

Aggregation Functions

Lecture By
Binu Jasim
18-Aug-2016

Slides are available @ cslab.org/dbms

Aggregation Functions

- *Max*
- *Avg*
- *Min*
- *Count*
- *Count distinct*

Aggregation functions

```
select max(CGPA)  
from Student;
```

Q18. Find average CGPA of students enrolled in the DBMS course?

```
select avg(CGPA)
from Student, Course
where Student.rollNo = Course.rollNo
and Course.cName = "DBMS";
```

Q19. Find average CGPA of students who have taken a course in the CSE department?

Course

| rollNo | cName | dept | marks |
|--------|------------|-------|-------|
| 123 | DBMS | CSE | 48 |
| 123 | OS | CSE | 36 |
| 399 | DBMS | CSE | 25 |
| 201 | DBMS | CSE | 40 |
| 123 | Statistics | Maths | 39 |
| 201 | Control | EEE | 35.5 |

Q19. Find average CGPA of students who have taken a course in the CSE department?

```
select avg(CGPA)
from Student S, Course C
where S.rollNo = C.rollNo
and C.dept = "CSE";
```

```
select avg(CGPA) from Student
where rollNo in
(select rollNo from Course
where dept="CSE");
```

Q20. Find the number of students enrolled in the DBMS course?

```
select count(*) from Course where  
cName="DBMS";
```

```
select * from Course where cName="DBMS";
```

| rollNo | cName | dept | marks |
|--------|-------|------|-------|
| 123 | DBMS | CSE | 48 |
| 399 | DBMS | CSE | 25 |
| 201 | DBMS | CSE | 40 |

Q21. Find the number of students enrolled in the CSE department?

```
count(distinct rollNo)
```

Q22. Find all courses in which the highest mark scored is ≥ 38 ?

```
select distinct cName from Course C1
where
  (select max(marks) from Course C2
   where C1.cName=C2.cName)  $\geq$  38;
```

How do you simplify the above query?

Q23. Find the courses in which all students scored the same mark?

```
select distinct cName from Course C1
where
    (select max(marks) from Course C
     where C.cName=C1.cName)
=
    (select min(marks) from Course C
     where C.cName=C1.cName);
```

Q24. Find the difference b/w average CGPAs of students enrolled in CSE and Maths department courses?

Student

| rollNo | name | dept | CGPA |
|--------|--------|------|------|
| 123 | Alice | CSE | 8.2 |
| 201 | Bob | EEE | 5.6 |
| 399 | Cherry | CSE | 8.2 |

Q24. By how much the average CGPA of students enrolled in the CSE exceeds the the average CGPA of students enrolled in the Maths department?

```
select CS.avgCGPA - Ms.avgCGPA from  
  
(select avg(CGPA) as avgCGPA  
from Student  
where rollNo in (select rollNo  
from Course where dept="CSE")) CS,  
  
(select avg(CGPA) as avgCGPA  
from Student  
where rollNo in (select rollNo  
from Course where dept="Maths")) Ms;
```

Q24. By how much the average CGPA of students enrolled in the CSE exceeds the the average CGPA of students enrolled in the Maths department?

?

```
(select avg(CGPA)
from Student
where rollNo in (select rollNo
from Course where dept="CSE")) -
```

```
(select avg(CGPA)
from Student
where rollNo in (select rollNo
from Course where dept="Maths"));
```

Q24. By how much the average CGPA of students enrolled in the CSE exceeds the the average CGPA of students enrolled in the Maths department?

```
select
```

```
(select avg(CGPA) from Student  
where rollNo in (select rollNo  
from Course where dept="CSE")) -  
(select avg(CGPA) from Student  
where rollNo in (select rollNo  
from Course where dept="Maths"))  
as diff
```

```
from Student;
```

Q25. Find the number of students enrolled in each course?

```
select cName, count(*)  
from Course group by cName;
```

| cName | count(*) |
|------------|----------|
| ----- | ----- |
| Control | 1 |
| DBMS | 3 |
| OS | 1 |
| Statistics | 1 |

Q25. Find the number of students enrolled in each course?

```
select *  
from Course order by cName;
```

| rollNo | cName | dept | marks |
|--------|------------|-------|-------|
| 201 | Control | EEE | 35.5 |
| 123 | DBMS | CSE | 48 |
| 399 | DBMS | CSE | 25 |
| 201 | DBMS | CSE | 40 |
| 123 | OS | CSE | 36 |
| 123 | Statistics | Maths | 39 |

Q26. Give the name of each student and the number of courses enrolled by them?

```
select S.rollNo, count(*) as NumCourses  
from Course C, Student S  
where S.rollNo=C.rollNo  
group by S.rollNo;
```

Q27. What is the output of the following query?

```
select rollNo, dept, count(*)  
from Course  
group by rollNo, dept;
```

| rollNo | dept | count(*) |
|--------|-------|----------|
| 123 | CSE | 2 |
| 123 | Maths | 1 |
| 201 | CSE | 1 |
| 201 | EEE | 1 |
| 399 | CSE | 1 |

Q28. Find the number of Maths courses attended by each student?

```
select rollNo, count(*)  
from Course  
where dept="Maths"  
group by rollNo;
```

Q28. Find the number of Maths courses attended by each student? (we need all students in the output)

```
select rollNo, count(*)  
  from Course where dept="Maths"  
  group by rollNo
```

union

```
select rollNo, 0 from Student  
  where rollNo not in  
    (select rollNo from Course where  
dept="Maths");
```

| rollNo | count(*) |
|--------|----------|
| ----- | ----- |
| 123 | 1 |
| 201 | 0 |
| 399 | 0 |

Q29. Find the departments offering more than 1 course?

```
select dept, count(*)  
from Course  
group by dept  
having count(*) > 1;
```

| dept | count(*) |
|------|----------|
| CSE | 5 |

It actually finds the departments having more than 1 enrolment.

Q29. Find the departments offering more than 1 course?

```
select dept
from Course
group by dept
having count(distinct cName) > 1;
```

Q29. Find the departments offering more than 1 course?

```
select dept from Course C1
where 1 < (select count(distinct cName)
           from Course C2 where C2.dept=C1.dept);
```

Group by and having are not absolutely essential!

Difference b/w *where* and *having* clauses

- Having applies to the group as a whole
- where applies to each row of the relations (or product of relations) in the from clause

Data Modification Statements

- Insertion
- Deletion
- Updation

Insertion

```
insert into Student  
  values(456, "Trudy", "EEE", 7.6)
```

Q30. All the students taking *Control Theory* should enroll in Statistics as well

Course

| rollNo | cName | dept | marks |
|--------|------------|-------|-------|
| 123 | DBMS | CSE | 48 |
| 123 | OS | CSE | 36 |
| 399 | DBMS | CSE | 25 |
| 201 | DBMS | CSE | 40 |
| 123 | Statistics | Maths | 39 |
| 201 | Control | EEE | 35.5 |

Q30. All the students taking DBMS should enroll in Statistics as well

```
insert into Course
select rollNo, "Statistics", "Maths", null
from Course where cName="Control";
```

Q31. Delete all students who are not taking any classes?

```
delete from Student  
where rollNo not in (select rollNo from  
Course);
```

Q32. Change all students enrolled in DBMS course who are having CGPA < 6 to the OS course

```
select Student.rollNo  
from Student, Course  
where CGPA < 6 and cName = "DBMS";
```

```
update Course  
set cName = "OS", dept = "CSE"  
where rollNo in (select Student.rollNo  
from Student, Course  
where CGPA < 6 and cName = "DBMS");
```

Q32. Change all students enrolled in DBMS course who are having CGPA < 6 to the OS course

```
update Course
set cName="OS", dept="CSE"
where rollNo in (select Student.rollNo
from Student,Course
where CGPA<6 and cName="DBMS" )
and cName="DBMS";
```


NULL

```
select rollNo  
from Student  
where CGPA<6 or CGPA>=6;
```

To check for null attributes:

```
is null
```