**Que – 1. Problem Statement:**

Use CTEs and display the **12th** highest salary from the employee’s table.

* Return the column '**salary**'.

**Ans – 1.**

with salary\_rank\_table as

(select employee\_id,

salary,

dense\_rank()over(order by salary desc) salary\_rank

from employees)

select salary from salary\_rank\_table

where salary\_rank=12

**Que – 2. Problem Statement:**

Display the employee information who works in **department 80** or has a **salary** of more than **10000**usingthe union operator.

* Return **all** the columns from the employee's table.
* Use the employees table.

**Ans – 2.**

select \* from employees

where department\_id=80

union

select \* from employees

where salary>10000

**Que – 3. Problem Statement:**

Create a view as ‘**emp\_view**’ that has the details i.e, employee\_id, first\_name, last\_name, salary, department\_id, department\_name, location\_id, street\_address, and city.

Display the details from the view of those employees who work in departments that are located in **Seattle** or **Southlake.**

* Return the view with columns '**employee\_id**', '**first\_name**', '**last\_name**', '**salary**', '**department\_id**', '**department\_name**', '**location\_id**', '**street\_address**', '**city**'.

**Ans – 3.**

create view emp\_view as

(select e.employee\_id,

e.first\_name,

e.last\_name,

e.salary,

e.department\_id,

d.department\_name,

d.location\_id,

l.street\_address,

l.city

from employees as e

left join departments as d

on e.department\_id = d.department\_id

left join locations as l

on d.location\_id = l.location\_id)

select \* from emp\_view

where emp\_view.city = "seattle" or emp\_view.city ="southlake"

**Que – 4. Problem Statement:**

Calculate the net salary for the employees and save the column as ‘**Net\_salary**’ and display the details of those employees whose net salary is greater than **15000**.

Use the CTE method.

**Note:** To calculate the ‘Net\_salary’ = salary + salary \*(comission\_pct).

If the column ‘comission\_pct’ consists of null values replace them with zeros using the ifnull() function.

**For example**: ifnull(comission\_pct,0).

* Return the columns '**employee\_id**', '**first\_name**', '**last\_name**', '**salary**', and '**Net\_Salary**'.
* Use the employees table.
* Select MySQL 8.0 in the drop-down.

**Ans – 4.**

with net\_salary\_cte as

select employee\_id,

first\_name,

last\_name,

salary,

case when commission\_pct is null

then salary

else salary+(salary\*commission\_pct)

end as Net\_Salary

from employees

select \* from net\_salary\_cte

where Net\_salary>15000

**Que – 5. Problem Statement:**

Extract all the id’s of the managers(only distinct values) from both the “Employees” and the “Departments” table.

* Return the column '**manager\_id**'.

**Ans – 5.**

select distinct manager\_id from employees

union

select distinct manager\_id from departments

**Que – 6. Problem Statement:**

Create a view as ‘**Manager\_details**’ that has manager details i.e, employee id, manager’s name(first name and last name separated by space) as ‘Manager’, salary, phone\_number, department\_id, department\_name, street\_address, city, country\_name.

Display the details of the **Top 5** managers with the highest salary from the view ‘Manager\_details’.

* Return the columns **employee\_id**, **Manager**, **salary**, and **department\_name**from the view**Manager\_details**.
* The result table must be ordered by salary column in **descending** manner.
* No duplication of manager details is expected in the output.

**Ans – 6.**

create view manager\_details as

(select distinct e.employee\_id,

concat(e.first\_name," ",e.last\_name) as Manager,

e.salary,

e.phone\_number,

d.department\_id,

d.department\_name,

l.street\_address,

l.city,

c.country\_name,

dense\_rank()over(order by e.salary desc)as salary\_rank

from employees as e

left join departments as d

on e.department\_id = d.department\_id

left join locations as l

on d.location\_id = l.location\_id

left join countries as c

on l.country\_id = c.country\_id

where e.employee\_id in

(select distinct manager\_id from employees

union

select distinct manager\_id from departments))

select employee\_id,

Manager,

salary,

department\_name

from manager\_details

where salary\_rank <= 5

order by salary desc

**Que – 7. Problem Statement:**

Write a query to add **5000** for every employee’s salary and save the column as ‘Net\_salary’ and display the details of those employees whose **Net\_Salary** is greater than **20000**.

* Use the CTE method.
* Return the columns '**employee\_id**', '**first\_name**', '**last\_name**', '**salary**', '**Net\_Salary**'.
* Use the employees table.

**Ans – 7.**

with net\_salary\_cte as

(select employee\_id,

first\_name,

last\_name,

salary,

(salary+5000) as Net\_salary

from employees)

select \* from net\_salary\_cte

where Net\_salary >= 20000

**Que – 8. Problem Statement:**

Extract the details of the employees who work under the **same manager**.

Return the details along with the manager’s full name (first name, last name separated by space) as ‘Manager’ and the employee’s full name (first name, last name separated by space) as ‘Employee’.

Order the output based on manager\_id and ‘Employee’ in ascending order.

* Return the columns '**manager\_id**', '**Manager**', '**Employee**'.
* Use the employees table.

**Ans – 8.**

select t.manager\_id,

t.Manager,

t.Employee

from

(select

e.manager\_id,

m.Manager,

concat(e.first\_name," ",e.last\_name)as Employee,

count(m.Manager)over(partition by m.Manager)as no\_of\_employee

from employees as e

left join Manager\_details as m

on e.manager\_id=m.employee\_id

order by e.manager\_id,concat(e.first\_name," ",e.last\_name))as t

where t.no\_of\_employee > 1