

```

1  /* Queries tables
2  course (course_id, title, dept_name, credits)
3  instructor (ID, name, dept_name, salary)
4  teaches (ID, course_id, sec_id, semester, year)
5  student (ID, name, dept_name, tot_cred)
6  takes (ID, course_id, sec_id, semester, year, grade)
7
8
9  clear break
10 clear comp
11 clear col
12
13 set linesize 9999
14 set trimspool ON
15 set tab off
16 set echo off
17 set feedback off
18 set recsep off
19
20 */
21
22
23
24 /* Pre-setup display for ey to look */
25 clear break
26 clear comp
27 clear col
28
29 set linesize 9999
30 set trimspool ON
31 set tab off
32 set echo off
33 set feedback off
34 set recsep off
35
36 -- 1. Find the student's name whose ID = "113". */
37
38 SELECT name
39 FROM student
40 WHERE id='113';
41
42 -- 2. List all courses which title starts with "G"
43
44 SELECT *
45 FROM course
46 WHERE title LIKE 'G%';
47
48 -- 3. List all instructor IDs who did not teach any courses in Fall 2016.
49 -- SELECT DISTINCT * FROM INSTRUCTOR NATURAL JOIN TEACHES WHERE SEMESTER!='Fall'
50 AND YEAR!='2016' ORDER BY ID;
51
52 SELECT * FROM instructor WHERE id NOT IN (SELECT id FROM teaches WHERE semester=
53 'fall' AND year= 2016);
54
55 -- 4. Find the total number of students in each department. Display the number in
56 ascending order.
57 SELECT DEPT_NAME, COUNT(DEPT_NAME) CNT
58 FROM student
59 GROUP BY dept_name
60 ORDER BY CNT ASC;
61
62 -- 5. Find the name of instructor who teaches the most students.
63 /* Works ON 3 tables, takes, teaches, AND instructor
64 instructor (ID, name, dept_name, salary)
65 teaches (ID, course_id, sec_id, semester, year)

```

```

63  takes (ID, course_id, sec_id, semester, year, grade)
64
65  */
66  -- new method
67  SELECT ins.id, ins.name
68  FROM instructor ins,
69  (SELECT t1.id, COUNT(t1.id)
70   FROM (SELECT teaches.id id, takes.id studentid, takes.course_id, takes.sec_id,
71         takes.semester, takes.year
72        FROM teaches INNER JOIN takes
73        ON takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id AND
74         takes.semester = teaches.semester AND takes.year = teaches.year
75        ORDER BY takes.year) t1
76   GROUP BY id
77   HAVING COUNT(t1.id) =
78   (SELECT MAX(COUNT(teachid))
79   FROM (SELECT teaches.id teachid, takes.id studentid, takes.course_id, takes.
80        sec_id, takes.semester, takes.year
81        FROM teaches INNER JOIN takes
82        ON takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id
83         AND takes.semester = teaches.semester AND takes.year = teaches.year
84        ORDER BY takes.year)
85   GROUP BY teachid)
86  ) rs
87  WHERE ins.id = rs.id;
88
89  -- Method 2, use more select
90
91  SELECT ins.id, ins.name FROM instructor ins,
92  (SELECT t2.teachid, t1.cnt
93   FROM (SELECT max(cnt) cnt
94        FROM (SELECT id, count(*) cnt
95              FROM (SELECT teaches.id id, takes.id studentid, takes.course_id, takes.
96                    sec_id, takes.semester, takes.year
97                   FROM teaches
98                   INNER JOIN takes ON takes.course_id = teaches.course_id AND takes.
99                    sec_id = teaches.sec_id AND takes.semester = teaches.semester AND
100                    takes.year = teaches.year
101                   ORDER BY takes.year)
102                  GROUP BY id)
103              ) t1,
104        (SELECT id teachid, cnt
105        FROM (SELECT id, count(*) cnt
106              FROM (SELECT teaches.id id, takes.id studentid, takes.course_id, takes.
107                    sec_id, takes.semester, takes.year
108                   FROM teaches
109                   INNER JOIN takes ON takes.course_id = teaches.course_id AND takes.
110                    sec_id = teaches.sec_id AND takes.semester = teaches.semester AND
111                    takes.year = teaches.year
112                   ORDER BY takes.year)
113                  GROUP BY id)
114              ) t2
115   WHERE t1.cnt = t2.cnt) rs
116  WHERE rs.teachid = ins.id;
117
118  -- 6. List all instructors who teach in all those years that the instructor
119  "McKinnon" teaches.
120  -- I remove the name of McKinnon in the list
121
122  SELECT teaches.id, instructor.name
123  FROM teaches INNER JOIN instructor

```

```

117 ON teaches.id = instructor.id
118 WHERE teaches.year IN
119     (SELECT DISTINCT t.year
120     FROM teaches t, instructor i
121     WHERE i.name='McKinnon' AND t.id=i.id) AND instructor.name != 'McKinnon'
122 GROUP BY teaches.id, instructor.name
123 ORDER BY teaches.id;
124
125 -- 7. For the department WHERE the instructors have the highest average salary,
126 list the top 2 instructors who have the highest salaries AND their salaries.
127
128 -- new method
129 SELECT name, salary
130 FROM instructor
131 WHERE dept_name =
132     (SELECT t1.dept_name
133     FROM instructor t1
134     GROUP BY t1.dept_name
135     HAVING AVG(t1.salary) >=
136         (SELECT MAX(AVG(salary))
137         FROM instructor
138         GROUP BY dept_name))
139 AND ROWNUM <=
140 ORDER BY salary DESC;
141
142 /*
143 Old method
144
145 SELECT name, salary
146 FROM instructor
147 WHERE dept_name =
148     (SELECT t2.dept_name
149     FROM (SELECT max(avg(salary)) max
150     FROM instructor
151     GROUP BY dept_name) t1,
152     (SELECT dept_name, avg(salary) avg_salary
153     FROM instructor
154     GROUP BY dept_name) t2
155     WHERE t1.max = t2.avg_salary)
156 AND rownum <=2
157 ORDER BY salary DESC;
158
159 */
160
161 -- 8. Generate "transcript records" for all students of "Comp. Sci." department. A
162 transcript record should include student name, course title, the year and semester
163 when the student took this course, the credits of this course and the grade of the
164 student on this course. The transcript records from one student should be shown
165 together, and in year, semester descending order. Return only 5 of those transcript
166 records.
167
168 -- 9. Increase the salary of instructors whose salary <= 50000 by 10000.
169 UPDATE instructor
170 SET salary = salary + 10000
171 WHERE salary <= 50000;
172
173 -- 10. Delete all the records in table "takes" which students' name = "Tomason".
174
175

```

176  
177  
178  
179  
180  
181  
182  
183