**ScienceQtech Employee Performance Mapping**

#(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 resource

Create database employee;

Use employee;

Create table data\_science\_team(

employee\_id varchar(5),

first\_name char(12),

last\_name char(12),

gender char(1),

role char(50),

department char(20),

experience int unsigned,

country char(10),

continent char(25)

);

Create table proj\_table(

project\_id varchar(5),

project\_name char(30),

domain char(15),

start\_date varchar(10),

closure\_date varchar(10),

dev\_qtr varchar(2),

status char(10)

);

Create table emp\_record\_table(

employee\_id varchar(5),

first\_name char(12),

last\_name char(12),

gender char(1),

role char(50),

department char(20),

experience int unsigned,

country char(10),

continent char(25),

salary int unsigned,

employee\_rating int unsigned,

manager\_id varchar(5),

project\_id varchar(5)

);

Insert into data\_science\_team(employee\_id, first\_name, last\_name, gender, role, department, experience, country, continent)

values

('E005', 'Eric', 'Hoffman', 'M', 'LEAD DATA SCIENTIST', 'FINANCE', 11, 'USA', 'NORTH AMERICA'),

('E010', 'William', 'Butler', 'M', 'LEAD DATA SCIENTIST', 'AUTOMOTIVE', 12, 'FRANCE', 'EUROPE'),

('E052', 'Dianna', 'Wilson', 'F', 'SENIOR DATA SCIENTIST', 'HEALTHCARE', 6, 'CANADA', 'NORTH AMERICA'),

('E057', 'Dorothy', 'Wilson', 'F', 'SENIOR DATA SCIENTIST', 'HEALTHCARE', 9, 'USA', 'NORTH AMERICA'),

('E204', 'Karene', 'Nowak', 'F', 'SENIOR DATA SCIENTIST', 'AUTOMOTIVE', 8, 'GERMANY', 'EUROPE'),

('E245', 'Nian', 'Zhen', 'M', 'SENIOR DATA SCIENTIST', 'RETAIL', 6, 'CHINA', 'ASIA'),

('E260', 'Roy', 'Collins', 'M', 'SENIOR DATA SCIENTIST', 'RETAIL', 7, 'INDIA', 'ASIA'),

('E403', 'Steve', 'Hoffman', 'M', 'ASSOCIATE DATA SCIENTIST', 'FINANCE', 4, 'USA', 'NORTH AMERICA'),

('E478', 'David', 'Smith', 'M', 'ASSOCIATE DATA SCIENTIST', 'RETAIL', 3, 'COLOMBIA', 'SOUTH AMERICA'),

('E505', 'Chad', 'Wilson', 'M', 'ASSOCIATE DATA SCIENTIST', 'HEALTHCARE', 5, 'CANADA', 'NORTH AMERICA'),

('E532', 'Claire', 'Brennan', 'F', 'ASSOCIATE DATA SCIENTIST', 'AUTOMOTIVE', 3, 'GERMANY', 'EUROPE'),

('E620', 'Katrina', 'Allen', 'F', 'JUNIOR DATA SCIENTIST', 'RETAIL', 2, 'INDIA', 'ASIA'),

('E640', 'Jenifer', 'Jhones', 'F', 'JUNIOR DATA SCIENTIST', 'RETAIL', 1, 'COLOMBIA', 'SOUTH AMERICA');

Insert into proj\_table(project\_id, project\_name, domain, start\_date, closure\_date, dev\_qtr, status)

values

('P103', 'Drug Discovery', 'HEALTHCARE', '2021-04-06', '2021-06-20', 'Q1', 'DONE'),

('P105', 'Fraud Detection', 'FINANCE', '2021-04-11', '2021-06-25', 'Q1', 'DONE'),

('P109', 'Market Basket Analysis', 'RETAIL', '2021-04-12', '2021-06-30', 'Q1', 'DELAYED'),

('P204', 'Supply Chain Management', 'AUTOMOTIVE', '2021-07-15', '2021-09-28', 'Q2', 'WIP'),

('P302', 'Early Detection of Lung Cancer', 'HEALTHCARE', '2021-10-08', '2021-12-18', 'Q3', 'YTS'),

('P406', 'Customer Sentiment Analysis', 'RETAIL', '2021-07-09', '2021-09-24', 'Q2', 'WIP');

Insert into emp\_record\_table(employee\_id, first\_name, last\_name, gender, role, department, experience, country, continent, salary, employee\_rating, manager\_id, project\_id)

values

('E001', 'Arthur', 'Black', 'M', 'PRESIDENT', 'ALL', 20, 'USA', 'NORTH AMERICA', 16500, 5, NULL, NULL),

('E005', 'Eric', 'Hoffman', 'M', 'LEAD DATA SCIENTIST', 'FINANCE', 11, 'USA', 'NORTH AMERICA', 8500, 3, 'E103', 'P105'),

('E010', 'William', 'Butler', 'M', 'LEAD DATA SCIENTIST', 'AUTOMOTIVE', 12, 'FRANCE', 'EUROPE', 9000, 2, 'E428', 'P204'),

('E052', 'Dianna', 'Wilson', 'F', 'SENIOR DATA SCIENTIST', 'HEALTHCARE', 6, 'CANADA', 'NORTH AMERICA', 5500, 5, 'E083', 'P103'),

('E057', 'Dorothy', 'Wilson', 'F', 'SENIOR DATA SCIENTIST', 'HEALTHCARE', 9, 'USA', 'NORTH AMERICA', 7700, 1, 'E083', 'P302'),

('E083', 'Patrick', 'Voltz', 'M', 'MANAGER', 'HEALTHCARE', 15, 'USA', 'NORTH AMERICA', 9500, 5, 'E001', NULL),

('E103', 'Emily', 'Grove', 'F', 'MANAGER', 'FINANCE', 14, 'CANADA', 'NORTH AMERICA', 10500, 4, 'E001', NULL),

('E204', 'Karene', 'Nowak', 'F', 'SENIOR DATA SCIENTIST', 'AUTOMOTIVE', 8, 'GERMANY', 'EUROPE', 7500, 5, 'E428', 'P204'),

('E245', 'Nian', 'Zhen', 'M', 'SENIOR DATA SCIENTIST', 'RETAIL', 6, 'CHINA', 'ASIA', 6500, 2, 'E583', 'P109'),

('E260', 'Roy', 'Collins', 'M', 'SENIOR DATA SCIENTIST', 'RETAIL', 7, 'INDIA', 'ASIA', 7000, 3, 'E583', 'NA'),

('E403', 'Steve', 'Hoffman', 'M', 'ASSOCIATE DATA SCIENTIST', 'FINANCE', 4, 'USA', 'NORTH AMERICA', 5000, 3, 'E103', 'P105'),

('E428', 'Pete', 'Allen', 'M', 'MANAGER', 'AUTOMOTIVE', 14, 'GERMANY', 'EUROPE', 11000, 4, 'E001', NULL),

('E478', 'David', 'Smith', 'M', 'ASSOCIATE DATA SCIENTIST', 'RETAIL', 3, 'COLOMBIA', 'SOUTH AMERICA', 4000, 4, 'E583', 'P109'),

('E505', 'Chad', 'Wilson', 'M', 'ASSOCIATE DATA SCIENTIST', 'HEALTHCARE', 5, 'CANADA', 'NORTH AMERICA', 5000, 2, 'E083', 'P103'),

('E532', 'Claire', 'Brennan', 'F', 'ASSOCIATE DATA SCIENTIST', 'AUTOMOTIVE', 3, 'GERMANY', 'EUROPE', 4300, 1, 'E428', 'P204'),

('E583', 'Janet', 'Hale', 'F', 'MANAGER', 'RETAIL', 14, 'COLOMBIA', 'SOUTH AMERICA', 10000, 2, 'E001', NULL),

('E612', 'Tracy', 'Norris', 'F', 'MANAGER', 'RETAIL', 13, 'INDIA', 'ASIA', 8500, 4, 'E001', NULL),

('E620', 'Katrina', 'Allen', 'F', 'JUNIOR DATA SCIENTIST', 'RETAIL', 2, 'INDIA', 'ASIA', 3000, 1, 'E612', 'P406'),

('E640', 'Jenifer', 'Jhones', 'F', 'JUNIOR DATA SCIENTIST', 'RETAIL', 1, 'COLOMBIA', 'SOUTH AMERICA', 2800, 4, 'E612', 'P406');

#(2) Create an ER diagram for the given employee database

-- steps are: database -> reverse engineer -> follow prompt

#(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 employee\_id, concat(first\_name, ' ', last\_name) as full\_name, gender, department

FROM emp\_record\_table

Order by department, full\_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 employee\_id, concat(first\_name, ' ', last\_name) as full\_name, gender, department, employee\_rating

From emp\_record\_table

Where employee\_rating <2 or employee\_rating > 4 or (employee\_rating between 2 and 4);

#(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

Order by department = '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 manager.employee\_id, concat (manager.first\_name, ' ', manager.last\_name) as full\_name, COUNT(report.employee\_id) AS num\_reporters

FROM emp\_record\_table AS manager

JOIN emp\_record\_table AS report ON manager.employee\_id = report.manager\_id

GROUP BY manager.employee\_id, manager.first\_name, manager.last\_name

HAVING COUNT(report.employee\_id) > 0;

#(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 employee\_id, concat(first\_name, ' ', last\_name) as full\_name, gender, role, department, experience, country, continent, salary, employee\_rating, manager\_id, project\_id

FROM emp\_record\_table

WHERE department = 'HEALTHCARE'

UNION

SELECT employee\_id, concat(first\_name, ' ', last\_name) as full\_name, gender, role, department, experience, country, continent, salary, employee\_rating, manager\_id, project\_id

FROM emp\_record\_table

WHERE department = 'FINANCE';

#(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 employee\_id, concat(first\_name, ' ', last\_name) as full\_name, role, department, employee\_rating

From emp\_record\_table

where (department, employee\_rating) In (

Select department, max(employee\_rating)

From emp\_record\_table

group by 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

Select department, min(salary) as min\_salary, max(salary) as max\_salary

From emp\_record\_table

Group by department;

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

SELECT employee\_id, concat (first\_name, ' ', last\_name) as full\_name, experience,

RANK() OVER (ORDER BY experience DESC) AS experience\_rank

FROM emp\_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

Create view highest\_salary\_employees as

Select employee\_id, concat(first\_name, ' ',last\_name) as full\_name, gender, department, country, salary

From emp\_record\_table

Where salary > 6000;

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

Select employee\_id, concat(first\_name, ' ', last\_name) as full\_name, experience

From emp\_record\_table

Where experience > 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 GetEmployeesWithExperience()

BEGIN

SELECT employee\_id, concat(first\_name, ' ', last\_name) as full\_name, gender, role, department, experience, country, continent, salary, employee\_rating, manager\_id, project\_id

FROM emp\_record\_table

WHERE experience > 3;

END //

DELIMITER ;

Call GetEmployeesWithExperience();

#(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 //

CREATE FUNCTION GetStandardRole(experience INT)

RETURNS VARCHAR(30)

DETERMINISTIC

BEGIN

DECLARE role VARCHAR(30);

IF experience <= 2 THEN

SET role = 'JUNIOR DATA SCIENTIST';

ELSEIF experience > 2 AND experience <= 5 THEN

SET role = 'ASSOCIATE DATA SCIENTIST';

ELSEIF experience > 5 AND experience <= 10 THEN

SET role = 'SENIOR DATA SCIENTIST';

ELSEIF experience > 10 AND experience <= 12 THEN

SET role = 'LEAD DATA SCIENTIST';

ELSEIF experience > 12 AND experience <= 16 THEN

SET role = 'MANAGER';

ELSE

SET role = 'UNASSIGNED';

END IF;

RETURN role;

END //

DELIMITER ;

SELECT employee\_id,

first\_name,

last\_name,

experience,

GetStandardRole(experience) AS role

FROM emp\_record\_table;

#(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

EXPLAIN SELECT \* FROM emp\_record\_table WHERE first\_name = 'Eric';

CREATE INDEX idx\_first\_name ON emp\_record\_table (first\_name);

SHOW INDEX FROM emp\_record\_table WHERE Key\_name = 'idx\_first\_name';

#(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 employee\_id, concat(first\_name, ' ', last\_name) as full\_name, salary, employee\_rating, (0.05 \* salary \* employee\_rating) as Bonus

From emp\_record\_table;

#(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;