**TASK -1 SQL QUERIES**

Internship project at SoulVibe.Tech

Pakkurthi.Naga Poornima Asritha

B.Tech Computer Science Engineering

Intern, SoulVibe.Tech

[poornimaasritha@gmail.com](mailto:poornimaasritha@gmail.com)

## Introduction :

A Dataset ‘Employee’ contains columns like

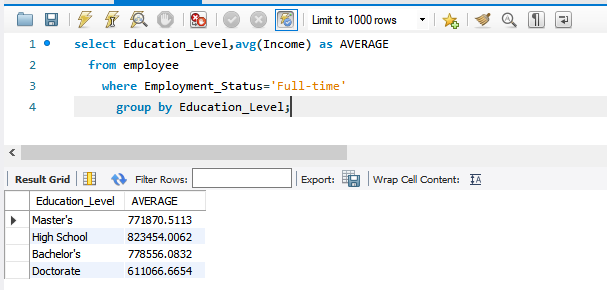
* Age int
* Education\_Level object
* Occupation object
* Number\_of\_Dependents int
* Location object
* Work\_Experience int
* Marital\_Status object
* Employment\_Status object
* Household\_Size int
* Home owner ship\_Status object
* Type\_of\_Housing object
* Gender object
* Primary\_Mode\_of\_Transportation object
* Income int

Describes different details of an employee.The 12 SQL queries are helped to deep dive into analysis of the relationships,meaningful insights in the data.

## Queries:

1. Find the average income for each Education\_Level for those who are employed full-time.

**QUERY & OUTPUT :**

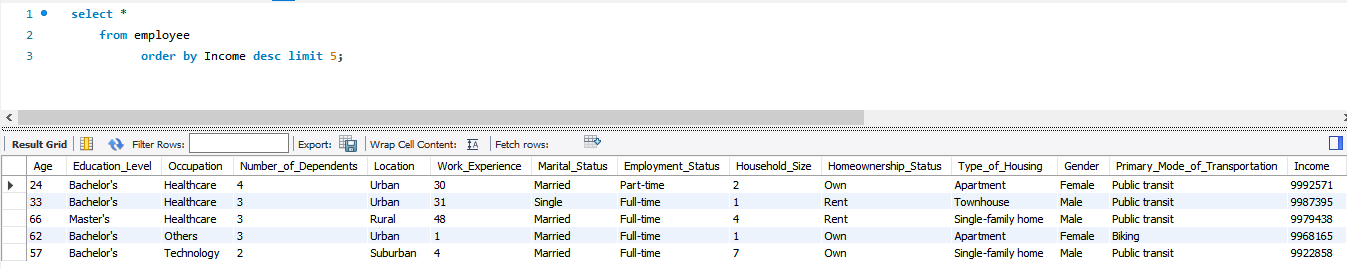


**Description:**

In the dataset ‘employee’, the average income in each education\_level whose employment is full-time is displayed by AVG() aggregation function and SELECT used to display the column as formatted way.GROUP BY is for specifying the education\_level.

1. Retrieve the top 5 highest earning individuals and their details.

**QUERY & OUTPUT :**

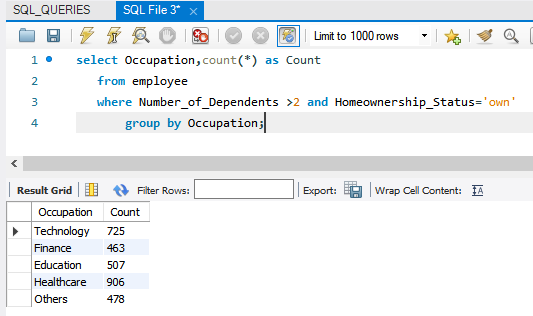


**Description:**

as for top 5, used LIMIT to filter top most earnings and ORDER BY to sort the data as per ‘Income’ and SELECT \* to print the employee details.

1. Count how many people in each Occupation have more than 2 dependents and own a house.

**QUERY & OUTPUT :**



**Description:**

For different occupations of employees who have more than 2 dependents and have own house count are displayed by using COUNT() function, WHERE to specify the conditions ,as it based on occupation so grouped using GROUP BY.

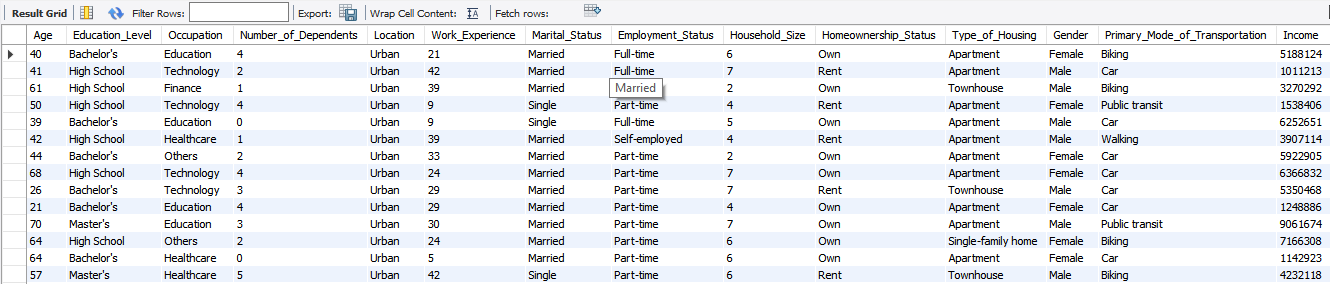
1. List all individuals living in Urban locations with an income above the average income.

**QUERY:**

Select \* from employee

where Location='Urban' and Income > (select avg(Income) from employee);

**OUTPUT:**



**Description:**

Here displaying employees details of urban people with highest earnings above average income using WHERE conditions for filter ’urban’ people and more income than average income using AVG() function in subquery ,the subquery retains the calculated average value of income and compared with selected income then display.

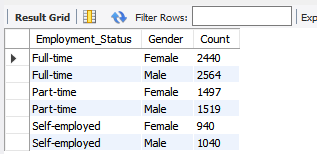
1. Identify how many males and females are in each Employment\_Status.

**QUERY:**

Select Employment\_Status , Gender, count(Gender) as Count

from employee group by Employment\_Status, Gender;

**OUTPUT:**



**Description:**

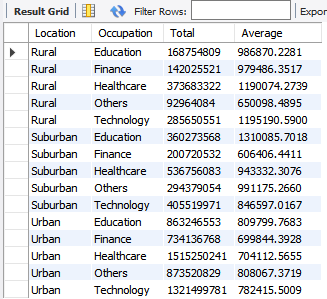
As in each employment\_status the count of males,females are displayed by GROUP BY operation and COUNT( ) function.Firstly grouped based on employment\_status and gender then make their count.

1. What is the total and average income by Location and Occupation?

**QUERY:**

Select Location,Occupation,sum(Income) as Total,avg(Income)asAverage from employee group by Location ,Occupation;

OUTPUT:



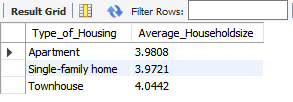
**Description:**  Using aggregation functions like SUM( ) for total Income calculation and AVG( ) for average Income calculation apply on groups based on ‘Location’ and ‘Occupation’ in ‘employee dataset’.

1. Find the average Household\_Size grouped by Type\_of\_Housing.

**QUERY:**

select Type\_of\_Housing , avg(Household\_size) asAverage\_Householdsize from employee group by Type\_of\_Housing;

OUTPUT:



**Description:**

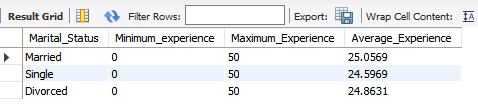
Using aggregation function AVG( ) on ‘Household\_size’ based on ‘Type\_of\_housing’ by GROUP BY gives each housingtype’s average size of house.

1. Calculate the minimum, maximum, and average Work\_Experience for each Marital\_Status.

**QUERY:**

select Marital\_Status, min(Work\_Experience)as Minimum\_experience, max(Work\_Experience) as Maximum\_Experience, avg(Work\_Experience) as Average\_Experience from employee group by Marital\_Status;

OUTPUT:



**Description:**

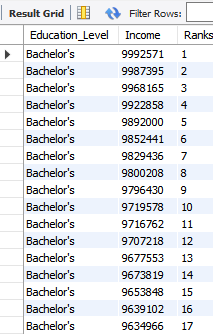
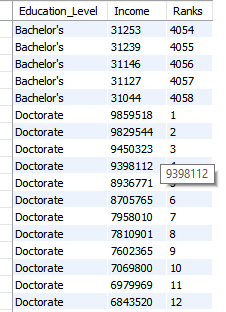
Usage of aggregation functions widely for calculating minimum, maximum ,average work experiences by MIN( ),MAX( ), AVG( ) functions respectively based on group ‘Marital\_status’ are selected to display. For example, minimum experience for married employee is 0, maximum is 50 and an average of 25 yrs.

1. Write a query to rank individuals by Income within each Education\_Level.

**QUERY:** select Education\_Level, Income,

rank() over (partition by Education\_Level order by Income desc) asRanks from employee;

OUTPUT:

**Description:**

With different ‘education \_levels’ like bachelors,masters etc, within each level of ranking the employees using RANK( ) window function followed by OVER ( ), and grouped by PARTITION followed by ORDER BY operations for rank the individuals based on their ‘Income’.

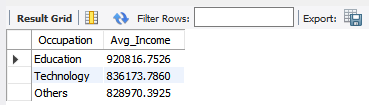
1. Find the top 3 Occupation types with the highest average income.

**QUERY:**

Select Occupation, AVG(Income) AS Avg\_Income from employee

group by Occupation order by Avg\_Income desc limit 3;

OUTPUT:



**Description:**  The highest average income earned occupations are displayed . Firstly calculating the average income based on group

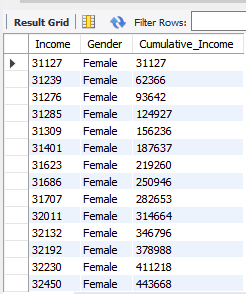
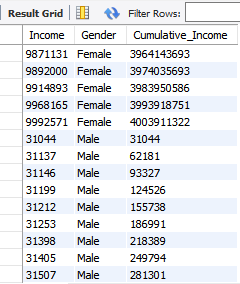
‘occupation’ , sort them using ORDER BY ‘Income’ for first 3 highest income.

1. Use a window function to calculate the cumulative income for each Gender.

**QUERY:**

select Income, Gender, sum(Income) over (partition by Gender order by Income) as Cumulative\_Income from employee;

OUTPUT:

**Description:**

Upon ‘gender’ the cumulative income calculated by WINDOW FUNCTION means performing operation based on sliding window of columns,rows implemented using SUM( ) on OVER ( ) followed by PARTITION ,ORDER BY ‘Income’.

1. List the people whose income is above the median income for the dataset.

**QUERY:**

WITH income\_ranked as (

select Income ,row\_number() over (order by Income) as rn,

count(\*) over () as total\_count from employee),

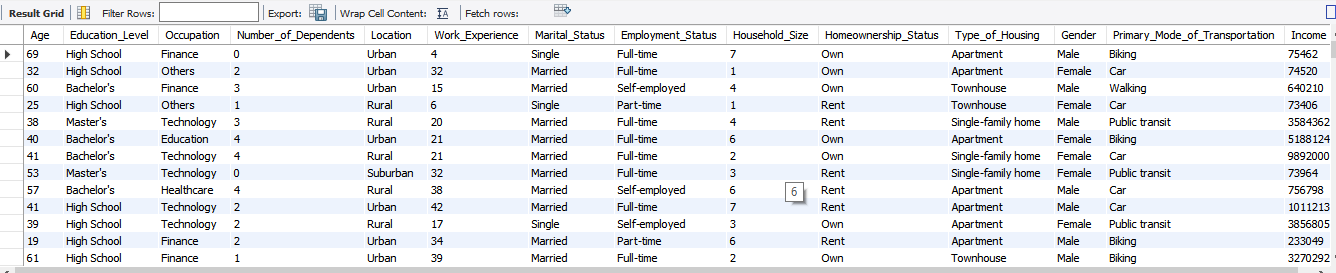
median\_income as (select avg(Income) as median

from income\_ranked

where rn in(floor((total\_count + 1) / 2), floor ((total\_count + 2) / 2)) )

select \* from employee where Income > (select median from median\_income);

OUTPUT:



**Description:**

create a CTE, a temporary table named income\_ranked to calculate the median income from all employees using AVG( ) for count/2 and (count+1)/2 as the dataset contain even number of rows and normal SELECT query using WHERE condition to get highest income above median income.

**Conclusion:**

**T**his project provided a comprehensive exploration of the ‘employee’ dataset using SQL. By formulating and executing a series of analytical queries , I was able to uncover meaningful patterns and insights related to workforce demographics, employment trends, and income distribution. The queries covered a range of operations — from basic filtering and aggregation to more advanced techniques like ranking and window functions. This enabled a deeper understanding of how variables such as education level, occupation, gender, and marital status influence income and work experience. Additionally, the process of cleaning and preparing the data ensured that the results were both accurate and reliable. Identifying duplicate records, handling inconsistencies, and applying conditional logic strengthened the overall quality of the analysis. Through this hands-on SQL experience, I enhanced my ability to translate business questions into structured queries and draw actionable insights from raw data.