

Please read Sections 5.1 – 5.1.3, 5.1.7 and 5.4 of the textbook and then answer the following, trying not to look at your notes or at the textbook. Quiz #3, on Wednesday 21st February, will consist of questions taken or inspired from Part I and Part II of this homework. The SQL code in this homework is slightly more advanced than what you'll be asked to do in the first exam, but make sure you understand it fully before taking the associated quiz.

Both Part I and Part II suppose that you have access to the snippets of code hosted at <http://spots.augusta.edu/caubert/teaching/2018/spring/csci3410/#snippets>, and that you created the HW_UNIV schema as we did in class.

Part I — Questions

1. Describe what the star do in the statement `SELECT ALL * FROM MYTABLE;`.
2. What is wrong with the statement `SELECT * WHERE Name = 'CS' FROM DEPARTMENT;`?
3. What is a select-project-join query?
4. What is a tuple variable, and when is it useful?
5. Write a query that changes the name of the professor whose login is 'caubert' to 'Hugo Pernet' in the table PROF.
6. Can an `UPDATE` statement have a `WHERE` condition using an attribute that isn't the primary key? If no, justify, if yes, tell what could happen.
7. What is a multi-set? What does it mean to say that SQL treats tables as multisets?
8. What is the difference between those two queries?

```
SELECT ALL * FROM MYTABLE;
```

and

```
SELECT DISTINCT * FROM MYTABLE;
```

How are the results the same? How are they different?

9. What are the possible meanings or interpretations for a `NULL` value?
10. What are the values of the following expressions (i.e., do they evaluate to `TRUE`, `FALSE`, or `UNKNOWN`)?
`TRUE AND FALSE` `TRUE AND UNKNOWN` `NOT UNKNOWN` `FALSE OR UNKNOWN`
11. What comparison expression should you use to test if a value is different from `NULL`?
12. Explain this query:

```
SELECT Login
FROM PROF
WHERE Department IN ( SELECT Major
                      FROM STUDENT
                      WHERE Login = 'jrakesh');
```

13. What is wrong with this query?

```
SELECT Name
FROM STUDENT
WHERE Login IN ( SELECT Code
                  FROM Department
                  WHERE head = 'aturing');
```

14. Write a query that returns the number of row (i.e., of entries, of tuples) in a table named BOOK.

15. Write a query that returns the sum of all the values stored in the Pages attribute of a BOOK table.

16. Write a query that adds a Pages attribute of type INT into a (already existing) BOOK table.

17. Write a query that removes the default value for a Pages attribute in a BOOK table.

Part II — Programming Exercises

This time, the (only) problem requires a computer and to have access to the code used during the lecture. I'll assume that you will have successfully completed it by the time Homework #4 is released (Wednesday 21st February), so don't wait and let me know if you had difficulties doing it.

Problem 1

Create the PROF, STUDENT, DEPARTMENT and GRADE tables as during the lecture. Populate them with some data.

1. Create a LECTURE table as follows:

- It should have four attributes, Name, Instructor, Code, and Year, of type VARCHAR(25) for the two first, YEAR(4), and CHAR(5).
- The Year and Code attributes should be the primary key (yes, have *two* attributes be the primary key).
- The Instructor attribute should be a foreign key referencing the Login attribute in PROF.

2. Populate the LECTURE table with some made-up data.

3. Add two columns to the GRADE table, using

```
ALTER TABLE GRADE
ADD COLUMN LectureCode CHAR(5),
ADD COLUMN LectureYear YEAR(4);
```

4. Use DESCRIBE and SELECT to observe the schema of the GRADE table and its rows (i.e., tuples). Is that what you would have expected?

5. Add a foreign key in GRADE, using

```
ALTER TABLE GRADE
ADD FOREIGN KEY (LectureYear, LectureCode)
REFERENCES LECTURE(Year, Code);
```

6. Update the tuples in GRADE with some made-up data: now, every row should contain, on top of a login and a grade, a lecture year and a lecture code.

7. Write **SELECT** statements answering the following questions (where XXX and YY should be relevant values considering your data):
- (a) "I taught class XXX in YYYY, could you give me the logins and grades of the students who took it?"
 - (b) "Could you list the instructors who taught in YYYY?" (and, please, avoid duplicates)
 - (c) "Could your list the name and grade of all the student who took class XXX (no matter the year)?"
 - (d) "Could you tell me which years was the class XXX taught?"
 - (e) "Could you list the classes taught the same year as class XXX?"
 - (f) "Could you print the name of the students who registered after Ava Alyx?"
 - (g) "How many departments' heads are teaching this year?"

