GROUP BY and HAVING: Solutions

Schema

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

Offering[dept, cNum] \subseteq Course[dept, cNum]
Took[sID] \subseteq Student[sID]
Took[oID] \subseteq Offering[oID]

Questions

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

   Solution:

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

   Output:

<table>
<thead>
<tr>
<th>avg</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>59.0000000000000000</td>
<td>39</td>
<td>98</td>
</tr>
<tr>
<td>60.6666666666666667</td>
<td>45</td>
<td>75</td>
</tr>
<tr>
<td>70.5000000000000000</td>
<td>52</td>
<td>89</td>
</tr>
<tr>
<td>. . . rows omitted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>75.0000000000000000</td>
<td>54</td>
<td>96</td>
</tr>
<tr>
<td>78.0000000000000000</td>
<td>78</td>
<td>78</td>
</tr>
<tr>
<td>83.0000000000000000</td>
<td>71</td>
<td>89</td>
</tr>
<tr>
<td>(23 rows)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(1 row)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Suppose we wrote

   ```sql
   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 from our list:
SELECT count(sID), avg(grade), dept, count(dept), min(term)
FROM Offering, Took
WHERE Offering.oID = Took.oID
GROUP BY dept;

It makes sense that these are allowed. We asked to group by dept, so there will be one row per dept in
the resulting table. This means that we can include in our SELECT only things that have one value per
per dept. There is one count(sID) per dept, one avg(grade) per dept, one dept per dept (so to speak), one
count(dept) per dept, and one min(term) per department.

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>83.2000000000000000</td>
</tr>
<tr>
<td>99132</td>
<td>Marchmount</td>
<td>76.2857142857142857</td>
</tr>
<tr>
<td>99999</td>
<td>Ali</td>
<td>84.5833333333333333</td>
</tr>
</tbody>
</table>
(3 rows)

Here's a more interesting question: For each student who has passed at least 10 courses, report their sid and
average grade on the courses that they passed.

Solution:

SELECT sid, AVG(grade), COUNT(*)
FROM took
WHERE grade >= 50
GROUP BY sid
HAVING COUNT(*) >= 10;

Output:
There is a lot going on here. Be sure you are clear on the difference between WHERE and HAVING, and which rows are left at the moment where the HAVING condition is checked for each group.

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:

```
sid
-----
98000
99999
```

(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, 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:

```
depth                  oid | avg
-----------------------+---------------------
EEB                    8 | 92.0000000000000000
ANT                    28 | 91.0000000000000000
HIS                   ... rows omitted
CSC                    7 | 83.0000000000000000
```

(4 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)