SABURI MADUR

Big Data – Hadoop - Hive

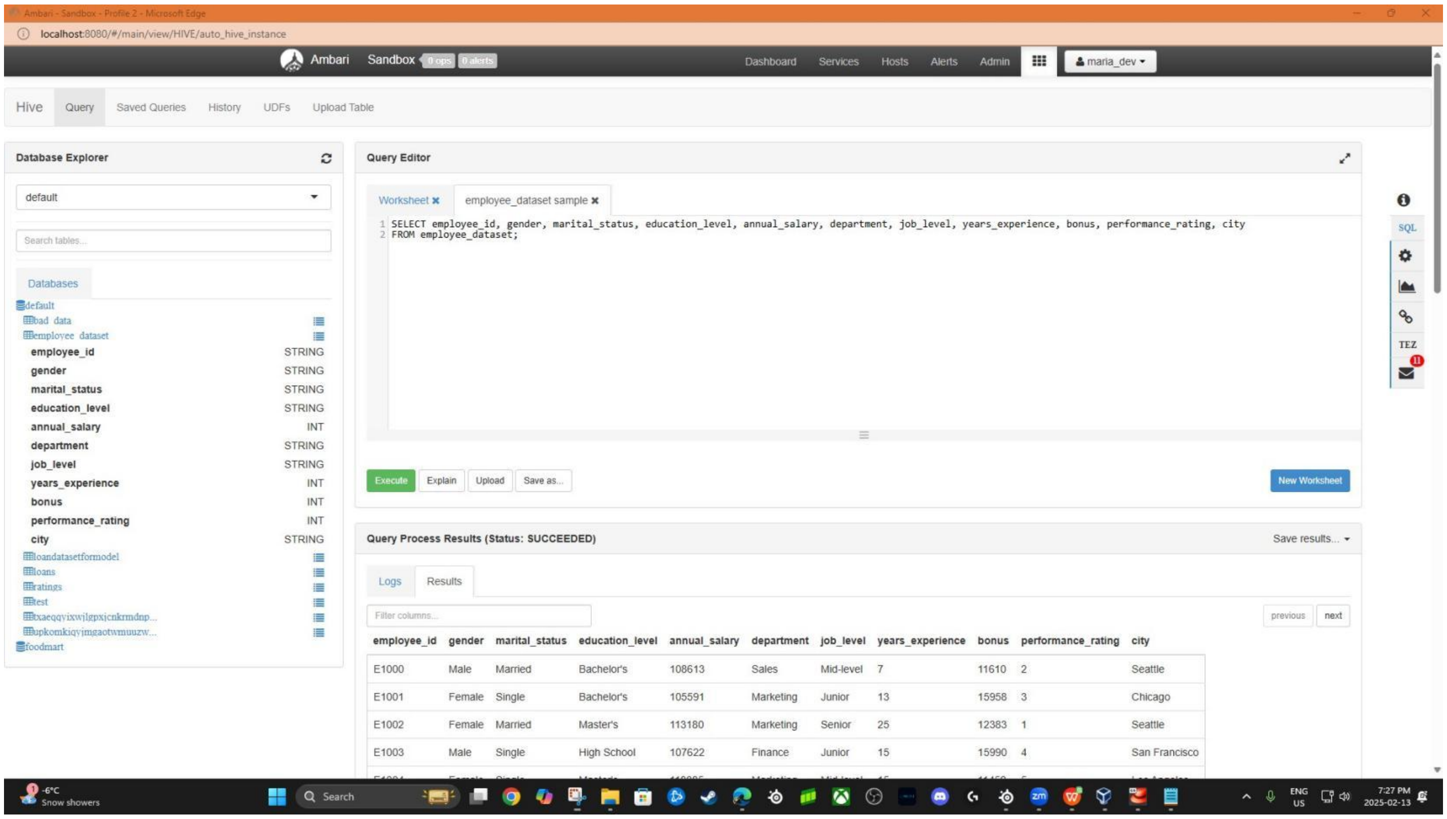
Date: 02/13/2025

Scenario Question:

You are a data analyst working for a company that manages a large workforce. Your manager has asked you to analyze the employee data to gather insights on employee performance, salary distribution, and department-wise headcount. You are given an employee dataset with the following columns: Employee_ID, Gender, Marital_Status, Education_Level, Annual_Salary, Department, Job_Level, Years_Experience, Bonus, Performance_Rating, and City that needs to be uploaded to **Hive** (Hadoop) for further analysis.

Task 1: Select Basic Data

- Your first task is to retrieve a list of all employees along with their key details. This includes their ID, gender, marital status, education level, annual salary, bonus, and performance rating.



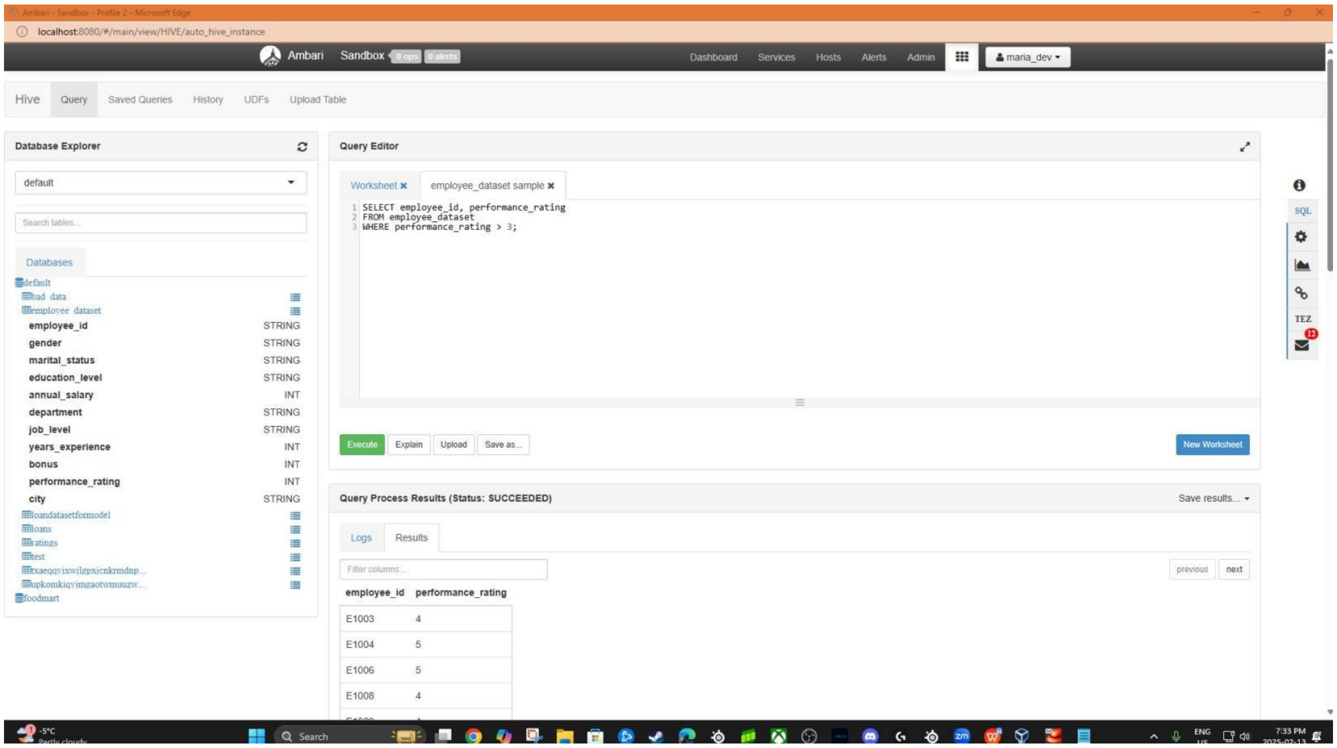
```sql
SELECT employee_id, gender, marital_status, education_level, annual_salary, department, job_level, years_experience, bonus, performance_rating, city
FROM employee_dataset;
```

Query Process Results (Status: SUCCEEDED)

employee_id	gender	marital_status	education_level	annual_salary	department	job_level	years_experience	bonus	performance_rating	city
E1000	Male	Married	Bachelor's	108613	Sales	Mid-level	7	11610	2	Seattle
E1001	Female	Single	Bachelor's	105591	Marketing	Junior	13	15958	3	Chicago
E1002	Female	Married	Master's	113180	Marketing	Senior	25	12383	1	Seattle
E1003	Male	Single	High School	107622	Finance	Junior	15	15990	4	San Francisco

Task 2: Filtering Data

- Next, you need to filter the data to find employees who have a **performance rating of 4 or higher**. This information will help you identify top performers in the organization.



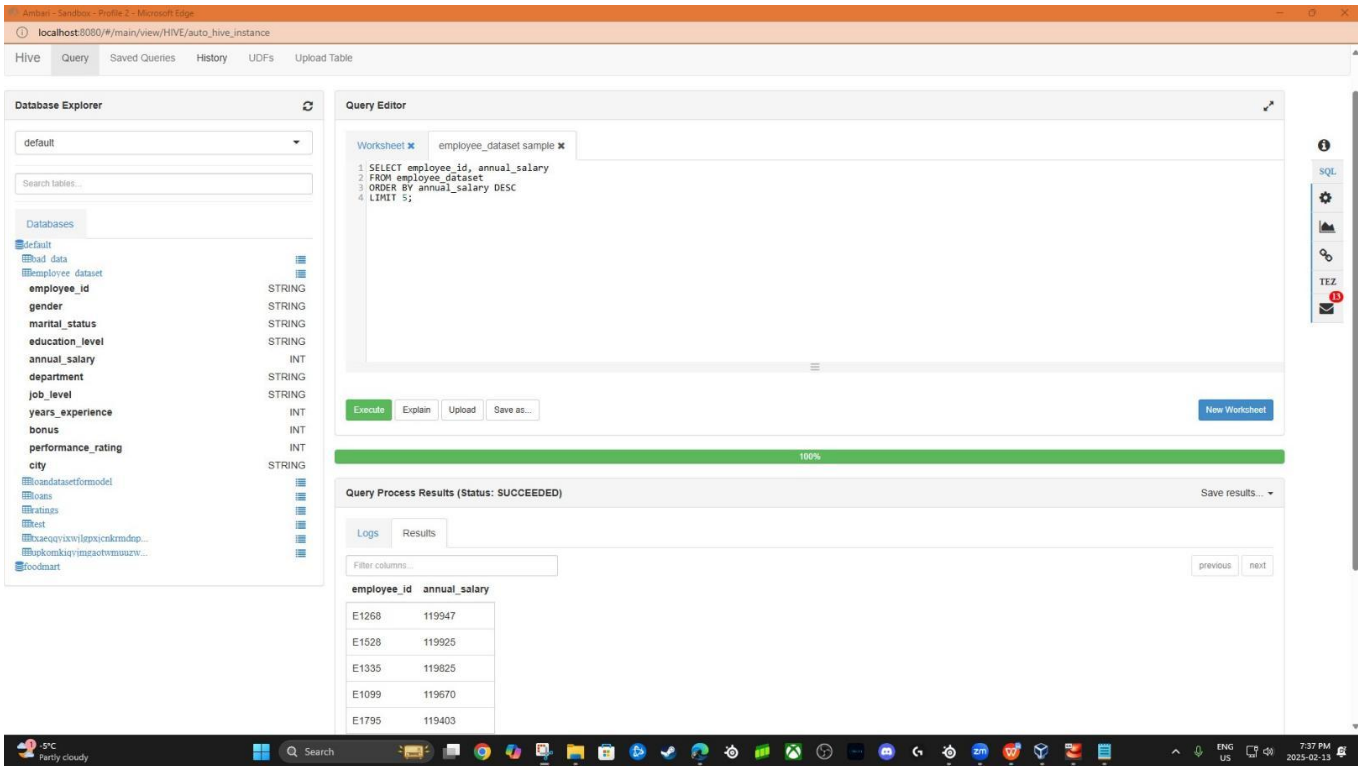
```
SELECT employee_id, performance_rating
FROM employee_dataset
WHERE performance_rating > 3;
```

Query Process Results (Status: SUCCEEDED)

```
employee_id  performance_rating
E1003        4
E1004        5
E1006        5
E1008        4
```

Task 3: Sorting Data

- You are then tasked with identifying the **top 5 highest-paid employees** in the company.
- Sorting the employees by their salary in descending order will give you this information.



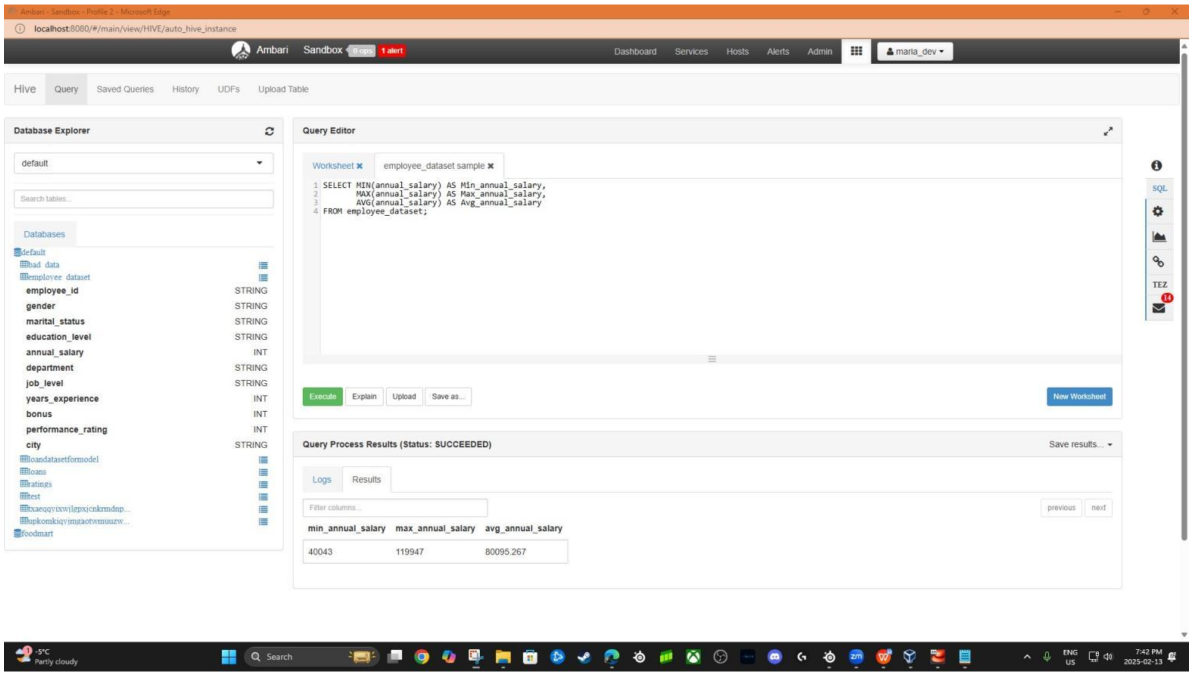
```
SELECT employee_id, annual_salary
FROM employee_dataset
ORDER BY annual_salary DESC
LIMIT 5;
```

Query Process Results (Status: SUCCEEDED)

```
employee_id  annual_salary
E1268        119947
E1528        119925
E1335        119825
E1099        119670
E1795        119403
```

Task 4: Salary Statistics

- Your manager is interested in understanding the distribution of employee salaries. You are required to retrieve the **minimum**, **maximum**, and **average salary** from the dataset to provide a general overview of the company's salary range.



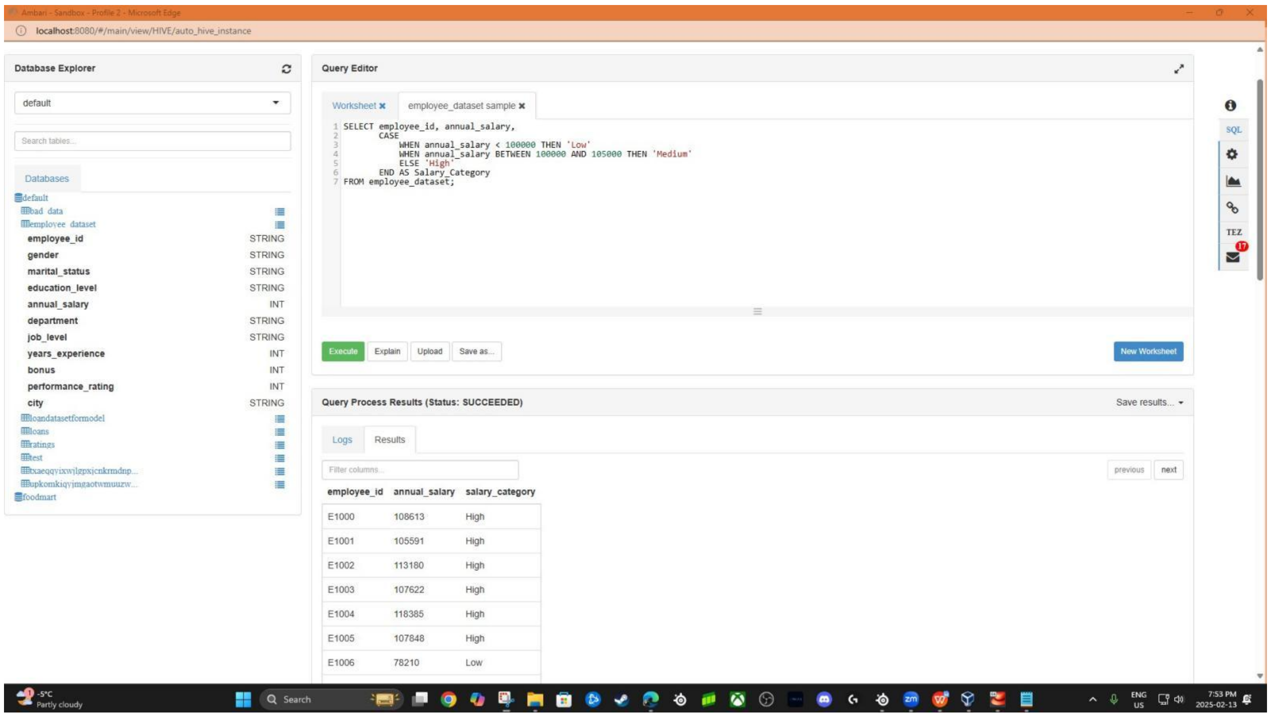
```
SELECT MIN(annual_salary) AS Min_annual_salary,
       MAX(annual_salary) AS Max_annual_salary,
       AVG(annual_salary) AS Avg_annual_salary
FROM employee_dataset;
```

Query Process Results (Status: SUCCEEDED)

min_annual_salary	max_annual_salary	avg_annual_salary
40043	119947	80095.267

Task 5: Salary Category

- To make the data more insightful, you are asked to categorize employees into **"Low,"** **"Medium,"** and **"High"** salary groups based on their annual salary. This categorization will be used in further analysis.



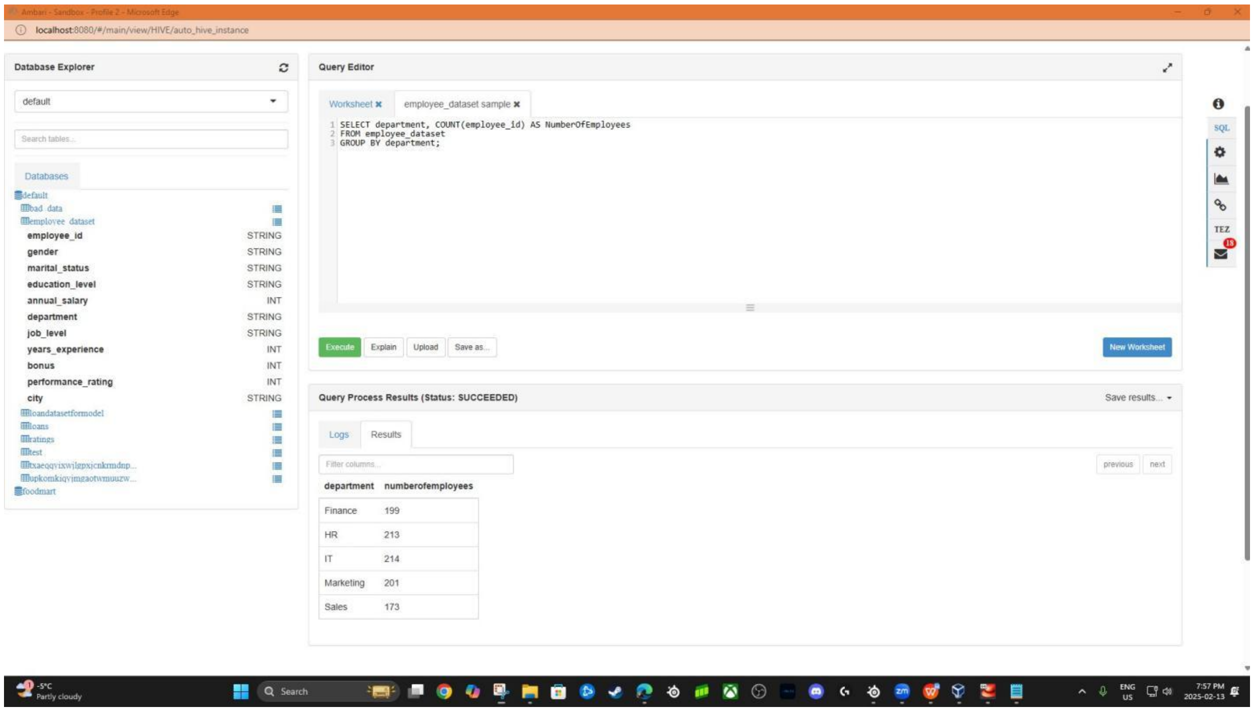
```
SELECT employee_id, annual_salary,
       CASE
           WHEN annual_salary < 100000 THEN 'Low'
           WHEN annual_salary BETWEEN 100000 AND 105000 THEN 'Medium'
           ELSE 'High'
       END AS Salary_Category
FROM employee_dataset;
```

Query Process Results (Status: SUCCEEDED)

employee_id	annual_salary	salary_category
E1000	108613	High
E1001	105591	High
E1002	113180	High
E1003	107622	High
E1004	118385	High
E1005	107848	High
E1006	78210	Low

Task 6: Grouping by Department

- Finally, your manager wants to see how employees are distributed across different **departments**. You will need to group the data by department and count the number of employees in each department.



localhost:8080/#/main/view/HIVE/auto_hive_instance

Database Explorer

employee_id	STRING
gender	STRING
marital_status	STRING
education_level	STRING
annual_salary	INT
department	STRING
job_level	STRING
years_experience	INT
bonus	INT
performance_rating	INT
city	STRING

Query Editor

```
SELECT department, COUNT(employee_id) AS NumberOfEmployees
FROM employee_dataset
GROUP BY department;
```

Query Process Results (Status: SUCCEEDED)

department	numberofemployees
Finance	199
HR	213
IT	214
Marketing	201
Sales	173