**Career stimulation: Employee performance Mapping**

1. **Create emp\_record\_table**
2. **create** **table** emp\_record\_table(
3. emp\_id **varchar**(10) **primary** **key**,
4. first\_name **varchar**(50),
5. last\_name **varchar**(50),
6. gender **varchar**(10),
7. role **varchar**(100),
8. dept **varchar**(100),
9. exp **int**,
10. country **varchar**(100),
11. continent **varchar**(100),
12. salary **int**,
13. emp\_rating **decimal**(10,2),
14. manager\_id **varchar**(50),
15. proj\_id **varchar**(50));

A screenshot of a computer

AI-generated content may be incorrect.

* **import data in emp\_record\_table**

I used import data function directly transform data from file to table

**A screenshot of a computer

AI-generated content may be incorrect.**

**2. create data\_science\_team table**

create table fullstack.data\_science\_team(

emp\_id varchar(10) primary key,

first\_name varchar(50),

last\_name varchar(50),

gender varchar(10),

role varchar(100),

dept varchar(100),

exp int,

country varchar(100),

continent varchar(100));

**A screenshot of a computer

AI-generated content may be incorrect.**

* **Import data in data\_science\_team table**

I used import data function directly transform data from file to table

**A screenshot of a computer

AI-generated content may be incorrect.**

**3. create proj\_table**

create table fullstack.proj\_table1(

proj\_id varchar(50) primary key,

proj\_name varchar(100),

domain varchar(100),

start\_date varchar(10),

closure\_date varchar(10),

dev\_qtr varchar(10),

status varchar(50));

**A screenshot of a computer

AI-generated content may be incorrect.**

**--Import data in project table**

I used import data function directly transform data from file to table

**A screenshot of a computer

AI-generated content may be incorrect.**

**---create another table proj\_table( --where start\_date and closure\_date is date format)**

create table fullstack.proj\_table(

proj\_id varchar(50) primary key,

proj\_name varchar(100),

domain varchar(100),

start\_date date,

closure\_date date,

dev\_qtr varchar(10),

status varchar(50));

**A screenshot of a computer

AI-generated content may be incorrect.**

**---Insert data into proj\_table**

insert into fullstack.proj\_table (proj\_id ,

proj\_name ,

domain ,

start\_date ,

closure\_date ,

dev\_qtr,

status )

select proj\_id ,proj\_name ,domain,

CASE

WHEN start\_date LIKE '%/%' THEN STR\_TO\_DATE(start\_date, '%m/%d/%Y') -- for MM/DD/YYYY

WHEN start\_date LIKE '%-%' THEN STR\_TO\_DATE(start\_date, '%m-%d-%Y') -- for MM-DD-YYYY

ELSE NULL

end as *start\_date* ,

CASE

WHEN closure\_date LIKE '%/%' THEN STR\_TO\_DATE(closure\_date , '%m/%d/%Y') -- for MM/DD/YYYY

WHEN closure\_date LIKE '%-%' THEN STR\_TO\_DATE(closure\_date , '%m-%d-%Y') -- for MM-DD-YYYY

ELSE NULL

end as *closure\_date* ,

dev\_qtr ,

status

from fullstack.proj\_table1

;

**A screenshot of a computer

AI-generated content may be incorrect.**

**----proj\_table with correct date format**

**A screenshot of a computer

AI-generated content may be incorrect.**

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

**A diagram of a data

AI-generated content may be incorrect.**

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.**
2. **select** emp\_id,first\_name,last\_name,gender,dept
3. **from** fullstack.emp\_record\_table
4. **order** **by** dept;

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Write a query to fetch EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER, DEPARTMENT, and EMP\_RATING if the EMP\_RATING is:   
   less than two return "Below Average", greater than four return "Above Average", and between two and four return "Average".**
2. **select** emp\_id,first\_name,last\_name,gender,dept,emp\_rating,
3. **case** **when** emp\_rating < 2 **then** **"belove average"**
4. **when** emp\_rating > 4 **then** **"above average"**
5. **else** **"average"**
6. **end** **as** *"rating\_category"*
7. **from** fullstack.emp\_record\_table
8. ;

**A screenshot of a computer

AI-generated content may be incorrect.**

**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.**

1. **select** first\_name,last\_name, **concat**(first\_name,**" "**,last\_name) **as** *Name*
2. **from** fullstack.emp\_record\_table ;

**A screenshot of a computer

AI-generated content may be incorrect.**

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

select first\_name as *employee* ,count(first\_name) as *report*

from(

select *e*.emp\_id,*e*.first\_name ,*e*.manager\_id

from fullstack.emp\_record\_table *e*

inner join fullstack.emp\_record\_table *m*

on *e*.emp\_id=*m*.manager\_id )*x*

group by first\_name;

**A screenshot of a computer

AI-generated content may be incorrect.**

**Write a query to list all the employees from the healthcare and finance departments using union.**

**-- Take data from the employee record table.**

select \*

from fullstack.emp\_record\_table

where dept = "HEALTHCARE"

union

select \*

from fullstack.emp\_record\_table

where dept = "FINANCE";

A screenshot of a computer

AI-generated content may be incorrect.

**Write a query to list 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 *e1*.emp\_id, *e1*.first\_name, *e1*.last\_name, *e1*.role, *e1*.dept,*e1*.emp\_rating

from fullstack.emp\_record\_table *e1*

inner join

(select dept ,max(emp\_rating) as *max\_rating*

from fullstack.emp\_record\_table

group by dept)*x*

on *e1*.dept=*x*.dept and *e1*.emp\_rating =*x*.*max\_rating***;**

**A screenshot of a computer

AI-generated content may be incorrect.**

**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,max(salary) as *max\_salary*, min(salary) as *min\_salary*

from fullstack.emp\_record\_table

group by `role`

order by `role` ;

A screenshot of a computer

AI-generated content may be incorrect.

**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)as *rn*

from fullstack.emp\_record\_table ;

A screenshot of a computer

AI-generated content may be incorrect.

**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 employee as

select \*

from fullstack.emp\_record\_table

where salary > 6000;

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Write a query to check whether the job profile assigned to each employee in the data science team matches the organization’s set standard. The standard being:**

select \*

from (

select \*,

case when exp <= 2 then "junior data scientist"

when exp between 2 and 5 then "associate data scientist"

when exp between 5 and 10 then "senior data scientist"

when exp between 10 and 12 then "lead data scientist"

when exp between 12 and 16 then "manager"

end as *"role1"*

from fullstack.data\_science\_team )*x*

where role = *role1*;

**A screenshot of a computer

AI-generated content may be incorrect.**

**[Comment: all employee in the data science team matches the organization’s set standard]**

**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 ,`role` ,dept ,salary ,`exp` ,emp\_rating ,(salary \* 0.05)\* emp\_rating as *bonus*

from fullstack.emp\_record\_table ;

A screenshot of a computer

AI-generated content may be incorrect.

**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,round( avg(salary),2) as *avg\_salary*

from fullstack.emp\_record\_table

group by continent ,country

order by avg(salary)desc ;

A screenshot of a computer

AI-generated content may be incorrect.