Yuvraj Singh

Data analyst Intern Assignment

singhraj153989@gmail.com

<https://yuvraj153989.github.io/Portfolio/>

http://linkedin.com/in/yuvraj-singh-3018741b3

+91 8591476799

*TASK 01*

*Your task is to give SQL queries for problem statements with the help of below mentioned tables -*

*Table - LINK*

*A. Retrieve the names and salaries of all employees, along with the average salary in their respective*

*departments.*

*B. Calculate the total sales amount for each employee, including those who have not made any sales.*

*Display their names and total sales amount.*

*C. Rank employees within each department based on their salary in descending order. The ranking should*

*reset for each department.if two employees have the highest salary, they will both be assigned the rank*

*1, and the next distinct salary will be assigned the rank 3 (skipping 2).*

*D. Rank employees within each department based on their salary in descending order. The ranking should*

*reset for each department.if two employees have the highest salary, they will both be assigned the rank*

*1, and the next distinct salary will be assigned the rank 2."*

**Queries**

*use assignment;*

*-- viewing employee table--*

*select \* from employee;*

*-- Viewing sales table--*

*select \* from SALES;*

**-- Question 1: Retrieve the names and salaries of all employees, along with the average salary in their respective departments**.

*SELECT*

*EMP\_ID,*

*Name,*

*Department,*

*Salary,*

*AVG(Salary) OVER (PARTITION BY Department) AS Avg\_Salary\_In\_Department*

*FROM*

*employee;*

**-- Question 2: Calculate the total sales amount for each employee, including those who have not made any sales. Display their names and total sales amount.**

*SELECT*

*employee.Name,*

*IFNULL(SUM(SALES.Sale\_Amount), 0) AS Total\_Sales\_Amount*

*FROM*

*employee*

*LEFT JOIN*

*SALES ON employee.EMP\_ID = SALES.Employee\_ID*

*GROUP BY*

*employee.EMP\_ID, employee.Name;*

**-- Question 3: Rank employees within each department based on their salary in descending order. The ranking should reset for each department.if two employees have the highest salary, they will both be assigned the rank** 1**, and the next distinct salary will be assigned the rank 3 (skipping 2).**

*SELECT*

*EMP\_ID,Name,*

*Department,*

*Salary,*

*RANK() OVER (PARTITION BY Department ORDER BY Salary DESC) AS Salary\_Rank*

*from*

*employee;*

**-- Question 4: Rank employees within each department based on their salary in descending order. The ranking should reset for each department.if two employees have the highest salary, they will both be assigned the rank 1, and the next distinct salary will be assigned the rank 2."**

*SELECT*

*Department,*

*EMP\_ID,*

*Name,*

*Salary,*

*DENSE\_RANK() OVER (PARTITION BY Department ORDER BY Salary DESC) AS Salary\_Rank*

*FROM*

*employee;*

*TASK 02*

*Your task is to give queries for problem statements with the help of below mentioned 3 tables -*

*Table 1 - rm\_master - contains info of the raw material sold by the vendor - LINK*

*Table 2 - rm\_mapping\_master - contains info about the raw material used in SKU - LINK*

*Table 3 - sales\_master - quantity of SKU sold on a given day - LINK*

*A. To get the total quantity of each rm(raw material) sold in each month.*

*B. To get the name of vendors for each SKU.*

*C. Get the most used and least used raw material based on the SKU sold.*

**Queries**

*use openinapp;*

*-- viewing rm\_master table*

*select \* from rm\_master;*

*-- viewing rm\_mapping\_master*

*select \* from rm\_mapping\_master;*

*-- viewing sales\_master*

*select \* from sales\_master;*

-- **Q1. To get the total quantity of each rm(raw material) sold in each month.**

*-- formatting date column*

*UPDATE sales\_master*

*SET date = STR\_TO\_DATE(date, '%d/%m/%Y');*

*select sum(Quantity), month(date) as sale\_month*

*from sales\_master*

*group by sale\_month;*

**-- Q2. To get the name of vendors for each SKU.**

*SELECT*

*rmm.SKU,*

*REPLACE(GROUP\_CONCAT(DISTINCT rm.VendorName ORDER BY rm.VendorName ASC), ',', ', ') AS Vendor\_Names*

*FROM*

*rm\_mapping\_master rmm*

*LEFT JOIN*

*rm\_master rm ON rmm.`RID` = rm.`RID` GROUP BY rmm.SKU;*

**-- Q3: Get the most used and least used raw material based on the SKU sold. Most Used Raw Material**

*SELECT*

*rmm.`RID` AS Raw\_Material\_ID,*

*rm.`RName` AS Raw\_Material\_Name,*

*SUM(rmm.Quantity) AS Total\_Quantity\_Sold*

*FROM*

*rm\_mapping\_master rmm*

*JOIN*

*rm\_master rm ON rmm.`RID` = rm.`RID`*

*GROUP BY*

*rmm.`RID`, rm.`RName`*

*ORDER BY*

*Total\_Quantity\_Sold DESC*

*LIMIT 1;*

**-- Least Used Raw Material**

*SELECT*

*rmm.`RID` AS Raw\_Material\_ID,*

*rm.`RName` AS Raw\_Material\_Name,*

*SUM(rmm.Quantity) AS Total\_Quantity\_Sold*

*FROM*

*rm\_mapping\_master rmm*

*JOIN*

*rm\_master rm ON rmm.`RID` = rm.`RID`*

*GROUP BY*

*rmm.`RID`, rm.`RName`*

*ORDER BY*

*Total\_Quantity\_Sold ASC*

*LIMIT 1;*