# **SQL Basics Practice Problems**

1. Find the titles of courses in the Comp. Sci. department that have 3 credits.

**Query:** Find the ID and name of each student who has taken at least one Comp. Sci. course; make sure there are no duplicate names in the result.

2. Find the IDs of all students who were taught by an instructor named Einstein; make sure there are no duplicates.

#### **Query:**

SELECT DISTINCT takes.ID, instructor.name

FROM takes

JOIN section ON takes.course\_id = section.course\_id AND takes.sec\_id = section.sec\_id

JOIN teaches ON section.course\_id = teaches.course\_id AND section.sec\_id = teaches.sec\_id

JOIN instructor ON teaches.ID = instructor.ID

WHERE instructor.name = 'Einstein';

44553 Einstein

3. Find the ID and name of each student who has taken at least one Comp. Sci. course; make sure there are no duplicate names in the result.

# **Query:**

SELECT DISTINCT student.ID, student.name

FROM student

JOIN takes ON student.ID = takes.ID

JOIN course ON takes.course\_id = course.course\_id WHERE course.dept\_name = 'Comp. Sci.';

ID name

00128 Zhang

12345 Shankar

45678 Levy

54321 Williams

76543 Brown

98765 Bourikas

4. Find the course id, section id, and building for each section of a Biology course. Query:

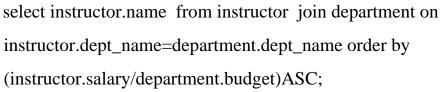
select section.course\_id,section.sec\_id ,section.building from section join course on section.course\_id=course.course\_id where course.dept\_name='Biology';

course\_id sec\_id building
BIO-101 1 Painter
BIO-301 1 Painter

5. Output instructor names sorted by the ratio of their salary to their department's budget(in ascending order).

# **Query:**

Einstein



(instructor.salary/department.budget)ASC;
name
Mozart
Srinivasan
Singh
Katz
Wu
Crick
Brandt
Kim
El Said
Califieri
Gold

6. Output instructor names and buildings for each building an instructor has taught in.Include instructor names who have not taught any classes (the building name should be NULL in this case).

SELECT instructor.name, section.building

FROM instructor

LEFT JOIN teaches ON instructor.ID = teaches.ID

LEFT JOIN section ON teaches.course\_id = section.course\_id AND teaches.sec\_id = section.sec\_id;

name building

Srinivasan Packard

Srinivasan Packard

Srinivasan Watson

Srinivasan Taylor

Wu Packard

Mozart Packard

Einstein Watson

El Said Painter

Gold NULL

Katz Packard

Katz Packard

Katz Watson

Califieri NULL

Singh NULL

Crick Painter

Crick Painter

Brandt Taylor

Brandt Taylor

Brandt Taylor

Kim Taylor

7. 7. Find department names with a budget higher than Astronomy, sorted alphabetically.

#### **Query:**

SELECT dept\_name

FROM department

WHERE budget > (SELECT budget FROM department WHERE dept\_name = 'Astronomy')

ORDER BY dept\_name;

8. Output instructor names and buildings for each building an instructor has taught in. Include instructor names who have not taught any classes (the building name should be NULL in this case).

# **Query:**

SELECT instructor.name, section.building

FROM instructor

LEFT JOIN teaches ON instructor.ID = teaches.ID

LEFT JOIN section ON teaches.course\_id = section.course\_id AND teaches.sec\_id = section.sec\_id;

9. For each student who has retaken a course at least twice (i.e., the student has taken the course at least three times), show the course ID and the student's ID. Please display your results in order of course ID and do not display duplicate rows.

#### **Query:**

SELECT course\_id,ID from takes group by course\_id,ID having count(\*)>=2 order by course\_id,ID;

CS-101 45678

10. Find the names of Biology students who have taken at least 3 Accounting Courses.

# **Query:**

select student.name from student join takes on student.ID=takes.ID join course on takes.course\_id=course.course\_id where student.dept\_name='Biology' and course.dept\_name='Accounting' group by student.ID,student.name having count(\*)>=2;

11. Find the rank and name of the 10 students who earned the most A grades (A-, A, A+), using alphabetical order for ties.

# **Query:**

SELECT name, COUNT(\*) AS grade\_count

FROM student

JOIN takes ON student.ID = takes.ID

WHERE takes.grade IN ('A-', 'A', 'A+')

GROUP BY student.ID, student.name

ORDER BY grade\_count DESC, student.name

LIMIT 10;

name grade\_count

Shankar 3

Brown 2

Zhang 2

Sanchez 1

Tanaka 1

Williams 1

#### PRACTICE PROBLEM

1.Find the names of those departments whose budget is higher than that of Astronomy. List them in alphabetic order.

#### **Query:**

SELECT dept\_name

FROM department

WHERE budget > (SELECT budget FROM department WHERE dept\_name = 'Biology')

ORDER BY dept\_name;

dept\_name Comp.

Sci.

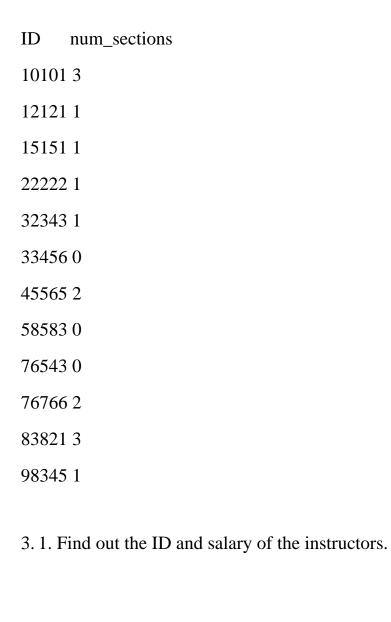
Finance

2. Display a list of all instructors, showing each instructor's ID and the number of sections taught. Make sure to show the number of sections as 0 for instructors who have not taught any section.

# **Query:**

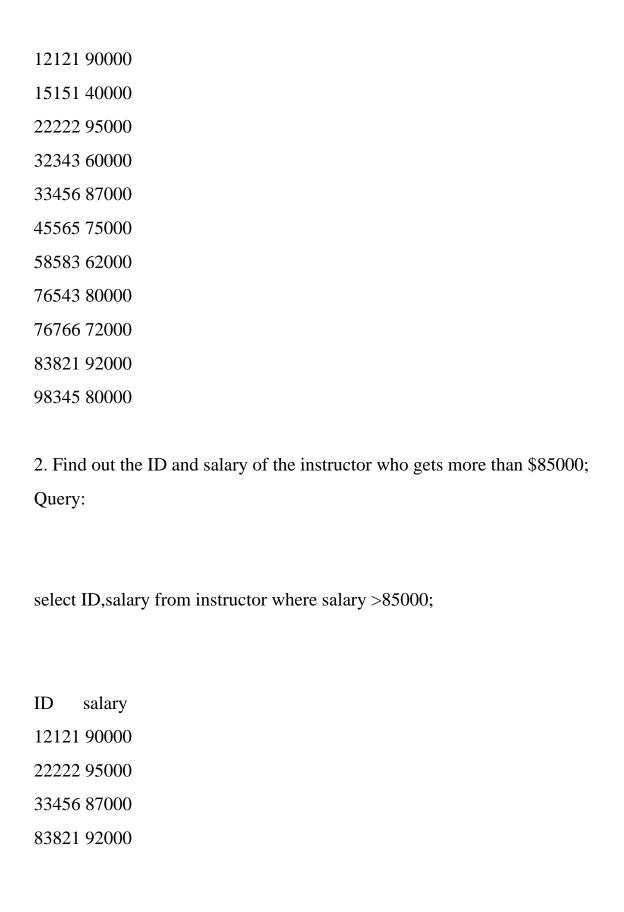
group by instructor.ID;

SELECT instructor.ID , count(teaches.course\_id) as num\_sections from instructor left join teaches on instructor.ID=teaches.ID



SELECT ID, salary from instructor;

ID salary10101 65000



3. Find out the department names and their budget at the university.

select dept\_name,budget from department;

dept\_name budget

Biology 90000

Comp. Sci. 100000

Elec. Eng. 85000

Finance 120000

History 50000

Music 80000

Physics 70000

4. List out the names of the instructors from Computer Science who have more than \$70,000.

# **Query:**

SELECT name from instructor where dept\_name='Comp.Sci.'and salary>70000;

5. For all instructors in the university who have taught some course, find their names and the course ID of all courses they taught.

Query: 127.0.0.1:3309/smalldatabase/instructor/ http://localhost/phpmyadmin/index.php?route=/table/sql&db=smalldatabase&table=instructor

Showing rows 0 - 13 (14 total, Query took 0.0007 seconds.)

SELECT distinct instructor.name ,teaches.course\_id from instructor join teaches on instructor.ID =teaches.ID;

name course\_id

Srinivasan CS-101

Srinivasan CS-315

Srinivasan CS-347

Wu FIN-201

Mozart MU-199

Einstein PHY-101

El Said HIS-351

Katz CS-101

Katz CS-319

Crick BIO-101

Crick BIO-301

Brandt CS-190

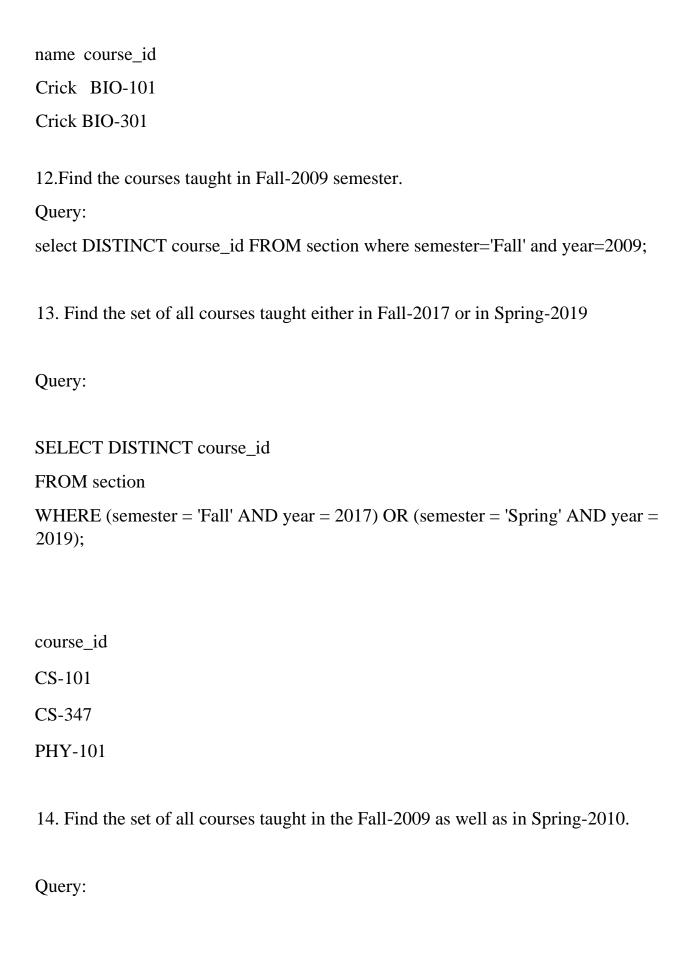
Brandt CS-319

EE-1	81
	EE-J

instructor in the Biology department.
Query:
SELECT name from instructor where salary >(select salary from instructor where dept_name='Biology');
name
Wu
Einstein
Gold
Katz
Singh
Brandt
Kim
7. Find the advisor of the student with ID 12345.
Query:
SELECT s_id from advisor where s_id=12345;
<u> </u>

12345
9. Find the everage colony of all instructors
8. Find the average salary of all instructors.
Query:
SELECT avg(salary) as avg_salary from instructor;
74833.3333
9. Find the names of all departments whose building name include the substring 'watson';
Query:
select dept_name from department where building like '%watson%';
dept_name Biology
DI :
Physics
10. Find the names of instructors with salary amounts between 90000 and 10000;

Query:
select name from instructor where salary BETWEEN 90000 and 100000;
name
Wu
Einstein
Brandt
11. Find the instructor names and the courses they taught for all instructors in the Biology department who have taught some course.
Query:
SELECT instructor.name
,teaches.course_id from instructor join
teaches on instructor.ID=teaches.ID
where instructor.dept_name='Biology';



SELECT DISTINCT	course	id
-----------------	--------	----

FROM section

WHERE (semester = 'Fall' AND year = 2017) AND

course\_id in ( select course\_id from section where semester = 'Spring' AND year = 2018);

CS-101

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

Query:

SELECT DISTINCT course\_id

FROM section

WHERE (semester = 'Fall' AND year = 2017) AND

course\_id not in ( select course\_id from section where semester = 'Spring' AND year = 2018);

course\_id

CS-347

PHY-101

16. Find all instructors who appear in the instructor relation with null values for salary. Query:
select name from instructor where salary is null;
17. Find the average salary of instructors in the Finance department.
Query:
select avg(salary) from instructor where dept_name='Finance';
85000.0000
18. Find the total number of instructors who teach a course in the Spring-2017 semester.
Query:
SELECT count(DISTINCT id) from teaches where
semester='Spring' and year=2017;
2
19. Find the average salary in each department.

# Query:

select dept\_name ,avg(salary) as avg\_salary from instructor group by dept\_name; dept\_name avg\_salary Biology 72000.0000

Comp. Sci. 77333.3333

Elec. Eng. 80000.0000

Finance 85000.0000

History 61000.0000

Music 40000.0000

Physics 91000.0000

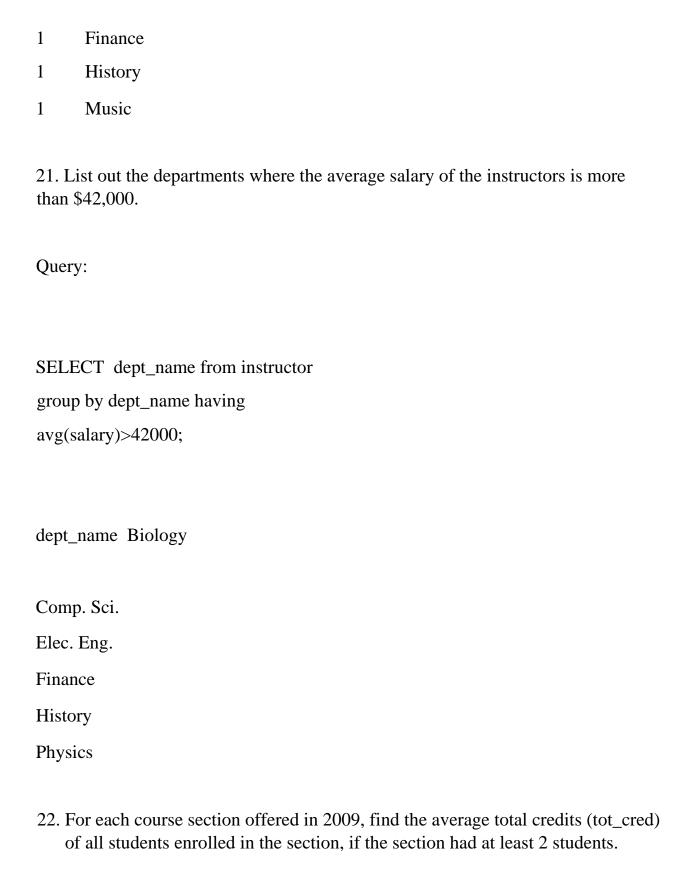
20. Find the number of instructors in each department who teach a course in the Spring-2010 semester.

Query:

SELECT count(DISTINCT teaches.ID) as num\_instructors,instructor.dept\_name from instructor join teaches on instructor.ID=teaches.ID where teaches.semester='Spring' and teaches.year=2018 group by instructor.dept\_name;

num\_instructors dept\_name

3 Comp. Sci.



Onerv	
Query	•

SELECT takes.course\_id, takes.sec\_id, AVG(student.tot\_cred) AS avg\_credits FROM takes JOIN student ON takes.ID = student.ID WHERE takes.year = 2009 GROUP BY takes.course\_id, takes.sec\_id HAVING COUNT(takes.ID) >= 2;

23. Find all the courses taught in both the Fall-2009 and Spring-2010 semesters. Query:

select course\_id from teaches where (semester='Fall' and year =2017)and course\_id in (select course\_id from teaches where semester='Spring' and year='2018');

CS-101

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

Query:

select course\_id from teaches where (semester='Fall' and year =2017)and course\_id not in (select course\_id from teaches where semester='Spring' and year='2018');

course_id
CS-347
PHY-101
25. Select the names of instructors whose names are neither Mozart nor Einstein.
Query:
select name from instructor where name not in ('Mozrat', 'Einstein');
name
Srinivasan
Wu
Mozart
El Said
Gold
Katz
Califieri
Singh
Crick
Brandt
Kim
26. Find the total number of (distinct) students who have taken course sections taught by the instructor with ID 110011.
Query:

SELECT COUNT(DISTINCT takes.ID) FROM takes JOIN teaches ON takes.course\_id = teaches.course\_id AND takes.sec\_id = teaches.sec\_id WHERE teaches.ID = 110011:

27. Find the ID and names of all instructors whose salary is greater than at least one instructor in the History department.

Query:

SELECT ID,name from instructor where salary > any (select salary from instructor where dept\_name='History');

ID name

10101 Srinivasan

12121 Wu

22222 Einstein

33456 Gold

45565 Katz

58583 Califieri

76543 Singh

76766 Crick

83821 Brandt

98345 Kim y:

28. Find the names of all instructors that have a salary value greater than that of each instructor in the Biology department.

Query:

SELECT name

FROM instructor
WHERE salary>all(select salary from instructor where dept_name ='Biology'); name
Wu
Einstein
Gold
Katz
Singh
Brandt
Kim
29. Find the departments that have the highest average salary.
Query:
SELECT dept_name from instructor GROUP by dept_name having avg(salary)=
(SELECT MAX(avg_salary)
FROM (SELECT AVG(salary) AS avg_salary FROM instructor GROUP BY dept_name) AS temp
);
Physics
30. 30. Find all courses taught in both the Fall 2009 semester and in the Spring-

2010 semester.

# Query:

select course\_id from teaches where (semester='Fall' and year =2017)and course\_id in (select course\_id from teaches where semester='Spring' and year='2018');

#### CS-101

31. Find all students who have taken all the courses offered in the Biology department.

#### Query:

```
SELECT DISTINCT takes.ID

FROM takes

WHERE NOT EXISTS (

SELECT course_id

FROM course

WHERE dept_name = 'Biology'

AND course_id NOT IN (

SELECT course_id

FROM takes

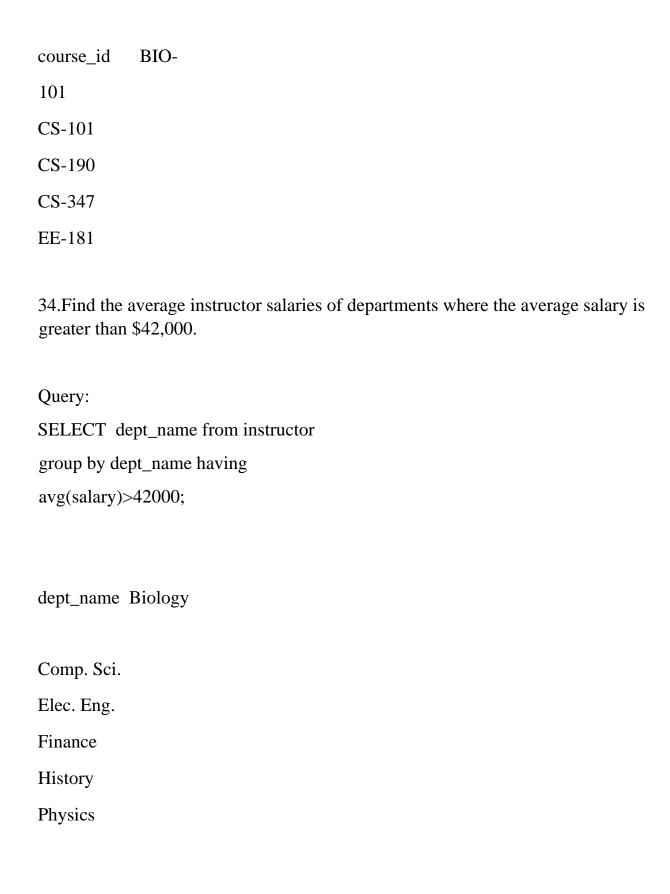
WHERE takes.ID = student.ID

)
```

32. Find all courses that were offered at most once in 20017.

# Query: SELECT course\_id FROM section WHERE year = 2017GROUP BY course\_id HAVING COUNT(sec\_id) <= 1;</pre> course\_id BIO-101 CS-101 CS-347 EE-181 PHY-101 33. Find all courses that were offered at least twice in 2009. Query: SELECT course\_id FROM section WHERE year = 2017GROUP BY course\_id

HAVING COUNT(sec\_id) <= 2;</pre>



35. Find the maximum across all departments of the total salary at each department. Query: SELECT MAX(total\_salary) FROM ( SELECT dept\_name, SUM(salary) AS total\_salary FROM instructor GROUP BY dept\_name ) AS dept\_salaries; 232000 36. List all departments along with the number of instructors in each department. Query: SELECT dept\_name, COUNT(\*) AS num\_instructors FROM instructor GROUP BY dept\_name; dept\_name num\_instructors Biology

# Comp. Sci. 3

Elec. Eng. 1

Finance 2

History 2

Music 1

Physics 2