```
create database employee;
use employee;

create table employee_record_table(
  emp_id
varchar(25) not null,
  first_name varchar(45) not null,
  last_name varchar(45) not null,

gender varchar(25) not null,
  role_ varchar(25) not null,
  dept varchar(45) not null,
  exp_
int not null,
  country varchar(45) not null,
  continent varchar(45) not null,
  salary int
not null,
  emp_rating int not null,
  manager_id varchar(25) null,
  proj_id varchar(25)
null,
  primary key (emp_id)
);

create table Proj_table(
  project_id varchar(25) not null,

proj_name varchar(45) not null,
  domain varchar(25) not null,
  start_date date not null,

closure_date date not null,
  dev_qtr varchar(25) not null,
  status_ varchar(25) not null,

primary key (project_id)
);

create table data_science_team(
  emp_id varchar(25) not null,

first_name varchar(45) not null,
  last_name varchar(45) not null,
  gender varchar(25) not
null,
  role_ varchar(45) not null,
  dept varchar(45) not null,
  exp_ int not null,
  country
varchar(45) not null,
  continent varchar(45) not null,
  primary key (emp_id)
);

select
emp_id,first_name,last_name,gender,dept
from employee_record_table;

select
emp_id,first_name,last_name,gender,dept,emp_rating from employee_record_table
where
emp_rating<2;

select emp_id,first_name,last_name,gender,dept,emp_rating from
employee_record_table
where emp_rating>4;

select
emp_id,first_name,last_name,gender,dept,emp_rating from employee_record_table
where emp_rating
between 2 and 4;

select concat(first_name,last_name) as Name_ from employee_record_table
```

```
where
dept= 'finance';

SELECT m.emp_id,m.first_name,m.last_name,m.role_,
m.exp_,COUNT(e.emp_id) as
"EMP_COUNT"
FROM employee_record_table m
INNER JOIN employee_record_table e
ON
m.emp_id = e.manager_id
GROUP BY m.emp_id
ORDER BY m.emp_id;

select
emp_id,first_name,last_name,dept
from employee_record_table
where
dept='healthcare'
union
select emp_id,first_name,last_name,dept
from
employee_record_table
where dept='finance';

select
emp_id,first_name,last_name,role_,dept,emp_rating,max(emp_rating)
over(partition by dept)
as
"max_dept_rating"
from employee_record_table;

SELECT
emp_id,first_name,last_name,role_, MIN(salary) AS min_salary, MAX(salary) AS max_salary
FROM
employee_record_table
where role_ in("president","lead data
scientist","senior
data scientist","manager","associate data
scientist"," juniour data scientist")
group by role_ ;

select
emp_id,first_name,last_name,exp_,
rank() over(order by exp_) as exp_rank
from
employee_record_table;

create view employees_in_various_countries as
select
emp_id,first_name,last_name,country,salary
from employee_record_table
where
salary>6000;

select * from employees_in_various_countries;

select
emp_id,first_name,last_name,exp_ from employee_record_table
where emp_id in(select manager_id
from employee_record_table);

DELIMITER &&
CREATE PROCEDURE
get_experience_details()
BEGIN
SELECT emp_id,first_name,last_name,exp_ FROM
employee_record_table WHERE EXP_>3;
END &&

CALL
get_experience_details();

DELIMITER &&
CREATE FUNCTION Employee_ROLE(
exp_
```

```
int
)
RETURNS VARCHAR(40)
DETERMINISTIC
BEGIN
DECLARE Employee_ROLE VARCHAR(40);
IF exp_>12
AND 16 THEN
SET Employee_ROLE="MANAGER";
ELSEIF exp_>10 AND 12 THEN
SET
Employee_ROLE ="LEAD DATA SCIENTIST";
ELSEIF exp_>5 AND 10 THEN
SET Employee_ROLE
="SENIOR DATA SCIENTIST";
ELSEIF exp_>2 AND 5 THEN
SET Employee_ROLE
="ASSOCIATE DATA SCIENTIST";
ELSEIF exp_<=2 THEN
SET Employee_ROLE ="JUNIOR
DATA SCIENTIST";
END IF;
RETURN (Employee_ROLE);
END &&

SELECT
exp_,Employee_ROLE(exp_)
FROM data_science_team;

CREATE INDEX idx_first_name
ON
employee_record_table(first_name(20));
SELECT * FROM employee_record_table
WHERE
first_name='Eric';

SELECT first_name,last_name,salary,emp_rating, (salary * emp_rating * 0.05)
AS bonus
FROM employee_record_table;

SELECT
emp_id,first_name,last_name,salary,country,continent,
AVG(salary)OVER(PARTITION BY
country)AVG_salary_IN_COUNTRY,
AVG(salary)OVER(PARTITION BY continent)AVG_salary_IN_CONTINENT,

COUNT(*)OVER(PARTITION BY country)COUNT_IN_COUNTRY,
COUNT(*)OVER(PARTITION BY
continent)COUNT_IN_CONTINENT
FROM employee_record_table;
```