```
create database employee;
use employee;

SELECT EMP_ID  ,           #QUERY-3
FIRST_NAME ,LAST_NAME ,
GENDER,
DEPT FROM EMP_RECORD_TABLE;


SELECT EMP_ID ,            #QUERY-4(1)
FIRST_NAME,LAST_NAME ,
GENDER,
DEPT,
EMP_RATING FROM EMP_RECORD_TABLE
WHERE EMP_RATING="2"


SELECT EMP_ID ,              #QUERY-4(2)
FIRST_NAME,LAST_NAME ,
GENDER,
DEPT,
EMP_RATING FROM EMP_RECORD_TABLE
WHERE EMP_RATING>"4"


SELECT EMP_ID ,                  #QUERY-4(3)
FIRST_NAME,LAST_NAME ,
GENDER,
DEPT,
EMP_RATING FROM EMP_RECORD_TABLE
WHERE EMP_RATING BETWEEN "2"AND"4"


SELECT FIRST_NAME,LAST_NAME ,CONCAT(FIRST_NAME," ",LAST_NAME)         #QUERY-5
AS "NAME"
FROM EMP_RECORD_TABLE
WHERE DEPT="FINANCE";

select                                                              #QUERY-6(1)
T2.FIRST_NAME AS "EMPLOYEE",
 T1.FIRST_NAME AS "MANAGER"
 FROM EMP_RECORD_TABLE T1 RIGHT JOIN EMP_RECORD_TABLE T2
 ON T1.EMP_ID=T2.MANAGER_ID;

select                                                               #QUERY-
6(2)
 T2.FIRST_NAME AS "EMPLOYEE",
 T1.FIRST_NAME AS "MANAGER"
 FROM EMP_RECORD_TABLE T1 RIGHT JOIN EMP_RECORD_TABLE T2
 ON T1.EMP_ID=T2.MANAGER_ID
 group by MANAGER;


 select                                                 #query-7
 first_name,
 last_name,
 dept
 from emp_record_table
 where dept in ("finance","healthcare");


 select                                                #query-8(1)
 emp_id,
 first_name,
```

```
 last_name,
 role,
 dept,
 emp_rating
 from emp_record_table;


 select                                                  #query-8(2)
 dept,
 max(emp_rating) as "max_rating"
 from emp_record_table
 group by dept;


select                                                  #query-9(1)
role,
min(salary) as "min_salary"
from emp_record_table
group by role;


select                                                      #query-9(2)
role,
max(salary) as "max_salary"
from emp_record_table
group by role;



select                                                  # query-10
row_number() over() as "rownumber",
first_name,last_name,
exp,
dense_rank()over(order by exp desc)as "dense rank"
from emp_record_table;


create view emp_sa as                                   #query-11
select country,
first_name,last_name,
salary
from emp_record_table
where salary>6000;


select                                                  #query-12
first_name,last_name,
exp
from emp_record_table
where exp>(select avg(exp) from emp_record_table);

#it can come out as exp>10 without nested if like

select                                                  #query-12(a)
first_name,last_name,
exp
from emp_record_table
where exp>10;



delimiter $$                                           #QUERY-13
create procedure getemployeebyexp(in exp int)
```

```
begin

select
 first_name,last_name,
exp
from emp_record_table
where exp>3


end $$
delimiter;

CALL getemployeebyexp;




select
#query-14
first_name,last_name,role ,exp,
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'
else 'manager'
end as role_verify

from data_science_team;




create index simplilearn on emp_record_table(first_name);
#query-15
explain select* from emp_record_table where first_name="eric";



#query-16
delimiter //

create procedure calculatebonus()
begin
update emp_record_table
set bonus=5/100*salary*rating;
end //
call calculatebonus()



select
#query-17
country,
continent,
avg(salary)as avg_salary from emp_record_table
group by country,continent;
```