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

```
63      GROUP BY teaches.id
64      HAVING COUNT(teaches.id) =
65          (SELECT MAX(COUNT(teaches.id))
66          FROM teaches INNER JOIN takes
67          ON takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id AND
              takes.semester = teaches.semester AND takes.year = teaches.year
68          GROUP BY teaches.id)
69      ) idtable
70  WHERE instructor.id = idtable.id;
71
72
73  /*
74  -- new method
75  SELECT ins.id, ins.name
76  FROM instructor ins,
77      (SELECT t1.id, COUNT(t1.id)
78      FROM (SELECT teaches.id id, takes.id studentid, takes.course_id, takes.sec_id,
        takes.semester, takes.year
79          FROM teaches INNER JOIN takes
80          ON takes.course_id = teaches.course_id AND takes.sec_id = teaches.sec_id
            AND takes.semester = teaches.semester AND takes.year = teaches.year
81          ORDER BY takes.year) t1
82      GROUP BY id
83      HAVING COUNT(t1.id) =
84          (SELECT MAX(COUNT(teachid))
85          FROM (SELECT teaches.id teachid, takes.id studentid, takes.course_id,
            takes.sec_id, takes.semester, takes.year
86              FROM teaches INNER JOIN takes
87              ON takes.course_id = teaches.course_id AND takes.sec_id =
                teaches.sec_id AND takes.semester = teaches.semester AND takes.year =
                teaches.year
88              ORDER BY takes.year)
89          GROUP BY teachid)
90      ) rs
91  WHERE ins.id = rs.id;
92
93  */
94
95  -- Method 2, use more select
96  /*
97  SELECT ins.id, ins.name FROM instructor ins,
98      (SELECT t2.teachid, t1.cnt
99      FROM (SELECT max(cnt) cnt
100         FROM (SELECT id, count(*) cnt
101             FROM (SELECT teaches.id id, takes.id studentid, takes.course_id,
                takes.sec_id, takes.semester, takes.year
102                 FROM teaches INNER JOIN takes
103                 ON takes.course_id = teaches.course_id AND takes.sec_id =
                    teaches.sec_id AND takes.semester = teaches.semester AND takes.year
                    = teaches.year
104                 ORDER BY takes.year)
105             GROUP BY id)
106         ) t1,
107         (SELECT id teachid, cnt
108         FROM (SELECT id, count(*) cnt
109             FROM (SELECT teaches.id id, takes.id studentid, takes.course_id,
                takes.sec_id, takes.semester, takes.year
110                 FROM teaches INNER JOIN takes
111                 ON takes.course_id = teaches.course_id AND takes.sec_id =
                    teaches.sec_id AND takes.semester = teaches.semester AND 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   */
119
120   -- 6. List all instructors who teach in all those years that the instructor
      "McKinnon" teaches.
121   -- I remove the name of McKinnon in the list
122
123   SELECT teaches.id, instructor.name
124   FROM teaches INNER JOIN instructor
125   ON teaches.id = instructor.id
126   WHERE teaches.year IN
127   ⟶(SELECT DISTINCT t.year
128   ⟶FROM teaches t, instructor i
129   ⟶WHERE i.name='McKinnon' AND t.id=i.id) AND instructor.name != 'McKinnon'
130   GROUP BY teaches.id, instructor.name
131   ORDER BY teaches.id;
132
133   -- 7. For the department WHERE the instructors have the highest average salary,
      list the top 2 instructors who have the highest salaries AND their salaries.
134
135   -- new method
136   SELECT name, salary
137   FROM instructor
138   WHERE dept_name =
139   ⟶(SELECT t1.dept_name
140   ⟶FROM instructor t1
141   ⟶GROUP BY t1.dept_name
142   ⟶HAVING AVG(t1.salary) >=
143   ⟶⟶(SELECT MAX(AVG(salary))
144   ⟶⟶FROM instructor
145   ⟶⟶GROUP BY dept_name))
146   AND ROWNUM <=
147   ORDER BY salary DESC;
148
149
150   /*
151   Old method
152
153   SELECT name, salary
154   FROM instructor
155   WHERE dept_name =
156       (SELECT t2.dept_name
157       FROM (SELECT max(avg(salary)) max
158           FROM instructor
159           GROUP BY dept_name) t1,
160       (SELECT dept_name, avg(salary) avg_salary
161           FROM instructor
162           GROUP BY dept_name) t2
163       WHERE t1.max = t2.avg_salary)
164   AND rownum <=2
165   ORDER BY salary DESC;
166
167   */
168
169   -- 8. Generate "transcript records" for all students of "Comp. Sci." department. A
      transcript record should include student name, course title, the year and semester
      when the student took this course, the credits of this course and the grade of the
      student on this course. The transcript records from one student should be shown
      together, and in year, semester descending order. Return only 5 of those transcript
      records.
170
171   WITH st AS
172    (SELECT  id, name
173     FROM (SELECT *
```

```
174         FROM student
175         WHERE dept_name='Comp. Sci.'
176         ORDER BY name)
177      WHERE ROWNUM <=5)
178   SELECT st.name, course.title, takes.semester, takes.year, takes.grade, course.credits
179   FROM st, takes INNER JOIN course
180   ON course.course_id=takes.course_id
181   WHERE takes.id=st.id
182   ORDER BY name;
183
184
185   -- 9. Increase the salary of instructors whose salary <= 50000 by 10000.
186
187   UPDATE instructor
188   SET salary = salary + 10000
189   WHERE salary <= 50000;
190
191   -- 10. Delete all the records in table "takes" which students' name = "Tomason".
192
193   DELETE FROM takes
194   WHERE id
195   IN (SELECT id
196       FROM student
197       WHERE name='Tomason');
198
199
200
201
202
203
204
205
206
207
208
209
210
```