GROUP BY and HAVING: Solutions

Schema

Student(sID, surName, firstName, campus, email, cgpa)  Offering[dept, cNum] ⊆ Course[dept, cNum]
Course(dept, cNum, name, breadth)  Took[sID] ⊆ Student[sID]
Offering(oID, dept, cNum, term, instructor)  Took[oID] ⊆ Offering[oID]
Took(sID, oID, grade)

Questions

1. Write a query to find the average grade, minimum grade, and maximum grade for each offering.

Solution:

```
select avg(grade), min(grade), max(grade)
from Took
group by oid;
```

Output:

```
 avg | min | max
---------------------+-----+-----
59.0000000000000000 | 39  | 98  
60.6666666666666667 | 45  | 75  
70.5000000000000000 | 52  | 89  
. . . rows omitted
75.0000000000000000 | 54  | 96  
78.0000000000000000 | 78  | 78  
83.0000000000000000 | 71  | 89  
(23 rows)
(1 row)
```

2. Suppose we wrote

```
SELECT --------------------
FROM Offering, Took
WHERE Offering.oID = Took.oID
group by dept;
```

Which of the following could go in the SELECT clause?

```
sID count(sID) grade avg(grade) dept count(dept) term min(term)
```

Solution: The only unaggregated item that can go in the SELECT is the one that is grouped by: dept. Everything else must be aggregated. And it is legal to aggregate dept too. Here is a query with all the allowed items included:
SELECT count(sID), avg(grade), dept, count(dept), min(term)
FROM Offering, Took
WHERE Offering.oID = Took.oID
group by dept;

Output:

<table>
<thead>
<tr>
<th>count</th>
<th>avg</th>
<th>dept</th>
<th>count</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>69.5000000000000000</td>
<td>ENV</td>
<td>4</td>
<td>20089</td>
</tr>
<tr>
<td>6</td>
<td>78.1666666666666667</td>
<td>EEB</td>
<td>6</td>
<td>20081</td>
</tr>
<tr>
<td>8</td>
<td>78.5000000000000000</td>
<td>ANT</td>
<td>8</td>
<td>20081</td>
</tr>
<tr>
<td>1</td>
<td>97.0000000000000000</td>
<td>HIS</td>
<td>1</td>
<td>20081</td>
</tr>
<tr>
<td>24</td>
<td>79.6666666666666667</td>
<td>CSC</td>
<td>24</td>
<td>20081</td>
</tr>
<tr>
<td>11</td>
<td>63.6363636363636364</td>
<td>ENG</td>
<td>11</td>
<td>20081</td>
</tr>
</tbody>
</table>

(6 rows)

3. Find the sid and average grade of each student, but keep the data only for those students who have an sid over 22222.

Solution:

SELECT Student.sID, surname, avg(grade)
FROM Student, Took
WHERE Student.sID = Took.sID
GROUP BY Student.sID
HAVING Student.sID > 22222;

Output:

<table>
<thead>
<tr>
<th>sid</th>
<th>surname</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>98000</td>
<td>Fairgrieve</td>
<td>83.2000000000000000000000</td>
</tr>
<tr>
<td>99132</td>
<td>Marchmount</td>
<td>76.2857142857142857142857</td>
</tr>
<tr>
<td>99999</td>
<td>Ali</td>
<td>84.5833333333333333333333</td>
</tr>
</tbody>
</table>

(3 rows)

4. Find only the sid (and not also the average grade) of each student with an average over 80.

Solution:

SELECT SID
FROM Took
GROUP BY sID
HAVING AVG(grade) > 80;

Output:

<table>
<thead>
<tr>
<th>sid</th>
</tr>
</thead>
<tbody>
<tr>
<td>98000</td>
</tr>
<tr>
<td>99999</td>
</tr>
</tbody>
</table>

(2 rows)
5. Which of these queries is legal?

```
SELECT dept
FROM Took, Offering
WHERE Took.oID = Offering.oID
GROUP BY dept
HAVING avg(grade) > 75;

SELECT Took.oID, dept, cNum, avg(grade)
FROM Took, Offering
WHERE Took.oID = Offering.oID
GROUP BY Took.oID
HAVING avg(grade) > 75;

SELECT Took.oID, dept, cNum, avg(grade)
FROM Took, Offering
WHERE Took.oID = Offering.oID
GROUP BY Took.oID
HAVING avg(grade) > 75;

SELECT oID, avg(grade)
FROM Took
GROUP BY sID
HAVING avg(grade) > 75;
```

**Solution:** Here’s the result of each:

<table>
<thead>
<tr>
<th>dept</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEB</td>
<td></td>
</tr>
<tr>
<td>ANT</td>
<td></td>
</tr>
<tr>
<td>HIS</td>
<td></td>
</tr>
<tr>
<td>CSC</td>
<td></td>
</tr>
</tbody>
</table>

(4 rows)

<table>
<thead>
<tr>
<th>oid</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>92.0000000000000000</td>
</tr>
<tr>
<td>28</td>
<td>91.0000000000000000</td>
</tr>
<tr>
<td>. . . rows omitted</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>83.0000000000000000</td>
</tr>
</tbody>
</table>

(11 rows)

ERROR: column "offering.dept" must appear in the GROUP BY clause or be used in an aggregate function

LINE 1: SELECT Took.oID, dept, cNum, avg(grade)

ERROR: column "took.oID" must appear in the GROUP BY clause or be used in an aggregate function

LINE 1: SELECT oID, avg(grade)