Database Management System – 27 (Aggregation functions in SQL, GROUP BY, HAVING)

Ajay James
Asst. Prof in CSE
Government Engineering College Thrissur

Outline

- Aggregate Functions in SQL
- GROUP BY
- HAVING

Aggregate Functions in SQL

- **Aggregate functions**
  - Used to summarize information from multiple tuples into a single-tuple summary
- **Grouping**
  - Used to create subgroups of tuples before summarization
- **COUNT, SUM, MAX, MIN**, and **AVG**
- **COUNT**
  - returns the number of tuples or values as specified in a query.
- **SUM, MAX, MIN**, and **AVG**
- Can be used in **SELECT** clause or in a **HAVING** clause
- **MAX** and **MIN** can also be used with attributes that have nonnumeric domains

Example

- Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary.

```
SELECT SUM (Salary), MAX (Salary), MIN (Salary),
       AVG (Salary)
FROM EMPLOYEE;


SELECT SUM (Salary) AS Total_Sal, MAX (Salary) AS
       Highest_Sal, MIN (Salary) AS Lowest_Sal, AVG
       (Salary) AS Average_Sal
FROM EMPLOYEE;
```

Aggregation example

- Find the sum of the salaries of all employees of the 'Research' department, as well as the maximum salary, the minimum salary, and the average salary in this department.

```
SELECT SUM (Salary), MAX (Salary), MIN (Salary),
       AVG (Salary)
FROM (EMPLOYEE JOIN DEPARTMENT ON Dno =
                                    Dnumber)
WHERE Dname = 'Research';
```

Aggregation example

- Retrieve the total number of employees in the company the number of employees in the 'Research' department

```
SELECT COUNT (*)
FROM EMPLOYEE;
```

- Retrieve the total number of employees in the 'Research' department

```
SELECT COUNT (*)
FROM EMPLOYEE, DEPARTMENT
WHERE DNO = DNUMBER AND DNAME = 'Research';
```

Aggregation example

- Count the number of distinct salary values in the database

```
SELECT COUNT (DISTINCT Salary)
FROM EMPLOYEE;
```

- Retrieve the names of all employees who have two or more dependents

```
SELECT Lname, Fname
FROM EMPLOYEE
WHERE ( SELECT COUNT (*)
        FROM DEPENDENT
        WHERE Ssn = Essn ) > = 2;
```

Grouping: GROUP BY and HAVING Clauses

- For each department, retrieve the department number, the number of employees in the department, and their average salary.

```
SELECT Dno, COUNT (*), AVG (Salary)
FROM EMPLOYEE
GROUP BY Dno;
```

Example

Fname	Minit	Lname	Ssn	...	Salary	Super_ssn	Dno
John	B	Smith	123456789		30000	333445555	5
Franklin	T	Wong	333445555		40000	888665555	5
Ramesh	K	Narayan	666884444		38000	333445555	5
Joyce	A	English	453453453	...	25000	333445555	5
Alicia	J	Zelaya	999887777		25000	987654321	4
Jennifer	S	Wallace	987654321		43000	888665555	4
Ahmad	V	Jabbar	987987987		25000	987654321	4
James	E	Bong	888665555		55000	NULL	1

Grouping EMPLOYEE tuples by the value of Dno

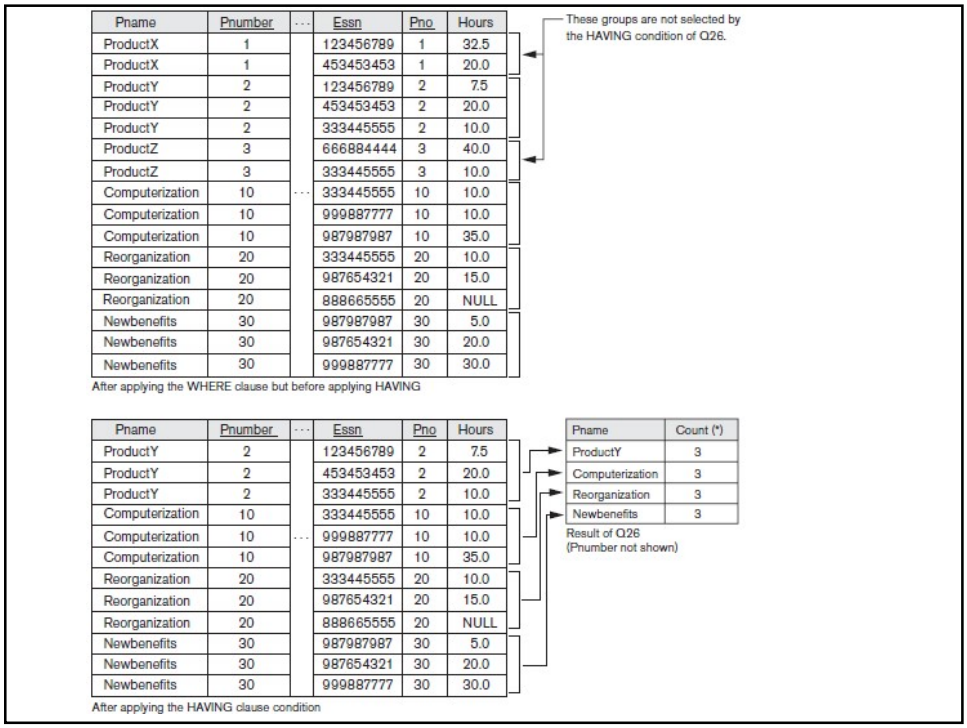
Dno	Count (*)	Avg (Salary)
5	4	33250
4	3	31000
1	1	55000

Result of Q24

HAVING Example

- For each project on which more than two employees work, retrieve the project number, the project name, and the number of employees who work on the project.

  SELECT Pnumber, Pname, **COUNT** (*)
  FROM PROJECT, WORKS_ON
  WHERE Pnumber = Pno
  GROUP BY Pnumber, Pname
  HAVING COUNT (*) > 2;



Pname	Pnumber	...	Essn	Pno	Hours
ProductX	1		123456789	1	32.5
ProductX	1		453453453	1	20.0
ProductY	2		123456789	2	7.5
ProductY	2		453453453	2	20.0
ProductY	2		333445555	2	10.0
ProductZ	3		666884444	3	40.0
ProductZ	3		333445555	3	10.0
Computerization	10	...	333445555	10	10.0
Computerization	10		999887777	10	10.0
Computerization	10		987987987	10	35.0
Reorganization	20		333445555	20	10.0
Reorganization	20		987654321	20	15.0
Reorganization	20		888665555	20	NULL
Newbenefits	30		987987987	30	5.0
Newbenefits	30		987654321	30	20.0
Newbenefits	30		999887777	30	30.0

After applying the WHERE clause but before applying HAVING

Pname	Pnumber	...	Essn	Pno	Hours
ProductY	2		123456789	2	7.5
ProductY	2		453453453	2	20.0
ProductY	2		333445555	2	10.0
Computerization	10		333445555	10	10.0
Computerization	10	...	999887777	10	10.0
Computerization	10		987987987	10	35.0
Reorganization	20		333445555	20	10.0
Reorganization	20		987654321	20	15.0
Reorganization	20		888665555	20	NULL
Newbenefits	30		987987987	30	5.0
Newbenefits	30		987654321	30	20.0
Newbenefits	30		999887777	30	30.0

Pname	Count (*)
ProductY	3
Computerization	3
Reorganization	3
Newbenefits	3

Result of Q26 (Pnumber not shown)

After applying the HAVING clause condition

Summary

SELECT <attribute and function list>
FROM <table list>
[**WHERE** <condition>]
[**GROUP BY** <grouping attribute(s)>]
[**HAVING** <group condition>]
[**ORDER BY** <attribute list>];

Reference

- Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education 6th edition and 7th edition

Thank you