

```

1  --=====
2  --
3  -- DBMS, HW6
4  -- Made by Agapov George, group M3437
5
6  -- =====
7
8
9  -- Schema (without constraints):
10
11 --
12 -- CREATE TABLE Student (
13 --     Id SERIAL NOT NULL PRIMARY KEY,
14 --     Name VARCHAR (50) NOT NULL,
15 --     GroupId INT NOT NULL
16 -- );
17 -- CREATE TABLE Lecturer (
18 --     Id SERIAL NOT NULL PRIMARY KEY,
19 --     Name VARCHAR (50) NOT NULL
20 -- );
21 -- CREATE TABLE StudentGroup (
22 --     Id SERIAL NOT NULL PRIMARY KEY,
23 --     Name VARCHAR (50) NOT NULL UNIQUE
24 -- );
25 -- CREATE TABLE Course (
26 --     Id SERIAL NOT NULL PRIMARY KEY,
27 --     Name VARCHAR (50) NOT NULL
28 -- );
29 -- CREATE TABLE Teaching (
30 --     GroupId INT NOT NULL,
31 --     CourseId INT NOT NULL,
32 --     LecturerId INT NOT NULL,
33 --     PRIMARY KEY (GroupId, CourseId)
34 -- );
35 -- CREATE TABLE Mark (
36 --     StudentId INT NOT NULL,
37 --     CourseId INT NOT NULL,
38 --     Mark DECIMAL (4, 2) NOT NULL,
39 --     PRIMARY KEY (StudentId, CourseId)
40 -- );
41
42
43 -- -----
44 -- #1
45 -- Информацию о студентах с заданной оценкой по предмету «Базы данных» .
46 -- -----
47
48 -- Let DBMS course have id = 1, interesting mark = 67
49
50 SELECT s.* FROM Student s
51 INNER JOIN Teaching t ON t.GroupId = s.GroupId
52 INNER JOIN Mark m ON s.Id = m.StudentId AND t.CourseId = m.CourseId
53 WHERE m.CourseId = 1 AND m.Mark = 67
54 ;
55
56
57
58
59
60
61
62

```

```

63  -- -----
64  -- #2a
65  -- Информацию о студентах не имеющих оценки по предмету «Базы данных» :
66  -- (1) среди всех студентов
67  -- -----
68
69  SELECT s.* FROM Student s
70  LEFT JOIN Mark m ON s.Id = m.StudentId AND m.CourseId = 1
71  WHERE m.Mark IS NULL
72  ;
73
74  -- -----
75  -- #2b
76  -- Информацию о студентах не имеющих оценки по предмету «Базы данных» :
77  -- (2) среди студентов , у которых есть этот предмет
78  -- -----
79
80  SELECT s.* FROM Student s
81  INNER JOIN Teaching t ON t.GroupId = s.GroupId
82  LEFT JOIN Mark m ON s.Id = m.StudentId AND m.CourseId = t.CourseId
83  WHERE m.Mark IS NULL AND t.CourseId = 1
84  ;
85
86  -- -----
87  -- #3
88  -- Информацию о студентах , имеющих хотя бы одну оценку у заданного лектора .
89  -- -----
90
91  SELECT s.* FROM Student s
92  INNER JOIN Teaching t ON t.GroupId = s.GroupId
93  INNER JOIN Mark m ON s.Id = m.StudentId AND t.CourseId = m.CourseId
94  WHERE t.LecturerId = 1
95  GROUP BY s.id, t.LecturerId
96  ;
97
98  -- -----
99  -- #4
100 -- Идентификаторы студентов , не имеющих ни одной оценки у заданного лектора
101 -- .
102 -- -----
103
104 SELECT s.* FROM Student s
105 LEFT JOIN Teaching t ON t.GroupId = s.GroupId AND t.LecturerId = 1
106 LEFT JOIN Mark m ON s.Id = m.StudentId AND t.CourseId = m.CourseId
107 WHERE m.Mark IS NULL
108 ;
109
110 -- -----
111 -- #5a
112 -- Студентов , имеющих оценки по всем предметам заданного лектора .
113 -- (имеющие оценки по всем предметам заданного лектора , которые они должны
114 -- посещать )
115 -- -----
116
117 SELECT s.* FROM Teaching t
118 INNER JOIN Student s ON s.GroupId = t.GroupId
119 LEFT JOIN Mark m ON m.CourseId = t.CourseId AND s.Id = m.StudentId
120 WHERE t.LecturerId = 1
121 GROUP BY s.Id
122 HAVING bool_and(m.Mark IS NOT NULL)
123 ;

```

```
123  -- -----
124  -- #5b
125  -- Студентов , имеющих оценки по всем предметам заданного лектора .
126  -- (имеющие оценки по всем предметам , которые читает заданный лектор )
127  -- -----
128
129  SELECT s.* FROM Teaching t
130  CROSS JOIN Student s
131  LEFT JOIN Mark m ON m.CourseId = t.CourseId AND s.Id = m.StudentId
132  WHERE t.LecturerId = 1
133  GROUP BY s.Id
134  HAVING bool_and(m.Mark IS NOT NULL)
135  ;
136
137  -- -----
138  -- #6
139  -- Для каждого студента имя и предметы , которые он должен посещать .
140  -- -----
141
142  SELECT s.Id, s.Name, c.Id, c.Name FROM Student s
143  INNER JOIN Teaching t ON t.GroupId = s.GroupId
144  INNER JOIN Course c ON c.Id = t.CourseId
145  ;
146
147  -- -----
148  -- #7
149  -- По лектору всех студентов , у которых он хоть что-нибудь преподавал .
150  -- -----
151
152  SELECT s.* FROM Teaching t
153  INNER JOIN Student s ON s.GroupId = t.GroupId
154  WHERE t.LecturerId = 1
155  ;
156
157  -- -----
158  -- #8
159  -- Пары студентов , такие , что все сданные первым студентом предметы сдал и
160  -- второй студент .
161  -- -----
162
162  SELECT s1.Id, s1.Name, s2.Id, s2.Name FROM Mark m1
163  INNER JOIN Student s1 ON s1.Id = m1.StudentId
164  INNER JOIN Student s2 ON s2.Id > m1.StudentId
165  LEFT JOIN Mark m2 ON m1.CourseId = m2.CourseId AND m2.StudentId = s2.Id
166  GROUP BY s1.Id, s2.Id
167  HAVING bool_and(m2.Mark IS NOT NULL)
168  ;
169
```