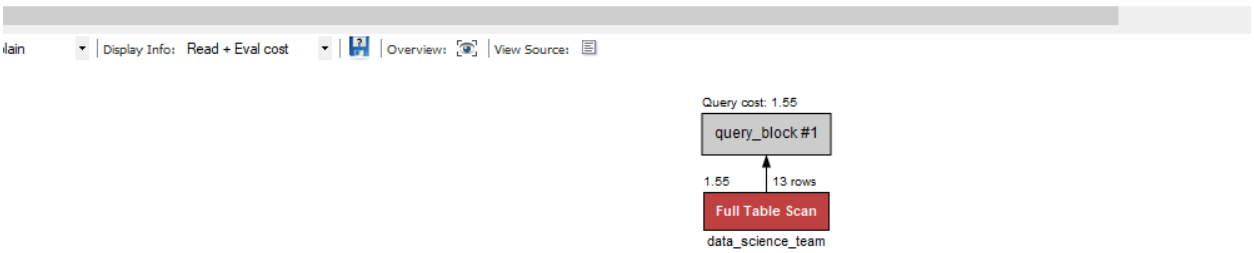
```
DELIMITER $$ ;
ALTER FUNCTION Role_VERIFY (
   EXP double
)
RETURNS VARCHAR(200)
DETERMINISTIC
BEGIN
   DECLARE Role_Validation VARCHAR(200);
   IF EXP<=2 THEN
      SET Role_Validation = 'JUNIOR DATA SCIENTIST';
   ELSEIF (EXP>=2 AND
         EXP<=5) THEN
      SET Role_Validation = 'ASSOCIATE DATA SCIENTIST';
```

```
   ELSEIF (EXP>=5 AND
          EXP<=10) THEN
       SET Role_Validation = 'SENIOR DATA SCIENTIST';
            ELSEIF (EXP>=10 AND
          EXP<=12) THEN
       SET Role_Validation = 'LEAD DATA SCIENTIST';
            ELSEIF (EXP>=12 AND
          EXP<=16) THEN
       SET Role_Validation = 'MANAGER';
   END IF;
   -- return the customer occupation
   RETURN (Role_Validation);
END $$
DELIMITER $$;


SELECT *,
CASE ROLE WHEN Role_VERIFY(EXP)
 THEN 'VALID' ELSE 'INVALID'
 END AS VALIDATION
 FROM employee.data_science_team;
```

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.

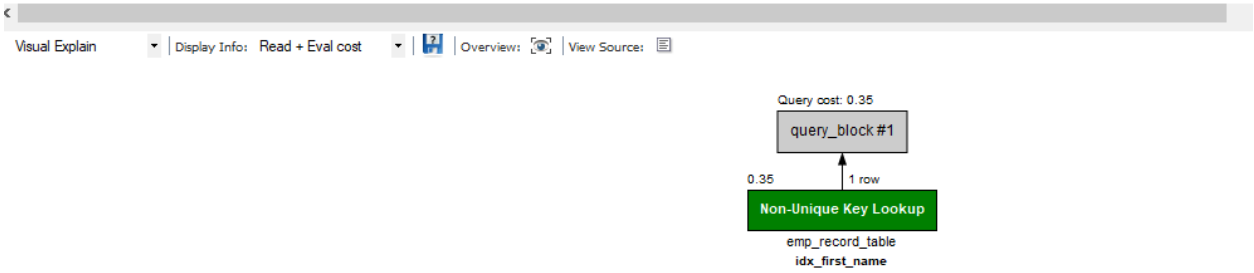
```
ALTER TABLE employee.emp_record_table DROP INDEX idx_first_name;

CREATE INDEX idx_first_name

ON employee.emp_record_table(FIRST_NAME(20));



SELECT * FROM employee.emp_record_table

WHERE FIRST_NAME='Eric';
```

```
SELECT * FROM employee.emp_record_table
WHERE FIRST_NAME='Eric';
```



Query cost before creating the index taking FIRST_NAME is 1.55

```
4 •  CREATE INDEX idx_first_name
5    ON employee.emp_record_table(FIRST_NAME(20));
6 •  SELECT * FROM employee.emp_record_table
7    WHERE FIRST_NAME='Eric';
```



.

Query cost decreased by 1.2 to get the final query cost as 0.35

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 EMP_ID, FIRST_NAME, LAST_NAME, GENDER, EXP, DEPT, EMP_RATING,
0.05*SALARY*EMP_RATING AS BONUS FROM  employee.emp_record_table ORDER BY BONUS
DESC;
```

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

```
SELECT EMP_ID, FIRST_NAME, LAST_NAME, GENDER, EXP, DEPT, EMP_RATING, AVG(SALARY) AS AVG_SAL
FROM  employee.emp_record_table GROUP BY CONTINENT,COUNTRY;
```