Theodore Kim

JinZhao Su

CS-UY 3083: Introduction to Databases

Homework #3

1. What’s the most episodes any tv series had?

**SELECT MAX**(max\_eps) **FROM** tvseries

2. Find the TV series that has the most episodes. Results should show the TV series and the number of episodes

**SELECT** title, max\_eps

**FROM** tvseries

**WHERE** max\_eps = (**SELECT MAX**(max\_eps)

**FROM** tvseries)

3. Find the number of TV series made before the year 2000

**SELECT** **COUNT**(\*)

**FROM** tvseries

**WHERE** year < 2000

4. Find the first and last names of persons who favorite TV series is an old TV series. (series made before 2000)

**SELECT** first\_name, last\_name

**FROM** favorite

**WHERE** year < 2000

5. Find the first name and last name of each person who watches more than 5 TV series.

**SELECT** first\_name, last\_name

**FROM** watches

**GROUP BY** first\_name, last\_name

**HAVING COUNT**(\*) > 5

6. Find the number of students who’ve taken CS-101

**SELECT COUNT**(**DISTINCT** ID, course\_id)

**FROM** takes

**WHERE** course\_id = “CS-101”

7. Find the number of students who’ve gotten each grade in CS-101. The result should be a relation with attributes named grade and num.

**SELECT** grade, **COUNT**(\*) **AS** num

**FROM** takes

**GROUP BY** grade

8. Find the number of courses each students has taken. The result should be a relation with attributes named ID and num.

**SELECT** ID, **COUNT**(**DISTINCT** ID, course\_id) **AS** num

**FROM** takes

**GROUP BY** ID

9. Find the total number of credits each student has passed (grades other than F or NULL), based on the information in the takes table (not on the student.tot cred attribute)

**SELECT** t.ID, **SUM**(c.credits) **AS** passed\_cred

**FROM** takes **AS** t

**JOIN** course **AS** c **ON** c.course\_id = t.course\_id

**WHERE** t.grade **IS NOT NULL** **AND** t.grade != “F”

**GROUP BY** t.ID

10. Find IDs and names of students in the Comp Sci department who have taken fewer than two courses

**SELECT DISTINCT** s.ID, s.name

**FROM** student **AS** s

**JOIN** (**SELECT** ID, **COUNT**(\*) **AS** num\_courses **FROM** takes **GROUP BY** ID) **AS** t **ON** t.ID = s.ID

**WHERE** s.dept\_name = “Comp. Sci.” **AND** t.num\_courses < 2

11. Find the average number of credits of courses in each department. The result should be a relation with attributes dept\_name and avg\_credits

**SELECT** dept\_name, **AVG**(credits) **AS** avg\_credits

**FROM** course

**GROUP BY** dept\_name

12. Find the number of sections taught in each building, each year.

**SELECT** building, year, **COUNT**(\*) **AS** sections\_taught

**FROM** section

**GROUP BY** building, year

13. Find the IDs of students who’ve taken ‘CS-315’ and any course with the ’Bio’ prefix (LIKE ’Bio%’)

**SELECT DISTINCT** ID

**FROM** takes

**WHERE** course\_id=“CS-315” **AND** course\_id **LIKE** “Bio%”

14. Find the IDs of students who’ve taken ’CS-101’ but have not taken any other ’CS’ courses (LIKE ’CS%’)

**SELECT DISTINCT** t1.ID

**FROM** takes **AS** t1

**JOIN** (**SELECT** ID, **COUNT**(\*) **AS** cscount **FROM** takes **WHERE** course\_id **LIKE** “CS%” **GROUP BY** ID) **AS** t2

**ON** t1.ID = t2.ID

**WHERE** t1.course\_id = “CS-101” **AND** t2.cscount = 1

15. Find the maximum salary of an instructor in the Physics department

**SELECT MAX**(salary) **FROM** instructor **WHERE** dept\_name = “Physics”

16. Find the name of the highest paid instructor in the Comp Sci department. (If there is a tie, report all of them.)

**SELECT** name

**FROM** instructor **AS** t1

**JOIN** (**SELECT** dept\_name, **MAX**(salary) **AS** max\_salary

**FROM** instructor

**GROUP BY** dept\_name) **AS** t2

**ON** t2.dept\_name = t1.dept\_name **AND** t2.max\_salary = t1.salary

**WHERE** t1.dept\_name = “Comp. Sci.”

17. Find the name of the student who got the most ’A’ grades

**SELECT** student.name

**FROM** student

**JOIN** takes **ON** takes.ID = student.ID

**WHERE** takes.grade = “A”

**GROUP BY** student.name

**HAVING** COUNT(\*) = (**SELECT MAX**(num) **AS** max\_a

**FROM** (**SELECT** ID, **COUNT**(\*) **AS** num

**FROM** takes

**WHERE** grade = "A"

**GROUP BY** ID) **AS** t1)

18. Find the department with the highest average salary

**SELECT** dept\_name

**FROM** instructor

**GROUP BY** dept\_name

**HAVING AVG**(salary) **=** (**SELECT MAX**(avg\_salary)

**FROM** (**SELECT AVG**(salary) **AS** avg\_salary

**FROM** instructor

**GROUP BY** dept\_name) **AS** t1)