```
-- Task 1

Create Database Employee;
select * from data_science_team;
select * from emp_record_table;
select * from proj_table;

-- Task 2

# ER Diagram

-- Task 3

select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT from emp_record_table order by
DEPT;

 -- TASK 4

select EMP_ID, FIRST_NAME, LAST_NAME, GENDER, DEPT, EMP_RATING,
        case
                when emp_rating < 2 then 'less than two'
        when emp_rating <= 4 then 'between two and four'
        else 'greater than four'
        end as Rating_Status
    from emp_record_table;

-- Task 5

select concat(FIRST_NAME, " ", LAST_NAME) as NAME from emp_record_table where DEPT
= 'Finance';

/* select E.EMP_ID, E.First_Name, M.First_Name Manager_Name, E.Manager_ID
from emp_record_table E join emp_record_table M on E.MANAGER_ID = M.EMP_ID; */

-- Task 6

select M.First_Name, count(E.Manager_ID) as No_Of_Emps, M.FIRST_NAME Manager_Name
from emp_record_table E
join emp_record_table M on E.MANAGER_ID = M.EMP_ID group by M.First_Name;


-- Task 7

select * from emp_record_table where Dept = 'Healthcare'
UNION
select * from emp_record_table where Dept = 'Finance';

-- Task 8

select EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, EMP_RATING, max(EMP_RATING) over
```

```
(partition by DEPT) as Max_RatingByDept
from emp_record_table;

/* select EMP_ID, FIRST_NAME, LAST_NAME, ROLE, DEPT, SALARY,
min(SALARY) over (partition by Role) as Min_Salary,
max(SALARY) over (partition by Role) as Max_Salary from emp_record_table; */

-- Task 9

select Role, min(salary), max(salary) from emp_record_table group by role;

-- Task 10

select * , Rank() over (order by EXP desc) EXP_wise_Rank from emp_record_table;

-- Task 11

create view Country_Wise_Salary
as
select EMP_ID, FIRST_NAME, COUNTRY, SALARY from emp_record_table where salary >
6000 order by COUNTRY;
select * from Country_Wise_Salary;

-- Task 12

select * from emp_record_table where EMP_ID in (
select EMP_ID from emp_record_table where Exp > 10);  -- Task 12

-- Task 13

delimiter //
CREATE procedure EmpWith_3plusExp()
Begin
        select * from emp_record_table where Exp > 3;
end //

call Emp_With_3plus_Exp;

-- Task 14

Delimiter $$
USE `employee`$$
CREATE FUNCTION `Task14`(eid   varchar(5))
RETURNS varchar(100)
    DETERMINISTIC
BEGIN
        declare ex int;
    declare r varchar(80);
    declare vrole varchar(100);
    declare flag varchar(10);
    select exp, ROLE into ex, VROLE from data_science_team where emp_ID = eid;

                if ex > 12 and ex < 16 then
                        if VROLE = 'Manager' then
                                set flag = 'Yes';
                        else
                                set flag = 'No';
                        end if;
                        # set r = 'Manager';
                elseif ex > 10 and ex <= 12 then
                        if VROLE = 'LEAD DATA SCIENTIST' then
                                set flag = 'Yes';
                        else
                                set flag = 'No';
                        end if;
                        #set r = 'LEAD DATA SCIENTIST';
                elseif ex > 5 and ex <=10 then
                        if VROLE = 'SENIOR DATA SCIENTIST' then
                                set flag = 'Yes';
                        else
                                set flag = 'No';
                        end if;
                        #set r ='SENIOR DATA SCIENTIST';
                elseif ex > 2 and ex <=5 then
                        if VROLE = 'ASSOCIATE DATA SCIENTIST' then
                                set flag = 'Yes';
                        else
                                set flag = 'No';
                        end if;
                        #set r = 'ASSOCIATE DATA SCIENTIST';
                elseif ex <= 2 then
                        if VROLE = 'JUNIOR DATA SCIENTIST' then
                                set flag = 'Yes';
                        else
                                set flag = 'No';
                        end if;
                        #set r = 'JUNIOR DATA SCIENTIST';
                end if;


RETURN flag;
END$$
DELIMITER ;
;

SELECT *,Task14(Emp_ID) FROM data_science_team;

-- Task 15

select * from emp_record_table where FIRST_NAME = 'Eric';
```

```
create index Idx_Emp_FName on emp_record_table(First_Name);
select * from emp_record_table where First_Name = 'Eric';

-- Task 16

select *, (salary * .05) * Emp_Rating Bonus from emp_record_table;

-- Task 17

select Emp_ID, First_Name, Country, Continent, Salary,
        avg(salary) over (partition by Continent, Country) Avg_Sal_ContinentCountry
from emp_record_table;
```