Bases de Données Fall, 2024

TD 3

Here are the tables we used in class:

course_id	*title*	*dept_name*	*credits*
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System Concepts	Comp. Sci.	3
EE-181	Intro. to Digital Systems	Elec. Eng.	3
FIN-201	Investment Banking	Finance	3
HIS-351	World History	History	3
MU-199	Music Video Production	Music	3
PHY-101	Physical Principles	Physics	4

(a) course

id	*name*	*dept_name*	*salary*
10101	Srinivasan	Comp. Sci.	65000.00
12121	Wu	Finance	90000.00
15151	Mozart	Music	40000.00
22222	Einstein	Physics	95000.00
32343	El Said	History	60000.00
33456	Gold	Physics	87000.00
45565	Katz	Comp. Sci.	75000.00
58583	Califieri	History	62000.00
76543	Singh	Finance	80000.00
76766	Crick	Biology	72000.00
83821	Brandt	Comp. Sci.	92000.00
98345	Kim	Elec. Eng.	80000.00

(b) teacher

id	*name*	*dept_name*	*tot_cred*
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

(c) student

course_id	*sec_id*	*semester*	*year*	*building*	*rn*	*time_id*
BIO-101	1	Summer	2009	Painter	514	B
BIO-301	1	Summer	2010	Painter	514	A
CS-101	1	Fall	2009	Packard	101	H
CS-101	1	Spring	2010	Packard	101	F
CS-190	1	Spring	2009	Taylor	3128	E
CS-190	2	Spring	2009	Taylor	3128	A
CS-315	1	Spring	2010	Watson	120	D
CS-319	1	Spring	2010	Watson	100	B
CS-319	2	Spring	2010	Taylor	3128	C
CS-347	1	Fall	2009	Taylor	3128	A
EE-181	1	Spring	2009	Taylor	3128	C
FIN-201	1	Spring	2010	Packard	101	B
HIS-351	1	Spring	2010	Painter	514	C
MU-199	1	Spring	2010	Packard	101	D
PHY-101	1	Fall	2009	Watson	100	A

(d) section

id	*course_id*	*sec_id*	*semester*	*year*
10101	CS-101	1	Fall	2009
10101	CS-315	1	Spring	2010
10101	CS-347	1	Fall	2009
12121	FIN-201	1	Spring	2010
15151	MU-199	1	Spring	2010
22222	PHY-101	1	Fall	2009
32343	HIS-351	1	Spring	2010
45565	CS-101	1	Spring	2010
45565	CS-319	1	Spring	2010
76766	BIO-101	1	Summer	2009
76766	BIO-301	1	Summer	2010
83821	CS-190	1	Spring	2009
83821	CS-190	2	Spring	2009
83821	CS-319	2	Spring	2010
98345	EE-181	1	Spring	2009

(e) teaches

id	*course_id*	*sec_id*	*semester*	*year*	*grade*
00128	CS-101	1	Fall	2009	A
00128	CS-347	1	Fall	2009	A-
12345	CS-101	1	Fall	2009	C
12345	CS-190	2	Spring	2009	A
12345	CS-315	1	Spring	2010	A
12345	CS-347	1	Fall	2009	A
19991	HIS-351	1	Spring	2010	B
23121	FIN-201	1	Spring	2010	C+
44553	PHY-101	1	Fall	2009	B-
45678	CS-101	1	Fall	2009	F
45678	CS-101	1	Spring	2010	B+
45678	CS-319	1	Spring	2010	B
54321	CS-101	1	Fall	2009	A-
54321	CS-190	2	Spring	2009	B+
55739	MU-199	1	Spring	2010	A-
76543	CS-101	1	Fall	2009	A
76543	CS-319	2	Spring	2010	A
76653	EE-181	1	Spring	2009	C
98765	CS-101	1	Fall	2009	C-
98765	CS-315	1	Spring	2010	B
98988	BIO-101	1	Summer	2009	A
98988	BIO-301	1	Summer	2010	

(f) takes

dept_name	*building*	*budget*
Biology	Watson	90000.00
Comp. Sci.	Taylor	100000.00
Elec. Eng.	Taylor	85000.00
Finance	Painter	120000.00
History	Painter	50000.00
Music	Packard	80000.00
Physics	Watson	70000.00

(g) department

1. Find the total enrollment of each course/section/semester/year that takes place in Spring 2010.

```
SELECT takes.course_id, takes.sec_id, takes.semester, takes.
   year, count(*)
FROM takes
WHERE takes.semester = 'Spring' and takes.year = 2010
GROUP BY takes.course_id, takes.sec_id, takes.semester, takes.
   year;
```

course_id	*sec_id*	*semester*	*year*	*count*
CS-101	1	Spring	2010	1
CS-315	1	Spring	2010	2
CS-319	1	Spring	2010	1
CS-319	2	Spring	2010	1
FIN-201	1	Spring	2010	1
HIS-351	1	Spring	2010	1
MU-199	1	Spring	2010	1

2. Compute the total credits over all taken courses for each year.

```
SELECT year, SUM(credits)
FROM takes, course
WHERE takes.course_id = course.course_id
GROUP BY year;
```

year	*sum*
2010	29
2009	49

3. For each department, find the total number of times a course was taught in that departments building by that departments teachers.

```
SELECT department.dept_name, count(*)
FROM section, department, teacher, teaches
WHERE (section.course_id, section.sec_id, section.semester,
   section.year) = (teaches.course_id, teaches.sec_id, teaches.
   semester, teaches.year) and
        teaches.id = teacher.id and teacher.dept_name =
   department.dept_name and department.building = section.
   building
GROUP BY department.dept_name;
```

dept_name	*count*
Comp. Sci.	4
Elec. Eng.	1
History	1
Music	1
Physics	1

4. For each of the three semesters (Summer, Fall, Spring), output the total number of courses taught in that semester over all the years.

```
SELECT section.semester, count(*)
FROM section
GROUP BY section.semester
```

semester	*count*
Summer	2
Spring	10
Fall	3

5. Output the name of each student and the total number of credits taken in courses that are *not* from his department.

```
SELECT student.name, sum(credits)
FROM student, course, takes
WHERE student.id = takes.id and takes.course_id = course.
   course_id and student.dept_name != course.dept_name
GROUP BY student.name;
```

name	*sum*
Bourikas	7
Levy	11

6. What will be the output of the following SQL query:

```
SELECT section.building, sum(course.credits)
FROM section, course
WHERE section.course_id = course.course_id
GROUP BY section.building;
```

The total number of credits from courses that took place in each building.

building	*sum*
Taylor	17
Packard	14
Painter	11
Watson	10