

SQL Peer Learning Documentation

Q1) Given a table of employees, find the number of male and female employees in each department:

Rohit's Approach:

He Created a procedure named `getNumMaleFemale()` to get the number of males and females.

The procedure has the following properties:

- if Department entry in employees table is null, then added gender details as Not Assigned
- He grouped the table bases on Departments
- Used Case and Then on the following
- incrementing the count by 1 whenever male entry is found
- incrementing the count by 1 whenever female entry is found
- Finally calling the procedure to get number of males and females department wise

Amit's Approach

He did this question directly instead of using procedure. The approach is similar to Rohit's approach. But the way of writing the query is very simple and understandable.

Q2) Given a table with salaries of employees for different month, find the max amount from the rows with month name:

Rohit's Approach:

He Created a procedure named `maxAmount ()` to get the desired output
The procedure has the following properties:

- Selects data from a table and finds the maximum value from the columns "Jan", "Feb", and "March" and stores it in the "value" column.
- Determines the month corresponding to the maximum value using the "field" function, which returns the index of a value in a list of values, and stores the result in the monthNum column.
- Uses a "**CASE WHEN**" statement to convert the monthNum column into the month name (Jan, Feb, or Mar) and returns the result in the "Month" column.

Amit's Approach

He has written a query using Case and Then, first selecting the greatest value out of Jan, Feb, and March as Value, and then Month name with the corresponding greatest value as Month.

Q3) Given the marks obtained by candidates in a test, rank them in proper order.

Rohit's Approach

He Created a procedure named `getRank ()` to get the desired output
The procedure has the following properties:

- Has the `DENSE_RANK()` function to determine the rank of a value in a group of values based on the order by marks as descending order in the `OVER` clause.
- concat the ids if students marks is same

Amit's Approach

He has done it without using procedures

- Used `DENSE_RANK()` function to determine the rank of a value in a group of values based on the order by marks as descending order in the `OVER` clause.
- concat the ids if students marks is same

Q4) If same value is repeated for different id, then keep the value that has smallest id and delete all the other rows having same value:

Rohit's Approach:

He created a Procedure named `DelDuplicateExceptSmallestID` () to keep the value that has the smallest id and delete all the other rows having the same value. The Procedure has the following properties-

- Performing inner join on the tables and fetching the candidate id having duplicate entries and greater in value.

- These duplicated ids are then deleted from the table using the **DELETE** clause.

Amit's Approach

He did this question by creating a view of minimum id out of the original table. Later deleted all those records whose ids are not present in the view. Because, they are duplicated.