Data Base Design

Practice Test

Name: ___________________________

True/False

1. T F SQL aggregate functions can be used in the **GROUP BY** clause of **SELECT** statements.
2. T F **Inner Joins** are only used in the **WHERE** clause of **SELECT** statements.
3. T F The **DELETE** statement can be used to delete rows from several tables at once.
4. T F The SQL language contains DDL and DML statements.
5. T F The following SQL statement can be used to update all the rows in table employee:
   update employee set wages = wages * 1.1;

Use the following tables for the rest of the questions:

student(student-id, ssn, firstname, lastname, address, city, state, zip, phone)

takes(student-id, class-id, grade)

class(class-id, department, class-no, description, credit-hours, semester, year, section)

teaches(class-id, faculty-id)

faculty(faculty-id, firstname, lastname, address, city, state, zip, home-phone, office-phone, office)

6. Find the full names of all the students who live in the same cities as any of their teachers.
   a). select fullname from student, takes, class, teaches, faculty where student.city = faculty.city)
   b). select firstname, lastname from student natural join faculty
   c). select firstname, lastname from student as s
   where city in (select city from takes natural join class natural join teaches natural join faculty where s.student-id = takes.student-id)

7. Find the full names of all students who obtained a grade higher than *C* in class CMPSC (department) 4213 (class-no).
   a). select firstname, lastname from student natural join takes natural join class
   where (grade = 'A' or grade = 'B')
   and department = 'CMPSC' and class-no = 4213
   b). select firstname, lastname from student natural join class
   where (grade = 'A' or grade = 'B')
   and department = 'CMPSC' and class-no = 4213
   c). select firstname, lastname from takes
   where (grade = 'A' or grade = 'B')
   and department = 'CMPSC' and class-no = 4213

8. Find the description of the class with the smallest enrollment.
a). select min(enrollment) from takes
b). select description, sum(student-id) as enrollment from takes natural join class
    group by class.class-id
    having sum(student-id) <= all (select sum(student-id) from takes
                                                group by class-id)
c). select sum(student-id) as enrollment from takes
    group by class-id

9. Find the class-id with the highest enrollment.
a). select class-id, enrollment from class
    group by class-id
    having max(enrollment)
b). select class-id, sum(student-id) from takes
    group by class-id
    having sum(student-id) >= all (select sum(student-id) from takes
                                                    group by class-id)
c). select max(enrollment) from class-id
    where sum(student-id) = max(student-id)

10. Produce a list of all student last and first names and their grades (if any). Insure that the list contains all students whether they have a grade or not.
a). select lastname, firstname, grade from takes;
b). select lastname, firstname, (select grade from takes) as s_grade from student
c). select lastname, firstname, grade from student natural left outer join takes
d). select lastname, firstname, grade from student

11. Which SQL statement will find the full names of all students that got less than a *C* in any of their classes.
a). select firstname, lastname from takes where grade = 'D' or grade = 'F';
b). select distinct firstname, lastname from student natural join takes where grade = 'D'
    or grade = 'F';
c). select distinct firstname, lastname from student natural join takes where grade < 'C';
d). select firstname, lastname from takes where grade < 'C';

12. Which SQL statement will find the full names of all students that have taken all classes in the Comp. Sci. department?
a). select firstname, lastname from takes where department = 'Comp. Sci.'
b). select firstname, lastname from student natural join takes where department = 'Comp.Sci.'
c). select firstname, lastname from student as S where not exists ((select course-id from class
    where department = 'Comp. Sci.')
    except
    (select course-id from takes where S.student-id = takes.student-id))