**My Solution to the Sql 50 challenge:**

Link to the challenge: <https://leetcode.com/studyplan/top-sql-50/>

Question 1:

select product\_id

from Products

where recyclable = 'Y' and low\_fats = 'Y';

Question 2:

select name

from Customer

where referee\_id != 2 or referee\_id is null;

Question 3:

select name, population, area

from World

where area >= 3000000 or population >= 25000000;

Question 4:

select distinct(author\_id) as id

from Views

where author\_id = viewer\_id

order by id;

Question 5:

select tweet\_id

from Tweets

where length(content) > 15;

Question 6:

select t2.unique\_id, t1.name

from Employees as t1

left join EmployeeUNI as t2

on t1.id = t2.id;

Question 7:

select t2.product\_name, t1.year, t1.price

from Sales as t1

inner join Product as t2

on t1.product\_id = t2.product\_id;

Question 8:

# worse performance

select customer\_id,count(visit\_id) as count\_no\_trans

from Visits

where visit\_id not in (select visit\_id from Transactions)

group by customer\_id

# better performance

select t1.customer\_id, count(t1.visit\_id) as count\_no\_trans

from Visits t1

left join Transactions t2 on t1.visit\_id = t2.visit\_id

where t2.visit\_id IS NULL

group by t1.customer\_id;

Question 9:

Option a:

select t1.id

from Weather t1

inner join Weather t2 on t1.recordDate = date\_add(t2.recordDate, interval 1 day)

where t1.temperature > t2.temperature;

Option b:

select t1.id

from Weather t1

inner join Weather t2 on datediff(t1.recordDate,t2.recordDate) =1

where t1.temperature > t2.temperature ;

Question 10:

select t1.machine\_id,round(avg(t1.timestamp - t2.timestamp),3) as processing\_time

from (select \* from Activity where activity\_type = 'end') as t1

inner join (select \* from Activity where activity\_type = 'start') as t2

on t1.machine\_id = t2.machine\_id

group by t1.machine\_id

Question 11:

select t1.name,t2.bonus

from Employee as t1

left join Bonus as t2

on t1.empId = t2.empId

where t2.bonus < 1000 or t2.bonus is null;

Question 12:

select t1.student\_id,t1.student\_name,t2.subject\_name,count(t3.student\_id) as attended\_exams

from Students as t1

cross join Subjects as t2

left join Examinations as t3

on t1.student\_id = t3.student\_id and t2.subject\_name = t3.subject\_name

group by t1.student\_id,t2.subject\_name

order by t1.student\_id,t2.subject\_name;

Question 13:

select name from (

select t1.id,t1.name

from Employee as t1

inner join Employee as t2

on t1.id = t2.managerId

group by t1.id,t1.name

having count(t2.managerId) >= 5) as sub\_1;

Question 14:

select t1.user\_id,

round(avg(case when t2.action = 'confirmed' then 1 else 0 end),2) confirmation\_rate

from Signups as t1

left join Confirmations as t2

on t1.user\_id = t2.user\_id

group by t1.user\_id;

Question 15:

select \*

from Cinema

where id %2 != 0 and description != 'boring'

order by rating desc;

Question 16:

select t1.product\_id, ifnull(round(sum(t1.price\*t2.units)/sum(units),2),0) as average\_price

from Prices as t1

left join UnitsSold as t2

on t1.product\_id = t2.product\_id

and t2.purchase\_date between t1.start\_date and t1.end\_date

group by t1.product\_id

Question 17:

select t1.project\_id,round(avg(t2.experience\_years),2) as average\_years

from Project as t1

inner join Employee as t2

on t1.employee\_id = t2.employee\_id

group by t1.project\_id

Question 18:

select t2.contest\_id,round(count(distinct(t1.user\_id))\*100/(select count(user\_id) from Users),2)

as percentage

from Users as t1

inner join Register as t2

on t1.user\_id = t2.user\_id

group by t2.contest\_id

order by percentage desc, contest\_id;

Question 19:

select query\_name,round(avg(rating/position),2) as quality,

round(avg(case when rating < 3 then 1 else 0 end)\*100,2) as poor\_query\_percentage

from Queries

where query\_name is not null

group by query\_name

Question 20:

select

date\_format(trans\_date,"%Y-%m") as month,

country,

count(trans\_date) as trans\_count,

sum(case when state  = 'approved' then 1 else 0 end) as approved\_count,

sum(amount) trans\_total\_amount,

sum(case when state  = 'approved' then amount else 0 end) as approved\_total\_amount

from Transactions

group by month,country

Question 21:

with first\_orders as(

select \* from (

select \*,dense\_rank() over(partition by customer\_id order by order\_date) as ranking

from Delivery) as sub\_1

where ranking = 1)

select round(count(customer\_id)\* 100/(select count(customer\_id) from first\_orders),2) as immediate\_percentage

from first\_orders

where order\_date = customer\_pref\_delivery\_date

Question 22:

with first\_log as (

select \* from (

select \*, dense\_rank() over(partition by player\_id order by event\_date) as ranking

from Activity) as sub\_1

where ranking <= 2)

select round(count(distinct t1.player\_id)/(select count(distinct player\_id) from activity),2) as fraction

from first\_log as t1

inner join first\_log as t2

on t1.player\_id = t2.player\_id

where t1.event\_date = date\_add(t2.event\_date,interval 1 day)

Question 23:

select teacher\_id,count(distinct subject\_id) as cnt

from Teacher

group by teacher\_id

Question 24:

select activity\_date as day,count(distinct(user\_id)) active\_users

from activity

where datediff(activity\_date,'2019-07-27') <= 29

group by activity\_date

Question 25:

select product\_id,year as first\_year ,quantity,price from (

select product\_id,year,quantity,price,

dense\_rank() over(partition by product\_id order by year) as ranking

from Sales) as sub\_1

where ranking = 1;

Question 26:

select class

from Courses

group by class

having count(student) >= 5

Question 27:

select user\_id,count(follower\_id) as followers\_count

from Followers

group by user\_id

order by user\_id

Question 28:

select max(num) as num from (

select num

from MyNumbers

group by num

having count(num) = 1) as sub\_1

Question 29:

select customer\_id

from Customer

group by customer\_id

having count(distinct product\_key) in (select count(product\_key) from Product)

Question 30:

select t1.employee\_id,t1.name,count(t2.reports\_to) as reports\_count,

round(avg(t2.age)) as average\_age

from Employees as t1

inner join Employees as t2

on t1.employee\_id = t2.reports\_to

group by employee\_id

order by employee\_id

Question 31:

select employee\_id,department\_id

from Employee

where primary\_flag = 'Y'

union

select employee\_id,department\_id

from Employee

group by employee\_id

having count(employee\_id) = 1;

Question 32:

select \*,

case when x + y > z and x + z > y and y + z > x then 'Yes' else 'No'

end as triangle

from Triangle;

Question 33:

Select distinct num AS ConsecutiveNums

from(

    select

        num,

        lead (num, 1) over (order by id) as num\_1,

        lead(num, 2) over (order by id) as num\_2,

        id,

        lead(id, 1) over (order by id) as id\_1,

        lead(id, 2) over (order by id) as id\_2

    from Logs

) as sub\_1

where num = num\_1 and num\_1 = num\_2 and id\_1 = id + 1 and id\_2 = id\_1 + 1;

Question 34:

with cte as (

select product\_id,price from (

select product\_id,new\_price as price,

dense\_rank() over(partition by product\_id order by change\_date desc) as ranking

from Products

where change\_date <= '2019-08-16') as sub\_1

where ranking = 1)

select \* from cte

union

select product\_id,10 as price

from Products

where product\_id not in (select product\_id from cte)

Question 35:

select person\_name from (

select person\_name,

sum(weight) over(order by turn) as comulative\_sum,

turn

from Queue) as sub\_1

where comulative\_sum <= 1000

order by turn desc

limit 1;

Question 36:

select "Low Salary" as category,

ifnull(sum(case when income < 20000 then 1 else 0 end),0) as accounts\_count

from Accounts

union

select "Average Salary" as category,

ifnull(sum(case when income >= 20000 and income <= 50000 then 1 else 0 end),0) as accounts\_count

from Accounts

union

select "High Salary" as category,

ifnull(sum(case when income > 50000 then 1 else 0 end),0) as accounts\_count

from Accounts

Question 37:

select employee\_id

from Employees

where salary < 30000 and manager\_id not in

(select employee\_id from Employees) and manager\_id is not null

order by employee\_id

Question 38:

select case when id % 2 != 1 then id -1

when id = (select max(id) from Seat) then id

else id + 1 end as id,

student

from Seat

order by id;

Question 39:

select name as results from (

select t1.name,count(t2.rating) as amount\_rated

from Users as t1

inner join MovieRating as t2

on t1.user\_id = t2.user\_id

group by t1.name

order by amount\_rated desc,t1.name

limit 1) as sub\_1

union all

select title as results from (

select t1.title,avg(t2.rating) as avg\_rating

from Movies as t1

inner join MovieRating as t2

on t1.movie\_id = t2. movie\_id

where created\_at like '2020-02%'

group by t1.title

order by avg\_rating desc,t1.title asc

limit 1) as sub\_1

Question 40:

Question 41:

select id,sum(counts) num from (

select requester\_id as id ,count(accepter\_id) as counts

from RequestAccepted

group by requester\_id

union all

select accepter\_id as id ,count(requester\_id) as counts

from RequestAccepted

group by accepter\_id) as sub\_1

group by id

order by sum(counts) desc

limit 1

Question 42:

select round(sum(tiv\_2016),2) as tiv\_2016

from (

    select tiv\_2016

    from Insurance

    where tiv\_2015 in (select tiv\_2015 from

    Insurance group by tiv\_2015

    having count(\*) > 1)

    and (lat,lon) in (select lat,lon

    from Insurance group by lat,lon having count(\*) = 1)

) as sub\_1

Question 43:

select Department,Employee,Salary from (

select t2.name as Department,t1.name as Employee,

salary as Salary,

dense\_rank() over(partition by t2.name order by salary desc) as ranking

from Employee as t1

inner join Department as t2

on t1.departmentId = t2.id) as sub\_1

where ranking <= 3;

Question 44:

select user\_id,concat(upper(left(name,1)),lower(substring(name,2))) as name

from Users

order by user\_id

Question 45:

select patient\_id, patient\_name, conditions

from Patients

where conditions like 'DIAB1%' or conditions like '% DIAB1%'

Question 46:

with cte as (select \* from Person)

delete from Person where id not in (select min(id) from cte group by email)

Question 47:

select max(salary) as SecondHighestSalary

from Employee

where salary < (select max(salary) from Employee)

Question 48:

select sell\_date,count(distinct product) as num\_sold,group\_concat(distinct product) as  products

from Activities

group by sell\_date

order by sell\_date

Question 49:

select t1.product\_name,sum(t2.unit) as unit

from Products as t1

inner join Orders as t2

on t1.product\_id = t2.product\_id

where t2.order\_date like '2020-02%'

group by t1.product\_name

having sum(t2.unit) >=  100 ;

Question 50: seems to be less relevant

 SELECT \*

FROM Users

WHERE mail REGEXP '^[A-Za-z][A-Za-z0-9\_\.\-]\*@leetcode(\\?com)?\\.com$';