

IST210 (Spring 2015): Midterm exam

March 13, 2015

Assume you have a database named COURSES_DB and the following table entries. The table design is listed as below:

```
CREATE TABLE STUDENTS
(
  STUDENT_ID INT PRIMARY KEY NOT NULL,
  LAST_NAME CHAR(20),
  FIRST_NAME CHAR(20),
  DEPT CHAR(15),
  FACULTY_ID INT
);
```

```
CREATE TABLE FACULTY
(
  FACULTY_ID INT PRIMARY KEY NOT NULL,
  LAST_NAME CHAR(20),
  FIRST_NAME CHAR(20),
  DEPT CHAR(15)
);
```

The table row entries are as follows:

STUDENTS table:

STUDENT_ID	LAST_NAME	FIRST_NAME	DEPT	FACULTY_ID
100	Cars	Alex	IST	1000
101	Dance	Betty	IST	1000
102	Eagle	Charles	CS	1000
103	Face	David	CHEM	1002
104	Gould	Elaine	PHY	1001

FACULTY table:

FACULTY_ID	LAST_NAME	FIRST_NAME	DEPT
1000	Height	John	IST
1001	River	Jane	CS
1002	Time	Kay	CHEM

1. Given the two tables (i.e. STUDENTS and FACULTY), which of the following best describes the relationship that the two tables represent?
 - (a) Which department students or faculty belong to
 - (b) Advisor, advisee relationship between students and faculty
 - (c) Assigning different ID numbers for students (starting from 100) and faculty (starting from 1000)
 - (d) None of the above
2. From Question 1 above, which specific field name (or category, table header) best represents the relationship between the two tables?
 - (a) STUDENT_ID
 - (b) FACULTY_ID
 - (c) LAST_NAME
 - (d) DEPT

Correct any errors from the following SQL statements, if any. Assume the tables above are already created and present in the database.

3. SELECT LAST_NAME, DEPARTMENT FROM FACULTY;
4. SELECT LAST_NAME, FIRST_NAME, FROM STUDENTS;
5. SELECT LAST_NAME, FIRST_NAME FROM STUDENTS WHERE DEPT=IST;
6. INSERT INTO STUDENTS VALUES 100, 'Henry', 'Fanny', 'BIO', 1002;
7. INSERT INTO FACULTY VALUES (1003, Urlichs, Linda, CHEM);
8. INSERT INTO FACULTY VALUES (1004, 'Great', 'Lehrer');
9. CREATE TABLE COURSES (COURSE_ID INT PRIMARY KEY, COURSE_TITLE);
10. SELECT * FROM STUDENTS WHERE FACULTY_ID (1000, 1001);

11. SELECT * FROM STUDENTS WHERE FACULTY_ID != NULL;
12. SELECT * FROM STUDENTS WHERE STUDENT_ID => 102;
13. SELECT * FROM STUDENTS WHERE FACULTY_ID = 1000 OR 1001;
14. SELECT * FROM STUDENTS
WHERE FACULTY_ID IN SELECT FACULTY_ID
FROM FACULTY WHERE DEPT = 'IST' OR DEPT = 'CHEM';
15. UPDATE STUDENTS (LAST_NAME = 'Gold', DEPT = 'BIO') WHERE STUDENT_ID = 104;
16. INSERT FACULTY (FACULTY_ID, LAST_NAME, FIRST_NAME) VALUES (1004, 'Winter', 'Snow');
17. DESC TABLE FACULTY;

Given the task and the output from the MySQL command-line client, what will be the corresponding SQL query? Fill in the blank.

18. Task: List all the entries from FACULTY table.

```
SQL query>
(/* FILL IN THE BLANK */) FROM FACULTY;
```

OUTPUT>

```
+-----+-----+-----+-----+
| FACULTY_ID | LAST_NAME | FIRST_NAME | DEPT |
+-----+-----+-----+-----+
|          1000 | Height   | John      | IST  |
|          1001 | River    | Jane      | CS   |
|          1002 | Time     | Kay       | CHEM |
+-----+-----+-----+-----+
```

19. What is the first name and last name of the students who are assigned to faculty ID 1000?

```
SQL query>
SELECT (/* FILL IN THE BLANK */)
FROM (/* FILL IN THE BLANK */)
WHERE (/* FILL IN THE BLANK */);
```

OUTPUT>

```
+-----+-----+
| FIRST_NAME | LAST_NAME |
+-----+-----+
| Alex       | Cars      |
| Betty      | Dance     |
| Charles    | Eagle     |
+-----+-----+
```

20. List the first names and last names of the faculty whose first name contains 'j'.

```
SQL query>
SELECT FIRST_NAME, LAST_NAME
FROM (/* FILL IN THE BLANK */)
WHERE FIRST_NAME (/* FILL IN THE BLANK */);
```

OUTPUT>

```
+-----+-----+
| FIRST_NAME | LAST_NAME |
+-----+-----+
| John       | Height    |
| Jane       | River     |
+-----+-----+
```

21. How many students are there in the IST department?

```
SQL query>
SELECT (/* FILL IN THE BLANK */)
FROM STUDENTS
WHERE (/* FILL IN THE BLANK */);
```

```
OUTPUT>
+-----+
| count(*) |
+-----+
|         2 |
+-----+
```

22. List first name, last name, department of the students either in CS department or CHEM department.

```
SQL query>
SELECT FIRST_NAME, LAST_NAME, DEPT
FROM STUDENTS
WHERE DEPT IN (/* FILL IN THE BLANK */);
```

```
OUTPUT>
+-----+-----+-----+
| FIRST_NAME | LAST_NAME | DEPT |
+-----+-----+-----+
| Charles    | Eagle     | CS   |
| David      | Face      | CHEM |
+-----+-----+-----+
```

23. List the names of the student and the assigned faculty advisor for all students.

```
SQL query>
SELECT ( /* FILL IN THE BLANK */ )
FROM STUDENTS, FACULTY
WHERE ( /* FILL IN THE BLANK */ );
```

```
OUTPUT>
+-----+-----+-----+-----+
| FIRST_NAME | LAST_NAME | FIRST_NAME | LAST_NAME |
+-----+-----+-----+-----+
| Alex       | Cars      | John       | Height    |
| Betty      | Dance     | John       | Height    |
| Charles    | Eagle     | John       | Height    |
| David      | Face      | Kay        | Time      |
| Elaine     | Gold      | Jane       | River     |
+-----+-----+-----+-----+
```

24. Repeat the same task as in Question 23 but change the table header names into aliases as shown in the following output.

```
SQL query>
SELECT (                /* FILL IN THE BLANK */                )
FROM STUDENTS, FACULTY
WHERE (                /* FILL IN THE BLANK */                );
```

OUTPUT>

STUDENT_FIRST_NAME	STUDENT_LAST_NAME	FACULTY_FIRST_NAME	FACULTY_LAST_NAME
Alex	Cars	John	Height
Betty	Dance	John	Height
Charles	Eagle	John	Height
David	Face	Kay	Time
Elaine	Gold	Jane	River

Now, assume you have a database named COURSES_DB2 and the following table entries. The table design is listed as below. Note that the table design is slightly different from that of COURSES_DB. In the STUDENTS table, CREDIT_REG denotes the number of the course credits the student registered for the current semester. Similarly, CREDIT_TEACH in FACULTY table shows the number of course credits that the faculty member teaches this semester.

```
CREATE TABLE STUDENTS
(
  STUDENT_ID INT PRIMARY KEY NOT NULL,
  LAST_NAME CHAR(20),
  FIRST_NAME CHAR(20),
  DEPT CHAR(15),
  CREDIT_REG DECIMAL(2, 0),
  FACULTY_ID INT
);
```

```
CREATE TABLE FACULTY
(
  FACULTY_ID INT PRIMARY KEY NOT NULL,
  LAST_NAME CHAR(20),
  FIRST_NAME CHAR(20),
  CREDIT_TEACH DECIMAL(2, 0),
  DEPT CHAR(15)
);
```

The table entries are as follows:

STUDENTS table:

STUDENT_ID	LAST_NAME	FIRST_NAME	DEPT	CREDIT_REG	FACULTY_ID
100	Cars	Alex	IST	15	1000
101	Dance	Betty	IST	18	1000
102	Eagle	Charles	CS	12	1000
103	Face	David	CHEM	16	1002
104	Gold	Elaine	BIO	9	1001
105	House	Flynn	NULL	10	1001

FACULTY table:

FACULTY_ID	LAST_NAME	FIRST_NAME	CREDIT_TEACH	DEPT
1000	Height	John	9	IST
1001	River	Jane	6	CS
1002	Time	Kay	3	CHEM

Given the task and the output from the MySQL command-line client, what will be the corresponding SQL query? Fill in the blank.

25. Task: List the last and first name of each student who registered for greater than 15 credits, together with the department, the number of credits registered, and the last name and first name of his/her advisor.

```
SQL query>
SELECT (                /* FILL IN THE BLANKS */                )
FROM STUDENTS S, FACULTY F
WHERE ( /* FILL IN THE BLANKS */ ) AND ( /* FILL IN THE BLANKS */ );
```

OUTPUT>

```
+-----+-----+-----+-----+-----+-----+
| LAST_NAME | FIRST_NAME | DEPT | CREDIT_REG | LAST_NAME | FIRST_NAME |
+-----+-----+-----+-----+-----+-----+
| Dance     | Betty      | IST  |          18 | Height    | John       |
| Face      | David      | CHEM |          16 | Time      | Kay        |
+-----+-----+-----+-----+-----+-----+
```

26. List the last and first name of a student who is not assigned any department. In the STUDENTS table, a NULL entry in the DEPT field is used when a student is not assigned any department yet.

```
SQL query>
SELECT LAST_NAME, FIRST_NAME
FROM STUDENTS
WHERE (                /* FILL IN THE BLANK */                );
```

OUTPUT>

```
+-----+-----+
| LAST_NAME | FIRST_NAME |
+-----+-----+
| House     | Flynn      |
+-----+-----+
```

27. List the last and first name of students either from the IST department, or his/her advisor is from the CHEM department, or both. (HINT: Consider a set operation.)

```
SQL query>
SELECT LAST_NAME, FIRST_NAME
FROM STUDENTS
WHERE (          /* FILL IN THE BLANK */          )
( /* FILL IN THE BLANK */ )
SELECT S.LAST_NAME, S.FIRST_NAME
FROM STUDENTS S, FACULTY F
WHERE S.FACULTY_ID = F.FACULTY_ID AND ( /* FILL IN THE BLANK */ );
```

OUTPUT>

```
+-----+-----+
| LAST_NAME | FIRST_NAME |
+-----+-----+
| Cars      | Alex       |
| Dance     | Betty      |
| Face      | David      |
+-----+-----+
```

28. List the last and first name of students who registered for greater than 11 credits and whose advisor is teaching more than 5 credits. (HINT: Consider how to work around INTERSECT set operation in MySQL.)

```
SQL query>
SELECT LAST_NAME, FIRST_NAME
FROM STUDENTS
WHERE (          /* FILL IN THE BLANK */          )
AND ( /* FILL IN THE BLANK */ )
(
  SELECT FACULTY_ID
  FROM FACULTY
  WHERE (          /* FILL IN THE BLANK */          )
);
```

OUTPUT>

```
+-----+-----+
| LAST_NAME | FIRST_NAME |
+-----+-----+
| Cars      | Alex       |
| Dance     | Betty      |
| Eagle     | Charles    |
+-----+-----+
```

29. List the department and the average credits of course registration for the students in the department.

```
SQL query>
SELECT ( /* FILL IN THE BLANK */ ) AS AVERAGE_CREDIT
FROM STUDENTS
( /* FILL IN THE BLANK */ );
```

OUTPUT>

```
+-----+-----+
| DEPT | AVERAGE_CREDIT |
+-----+-----+
| NULL |          10.0000 |
| BIO  |           9.0000 |
| CHEM |          16.0000 |
| CS   |          12.0000 |
| IST  |          16.5000 |
+-----+-----+
```

30. Find the last name and the first name of each student who registered for more credits than at least one IST student registered for.

```
SQL query>
SELECT LAST_NAME, FIRST_NAME FROM STUDENTS
WHERE ( /* FILL IN THE BLANK */ )
(
  SELECT CREDIT_REG
  FROM STUDENTS
  WHERE DEPT='IST'
);
```

OUTPUT>

```
+-----+-----+
| LAST_NAME | FIRST_NAME |
+-----+-----+
| Dance    | Betty     |
| Face     | David     |
+-----+-----+
```