--SOURCE CODE

--Task-1

CREATE DATABASE employee;

--Task-2

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";

-- Task-6

select m.First\_Name, count(e.Manager\_ID) as No\_Of\_Emps

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, GENDER,DEPT,Emp\_Rating,

max(emp\_Rating) over(partition by DEPT) as Max\_RatingByDept

from emp\_record\_table;

-- Task - 9

select EMP\_ID, FIRST\_NAME, LAST\_NAME, GENDER,DEPT,ROLE,Salary,

MIN(Salary) over(partition by ROLE) as Mim\_Salary,

max(Salary) over(partition by ROLE) as Max\_Salary

from emp\_record\_table;

select ROLE, Min(Salary),Max(Salary) from emp\_record\_table

group by Role;

-- Task - 10

select \*,Rank() over(order by Exp desc) from emp\_record\_table;

-- Task - 11

CREATE VIEW employees\_in\_various\_countries AS

SELECT EMP\_ID,FIRST\_NAME,LAST\_NAME,COUNTRY,SALARY

FROM emp\_record\_table

WHERE SALARY>6000

order by Country;

-- Task-12

select \* from emp\_record\_table where Emp\_ID in

(select Emp\_ID from emp\_record\_table where Exp > 10);

delimiter //

CREATE procedure EmpWith\_3plusExp()

Begin

select \* from emp\_record\_table where Exp > 3;

end //

Delimiter;

call EmpWith\_3plusExp;

-- 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,salary,CONTINENT,Country,

AVg(SALARY) over( partition by CONTINENT,Country ) from emp\_record\_table;