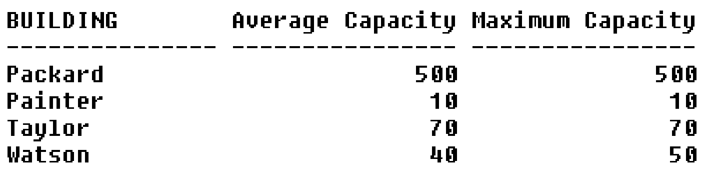
Name: T.Tayyab
Regd.no: Y21CS172

WEEK 4:

Write SQL Select Statements using Aggregate Functions, Group By and Having clauses for the following queries that retrieve data from university database:

1. Find the maximum and average capacity of buildings in the university.

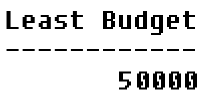
SQL> select building, avg(capacity) as "Average Capacity", max(capacity) as "Maximum Capacity" from classroom group by building;



```
BUILDING        Average Capacity Maximum Capacity
--------------- ---------------- ----------------
Packard                      500              500
Painter                       10               10
Taylor                        70               70
Watson                        40               50
```

2. Display the least budget of the departments.

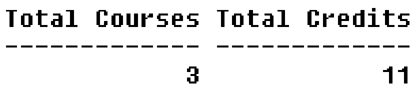
SQL> select min(budget) as "Least Budget" from department;



```
Least Budget
------------
       50000
```

3. Find the total number of courses and credits offered by Biology department.

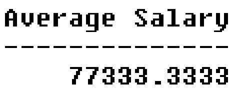
SQL> select count(course_id) as "Total Courses", sum(credits) as "Total Credits" from course where dept_name like 'Biology';



```
Total Courses Total Credits
------------- -------------
            3            11
```

4. Find the average salary of instructors in the Computer Science department.

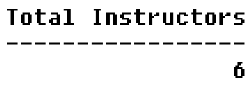
SQL> select avg(salary) as "Average Salary" from instructor where dept_name like 'Comp. Sci.';



```
Average Salary
--------------
    77333.3333
```

5. Find the total number of instructors who teach a course in the Spring 2010 semester.

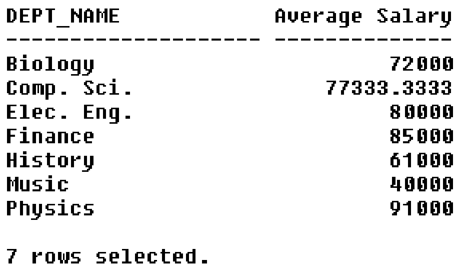
SQL> select count(distinct(id)) as "Total Instructors" from teaches where semester like 'Spring' and year like 2010;



```
Total Instructors
-----------------
                6
```

6. Find the average salary in each department.

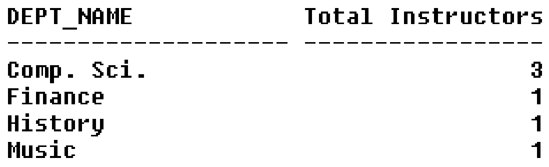
SQL> select dept_name, avg(salary) as "Average Salary" from instructor group by dept_name;



```
DEPT_NAME            Average Salary
-------------------- --------------
Biology                       72000
Comp. Sci.               77333.3333
Elec. Eng.                    80000
Finance                       85000
History                       61000
Music                         40000
Physics                       91000

7 rows selected.
```

7. Find the number of instructors in each department who teach a course in the Spring 2010 semester.

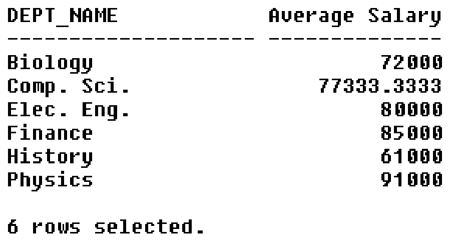
SQL> select i.dept_name, count(distinct(t.id)) as "Total Instructors" from instructor i, teaches t where semester like 'Spring' and year like 2010 and i.id = t.id group by i.dept_name;



```
DEPT_NAME            Total Instructors
-------------------- -----------------
Comp. Sci.                           3
Finance                              1
History                              1
Music                                1
```

8. Find the department name and average salary of the department for only those departments where the average salary of the instructors is more than $42,000.

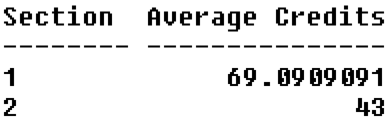
SQL> select dept_name, avg(salary) as "Average Salary" from instructor group by dept_name having avg(salary) >= 42000;



```
DEPT_NAME            Average Salary
-------------------- --------------
Biology                       72000
Comp. Sci.               77333.3333
Elec. Eng.                    80000
Finance                       85000
History                       61000
Physics                       91000

6 rows selected.
```

9. For each course section offered in 2009, find the average total credits (tot_cred) of all students enrolled in the section, if the section had at least 2 students.

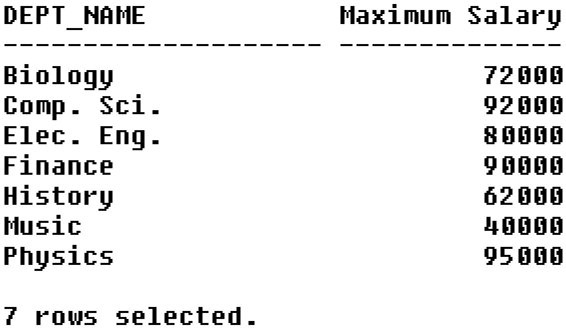
SQL> select t.sec_id as "Section ID", avg(tot_cred) as "Average Credits" from takes t, student s where t.id = s.id and year = 2009 group by t.sec_id having count(t.sec_id) > 1;



```
Section  Average Credits
-------- ---------------
1             69.0909091
2                     43
```

10. For each department, find the maximum salary of instructors in that department. You may assume that every department has at least one instructor.

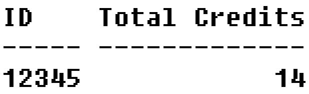
SQL> select dept_name, max(salary) as "Maximum Salary" from instructor group by dept_name;



```
DEPT_NAME            Maximum Salary
-------------------- --------------
Biology                       72000
Comp. Sci.                    92000
Elec. Eng.                    80000
Finance                       90000
History                       62000
Music                         40000
Physics                       95000

7 rows selected.
```

11. For the student with ID 12345 (or any other value), show the total number of credits scored for all courses (taken by that student). Don't display the tot_creds value from the student table, you should use SQL aggregation on courses taken by the student.

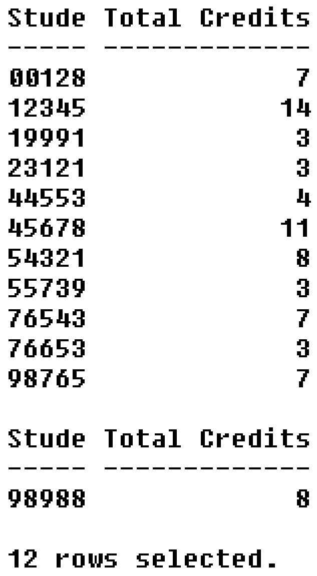
SQL> select t.id, sum(credits) as "Total Credits" from student s,course c,takes t where s.id=t.id and c.course_id=t.course_id group by t.id having t.id=12345;



```
ID    Total Credits
----- -------------
12345            14
```

12. Display the total credits for each of the students, along with the ID of the student; don't bother about the name of the student. (Don't display the tot_creds value from the student table, you should use SQL aggregation on courses taken by the student. For students who have not registered for any course, tot_creds should be 0)

SQL> select s.id as "Student ID", sum(credits) as "Total Credits" from student s, course c, takes t where t.id = s.id and c.course_id = t.course_id group by s.id;

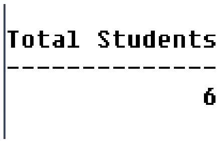


```
Stude Total Credits
----- -------------
00128             7
12345            14
19991             3
23121             3
44553             4
45678            11
54321             8
55739             3
76543             7
76653             3
98765             7

Stude Total Credits
----- -------------
98988             8

12 rows selected.
```

Write nested queries for answering the following queries that retrieve data from university database:

1. Find the total number of (distinct) students who have taken course sections taught by the instructor with ID 10101

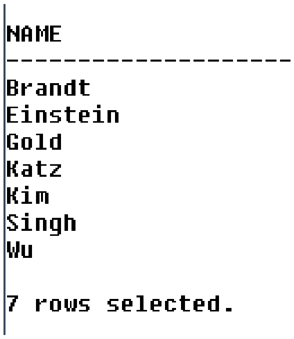
SQL> select count(distinct(t.id)) as "Total Students" from takes t, teaches th where th.id = 10101 and t.course_id = th.course_id and t.sec_id = th.sec_id;



```
Total Students
--------------
             6
```

2. Find the names of all instructors whose salary is greater than at least one instructor in the Biology department.

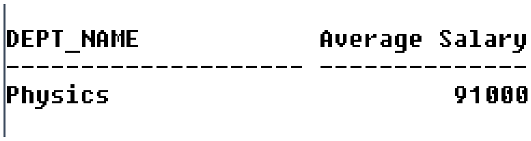
SQL> select distinct(name) from instructor where salary > some(select salary from instructor where dept_name like 'Biology');



```
NAME
--------------------
Brandt
Einstein
Gold
Katz
Kim
Singh
Wu

7 rows selected.
```

3. Find the department that has the highest average salary.

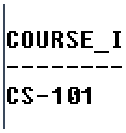
SQL> select dept_name, avg(salary) as "Average Salary" from instructor group by dept_name HAVING avg(salary) >= all(select avg(salary) from instructor group by dept_name);



```
DEPT_NAME            Average Salary
-------------------- --------------
Physics                       91000
```

4. Find all the courses taught in the both the Fall 2009 and Spring 2010 semesters.

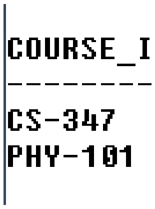
SQL> select distinct(course_id) from teaches where semester = 'Fall' and year = 2009 and course_id in(select distinct(course_id) from teaches where semester = 'Spring' and year = 2010);



```
COURSE_I
--------
CS-101
```

5. Find all the courses taught in the Fall 2009 semester but not in the Spring 2010 semester.

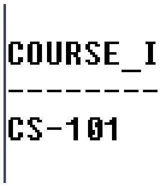
SQL> select distinct(course_id) from teaches where semester = 'Fall' and year = 2009 and course_id not in(select distinct(course_id) from teaches where semester = 'Spring' and year = 2010);



```
COURSE_I
--------
CS-347
PHY-101
```

6. Find all courses taught in both the Fall 2009 semester and in the Spring 2010 semester. (Write correlated nested Query)

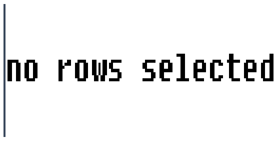
SQL> select course_id from teaches t1 where semester = 'Fall' and year = 2009 and exists(select course_id from teaches t2 where semester = 'Spring' and year = 2010 and t1.course_id = t2.course_id);



```
COURSE_I
--------
CS-101
```

7. Find all students who have taken all courses offered in the Biology department. (Write Correlated nested Query)

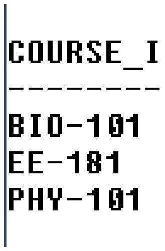
SQL> select distinct(s.id) from student s where not exists((select course_id from course where dept_name like 'Biology') minus (select t.course_id from takes t where t.id = s.id));



```
no rows selected
```

8. Find all courses that were offered at most once in 2009.

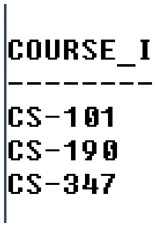
SQL> select distinct(course_id) from takes where year = 2009 group by course_id having count(course_id) <= 1;



```
COURSE_I
--------
BIO-101
EE-181
PHY-101
```

9. Find all courses that were offered at least twice in 2009.

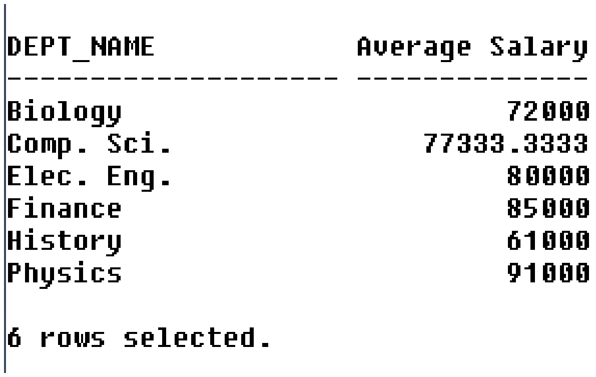
SQL> select distinct(course_id) from takes where year = 2009 group by course_id having count(course_id) >= 2;



```
COURSE_I
--------
CS-101
CS-190
CS-347
```

10. Find the average instructors salaries of those departments where the average salary is greater than $42000.

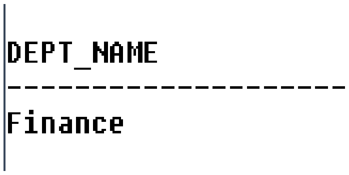
SQL> select dept_name, avg(salary) as "Average Salary" from instructor group by dept_name having avg(salary) >= 42000;



```
DEPT_NAME            Average Salary
-------------------- --------------
Biology                       72000
Comp. Sci.               77333.3333
Elec. Eng.                    80000
Finance                       85000
History                       61000
Physics                       91000

6 rows selected.
```

11. Find the department with the maximum budget.

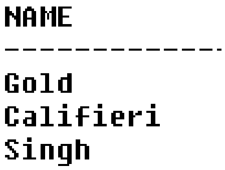
SQL> select dept_name from department where budget = (select max(budget) from department);



```
DEPT_NAME
--------------------
Finance
```

12. Find the names of instructors who have not taught any course.

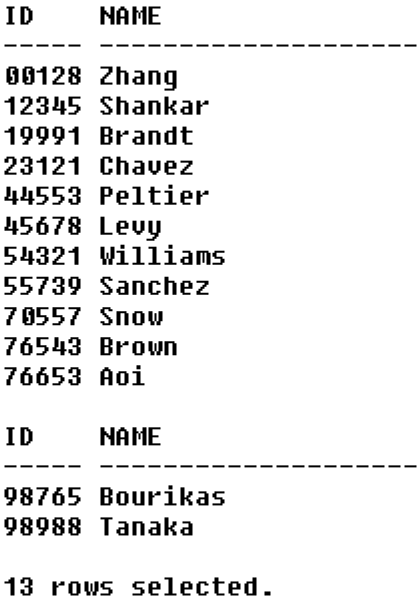
SQL> select name from instructor where ID not in (select ID from teaches);



```
NAME
-------------
Gold
Califieri
Singh
```

13. Find the IDs and names of all students who have not taken any course offering before Spring 2009.

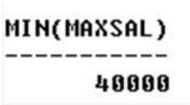
SQL> (select id, name from student) minus (select s.id, s.name from student s, takes t where t.id=s.id and t.year<2009);



```
ID    NAME
----- --------------------
00128 Zhang
12345 Shankar
19991 Brandt
23121 Chavez
44553 Peltier
45678 Levy
54321 Williams
55739 Sanchez
70557 Snow
76543 Brown
76653 Aoi

ID    NAME
----- --------------------
98765 Bourikas
98988 Tanaka

13 rows selected.
```

14. Find the lowest, across all departments, of the per-department maximum salary computed.

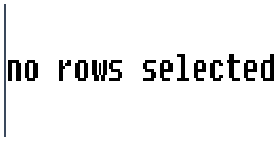
SQL> select min(maxsal) from (select dept_name, max(salary) as maxsal from instructor group by dept_name);



```
MIN(MAXSAL)
-----------
      40000
```

15. Display the IDs and names of the instructors who have taught all Comp. Sci. courses.

SQL> select i.id, i.name from instructor i where not exists((select course_id from course where dept_name='Comp. Sci.') minus (select t.course_id from teaches t where t.id=i.id));



```
no rows selected
```