```
-- Recap of GROUP BY

dbsrv1% psql csc343h-dianeh
psql (9.1.14)
Type "help" for help.

csc343h-dianeh=> set search_path to university;
SET
csc343h-dianeh=> \?
csc343h-dianeh=> \d
           List of relations
   Schema   |   Name   | Type  | Owner
------------+----------+-------+--------
 university | course   | table | dianeh
 university | offering | table | dianeh
 university | student  | table | dianeh
 university | took     | table | dianeh
(4 rows)

-- We saw queries like these last class.
-- Try them yourself to get more comfortable with what they do.
csc343h-dianeh=> select * from took;
csc343h-dianeh=> select * from took order by oid;
csc343h-dianeh=> select * from took group by oid;
ERROR:  column "took.sid" must appear in the GROUP BY clause or be used in an aggregate function
LINE 1: select * from took group by oid;
               ^
csc343h-dianeh=> select oid, max(sid), avg(grade) from took group by oid;
csc343h-dianeh=> select oid, max(sid), avg(grade) from took group by oid order by avg(grade);
csc343h-dianeh=> select oid, max(sid), avg(grade) from took group by oid order by avg(grade) desc;
csc343h-dianeh=> select oid, max(sid), avg(grade) from took group by oid order by oid;
csc343h-dianeh=> select oid, max(sid), avg(grade) from took group by oid order by avg(grade), oid;


-- Worksheet

csc343h-dianeh=> -- Question 1
csc343h-dianeh=> -- First let's find these values overall.
csc343h-dianeh=> select avg(grade), min(grade), max(grade) from took;
         avg         | min | max
---------------------+-----+-----
 75.6296296296296296 |   0 | 100
(1 row)

csc343h-dianeh=> -- Now let's find them per oid.
csc343h-dianeh=> select avg(grade), min(grade), max(grade) from took group by oid;
         avg         | min | max
---------------------+-----+-----
 59.0000000000000000 |  39 |  98
 60.6666666666666667 |  45 |  75
 70.5000000000000000 |  52 |  89
 92.0000000000000000 |  91 |  93
 69.5000000000000000 |  46 |  94
 91.0000000000000000 |  91 |  91
 87.2500000000000000 |  79 |  99
 31.0000000000000000 |   0 |  62
 71.0000000000000000 |  71 |  71
 79.0000000000000000 |  39 |  99
 92.0000000000000000 |  92 |  92
 73.5000000000000000 |  17 | 100
 97.0000000000000000 |  97 |  97
 74.0000000000000000 |  72 |  78
 71.0000000000000000 |  71 |  71
 82.0000000000000000 |  82 |  82
 78.0000000000000000 |  70 |  82
 75.0000000000000000 |  75 |  75
 74.6666666666666667 |  59 |  89
 95.6666666666666667 |  90 |  99
 75.0000000000000000 |  54 |  96
 78.0000000000000000 |  78 |  78
 83.0000000000000000 |  71 |  89
(23 rows)

csc343h-dianeh=> -- The results might be more interesting if we added in the oid, and ordered them
somehow.
csc343h-dianeh=> -- Let's do it by average grade.
csc343h-dianeh=> select oid, avg(grade), min(grade), max(grade) from took group by oid order by
avg(grade);
 oid |         avg         | min | max
-----+---------------------+-----+-----
  15 | 31.0000000000000000 |   0 |  62
  14 | 59.0000000000000000 |  39 |  98
  34 | 60.6666666666666667 |  45 |  75
  17 | 69.5000000000000000 |  46 |  94
  27 | 70.5000000000000000 |  52 |  89
  21 | 71.0000000000000000 |  71 |  71
  26 | 71.0000000000000000 |  71 |  71
  16 | 73.5000000000000000 |  17 | 100
   6 | 74.0000000000000000 |  72 |  78
   5 | 74.6666666666666667 |  59 |  89
  35 | 75.0000000000000000 |  75 |  75
  22 | 75.0000000000000000 |  54 |  96
  31 | 78.0000000000000000 |  70 |  82
   9 | 78.0000000000000000 |  78 |  78
  11 | 79.0000000000000000 |  39 |  99
   3 | 82.0000000000000000 |  82 |  82
   7 | 83.0000000000000000 |  71 |  89
   1 | 87.2500000000000000 |  79 |  99
  28 | 91.0000000000000000 |  91 |  91
   8 | 92.0000000000000000 |  91 |  93
  38 | 92.0000000000000000 |  92 |  92
  13 | 95.6666666666666667 |  90 |  99
  39 | 97.0000000000000000 |  97 |  97
(23 rows)

csc343h-dianeh=> -- Question 2:
csc343h-dianeh=> -- Which can go in the select?:
csc343h-dianeh=> -- (Notice that this query joins 2 tables.)
csc343h-dianeh=> -- sid: no
csc343h-dianeh=> -- count(sid): yes
csc343h-dianeh=> -- grade: no
csc343h-dianeh=> -- avg(grade): yes
csc343h-dianeh=> -- dept: yes -- there is one dept per dept, so to speak
csc343h-dianeh=> -- count(dept): yes
csc343h-dianeh=> -- term: no
csc343h-dianeh=> -- min(term): yes
csc343h-dianeh=>
csc343h-dianeh=> -- Let's run the query from Question 2:
csc343h-dianeh=> select count(sid), avg(grade), dept, count(dept), min(term)
csc343h-dianeh-> from offering, took
csc343h-dianeh-> where offering.oid = took.oid
csc343h-dianeh-> group by dept;
 count |         avg         | dept | count |  min
-------+---------------------+------+-------+-------
     4 | 69.5000000000000000 | ENV  |     4 | 20089
     6 | 78.1666666666666667 | EEB  |     6 | 20081
     8 | 78.5000000000000000 | ANT  |     8 | 20081
     1 | 97.0000000000000000 | HIS  |     1 | 20081
    24 | 79.6666666666666667 | CSC  |    24 | 20081
    11 | 63.6363636363636364 | ENG  |    11 | 20081
(6 rows)

csc343h-dianeh=> -- What if we include sid?
csc343h-dianeh=> -- It would make no sense, since there is not one sid per dept.
csc343h-dianeh=> -- So we get that error message that, by now, makes sense too.
csc343h-dianeh=> select count(sid), avg(grade), dept, count(dept), min(term), sid
csc343h-dianeh-> from offering, took
csc343h-dianeh-> where offering.oid = took.oid
csc343h-dianeh-> group by dept;
ERROR:  column "took.sid" must appear in the GROUP BY clause or be used in an aggregate function
LINE 1: ...ct count(sid), avg(grade), dept, count(dept), min(term), sid

                                                                   ^
-- Next feature: HAVING

csc343h-dianeh=> -- Here's a plain GROUP BY.
csc343h-dianeh=> select oid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid;
 oid |         avg         | count
-----+---------------------+-------
  14 | 59.0000000000000000 |     3
  34 | 60.6666666666666667 |     3
  27 | 70.5000000000000000 |     2
   8 | 92.0000000000000000 |     2
  17 | 69.5000000000000000 |     4
  28 | 91.0000000000000000 |     1
   1 | 87.2500000000000000 |     4
  15 | 31.0000000000000000 |     2
  26 | 71.0000000000000000 |     1
  11 | 79.0000000000000000 |     4
  38 | 92.0000000000000000 |     1
  16 | 73.5000000000000000 |     4
  39 | 97.0000000000000000 |     1
   6 | 74.0000000000000000 |     3
  21 | 71.0000000000000000 |     1
   3 | 82.0000000000000000 |     1
  31 | 78.0000000000000000 |     4
  35 | 75.0000000000000000 |     1
   5 | 74.6666666666666667 |     3
  13 | 95.6666666666666667 |     3
  22 | 75.0000000000000000 |     2
   9 | 78.0000000000000000 |     1
   7 | 83.0000000000000000 |     3
(23 rows)

csc343h-dianeh=> -- Now let's filter the results to only the groups having a desired property.
csc343h-dianeh=> select oid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid
csc343h-dianeh-> having count(*) > 1;
 oid |         avg         | count
-----+---------------------+-------
  14 | 59.0000000000000000 |     3
  34 | 60.6666666666666667 |     3
  27 | 70.5000000000000000 |     2
   8 | 92.0000000000000000 |     2
  17 | 69.5000000000000000 |     4
   1 | 87.2500000000000000 |     4
  15 | 31.0000000000000000 |     2
  11 | 79.0000000000000000 |     4
  16 | 73.5000000000000000 |     4
   6 | 74.0000000000000000 |     3
  31 | 78.0000000000000000 |     4
```

```
   5 | 74.6666666666666667 |     3
  13 | 95.6666666666666667 |     3
  22 | 75.0000000000000000 |     2
   7 | 83.0000000000000000 |     3
(15 rows)

csc343h-dianeh=> -- Here's the same query, but with a filter that involves an unaggregated attribute.
csc343h-dianeh=> select oid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid
csc343h-dianeh-> having oid > 10;
 oid |         avg         | count
-----+---------------------+-------
  14 | 59.0000000000000000 |     3
  34 | 60.6666666666666667 |     3
  27 | 70.5000000000000000 |     2
  17 | 69.5000000000000000 |     4
  28 | 91.0000000000000000 |     1
  15 | 31.0000000000000000 |     2
  26 | 71.0000000000000000 |     1
  11 | 79.0000000000000000 |     4
  38 | 92.0000000000000000 |     1
  16 | 73.5000000000000000 |     4
  39 | 97.0000000000000000 |     1
  21 | 71.0000000000000000 |     1
  31 | 78.0000000000000000 |     4
  35 | 75.0000000000000000 |     1
  13 | 95.6666666666666667 |     3
  22 | 75.0000000000000000 |     2
(16 rows)

csc343h-dianeh=> -- But if we filter using grade unaggregated, it doesn't work.
csc343h-dianeh=> -- That makes complete sense.
csc343h-dianeh=> -- We have grouped by oid, so there will be one row per oid (if that oid passes
csc343h-dianeh=> -- the filter).  But the filter is grade < 50, and there is not one grade to check
csc343h-dianeh=> -- per oid.
csc343h-dianeh=> select oid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid
csc343h-dianeh-> having grade < 50;
ERROR:  column "took.grade" must appear in the GROUP BY clause or be used in an aggregate function
LINE 4: having grade < 50;
               ^

csc343h-dianeh=> -- If we change the filter to min(grade) < 50, the query makes sense again.  There
csc343h-dianeh=> -- is only one average grade per oid, so we can compare it to 50.
csc343h-dianeh=> select oid, min(grade), avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid
csc343h-dianeh-> having min(grade) < 50;
 oid | min |         avg         | count
-----+-----+---------------------+-------
  14 |  39 | 59.0000000000000000 |     3
  34 |  45 | 60.6666666666666667 |     3
  17 |  46 | 69.5000000000000000 |     4
  15 |   0 | 31.0000000000000000 |     2
  11 |  39 | 79.0000000000000000 |     4
  16 |  17 | 73.5000000000000000 |     4
(6 rows)

csc343h-dianeh=> -- We can even filter on something that is not in the SELECT clause.
csc343h-dianeh=> select oid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> group by oid
csc343h-dianeh-> having min(grade) < 50;
```

```
 oid |         avg         | count
-----+---------------------+-------
  14 | 59.0000000000000000 |     3
  34 | 60.6666666666666667 |     3
  17 | 69.5000000000000000 |     4
  15 | 31.0000000000000000 |     2
  11 | 79.0000000000000000 |     4
  16 | 73.5000000000000000 |     4
(6 rows)


-- Back to the worksheet

csc343h-dianeh=> -- Question 3 (as originally published on the sheet)
csc343h-dianeh=>
csc343h-dianeh=> select sid, avg(grade)
csc343h-dianeh-> from took
csc343h-dianeh-> group by sid
csc343h-dianeh-> having sid > 22222;
  sid  |         avg
-------+---------------------
 99132 | 76.2857142857142857
 99999 | 84.5833333333333333
 98000 | 83.2000000000000000
(3 rows)

csc343h-dianeh=> -- Here it is without the filter, so we can see what was removed.
csc343h-dianeh=> select sid, avg(grade)
csc343h-dianeh-> from took
csc343h-dianeh-> group by sid;
  sid  |         avg
-------+---------------------
 11111 | 29.6000000000000000
 98000 | 83.2000000000000000
 99132 | 76.2857142857142857
 99999 | 84.5833333333333333
   157 | 75.9333333333333333
(5 rows)

csc343h-dianeh=> -- Question 3 (the new version we did in the noon and 2pm sections).
csc343h-dianeh=> -- For each student who has passed at least 2 courses,
csc343h-dianeh=> -- report their sid and average grade on the courses that they passed.

csc343h-dianeh=> select sid, avg(grade)
csc343h-dianeh-> from took
csc343h-dianeh-> where grade >= 50
csc343h-dianeh-> group by sid
csc343h-dianeh-> having count(*) >= 2;
  sid  |         avg
-------+---------------------
 98000 | 83.2000000000000000
 99132 | 82.5000000000000000
 99999 | 84.5833333333333333
   157 | 78.5714285714285714
(4 rows)

csc343h-dianeh=> -- Let's put the count in so that we can see what was filtered.
csc343h-dianeh=> -- Oh, it filtered nothing because everyone has passed at least two
csc343h-dianeh=> -- courses.
csc343h-dianeh=> select sid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> where grade >= 50
csc343h-dianeh-> group by sid;
  sid  |         avg         | count
-------+---------------------+-------
```

```
 98000 | 83.2000000000000000 |    15
 99132 | 82.5000000000000000 |     6
 99999 | 84.5833333333333333 |    12
   157 | 78.5714285714285714 |    14
(4 rows)

csc343h-dianeh=> -- Let's make the condition more stringent,
csc343h-dianeh=> -- so that there will be some actual filtering.

csc343h-dianeh=> select sid, avg(grade), count(*)
csc343h-dianeh-> from took
csc343h-dianeh-> where grade >= 50
csc343h-dianeh-> group by sid
csc343h-dianeh-> having count(*) >= 10;
  sid  |         avg         | count
-------+---------------------+-------
 98000 | 83.2000000000000000 |    15
 99999 | 84.5833333333333333 |    12
   157 | 78.5714285714285714 |    14
(3 rows)

csc343h-dianeh=> -- Question 4:
csc343h-dianeh=> select sid
csc343h-dianeh-> from took
csc343h-dianeh-> group by sid
csc343h-dianeh-> having avg(grade) > 80;
  sid
-------
 98000
 99999
(2 rows)

csc343h-dianeh=> -- Let's put the average grade in so we can check our results.
csc343h-dianeh=> select sid, avg(grade)
csc343h-dianeh-> from took
csc343h-dianeh-> group by sid
csc343h-dianeh-> having avg(grade) > 80;
  sid  |         avg
-------+---------------------
 98000 | 83.2000000000000000
 99999 | 84.5833333333333333
(2 rows)
```