

[COSE371 Database]

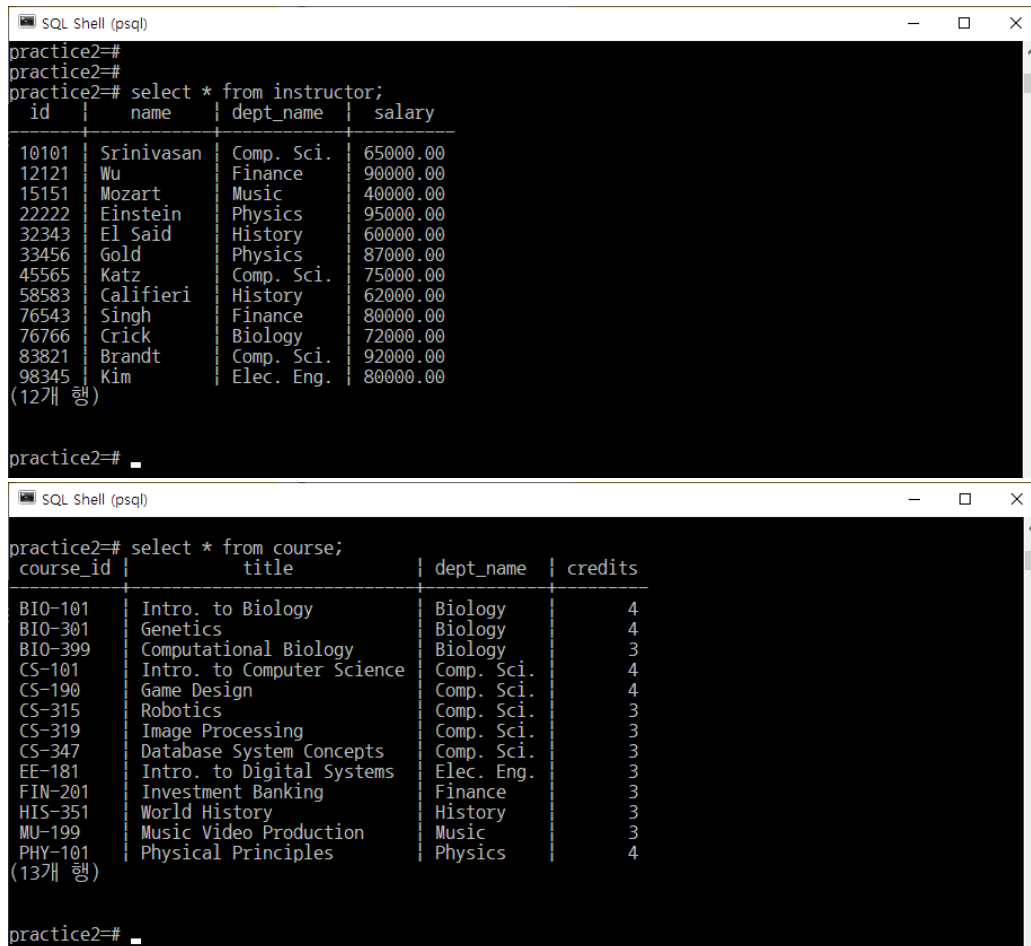
Lab – SQL 2

2019320137

컴퓨터학과 황상민

Exercise 1.

- a. Execute 'select \* from instructor;' and 'select \* from course;'



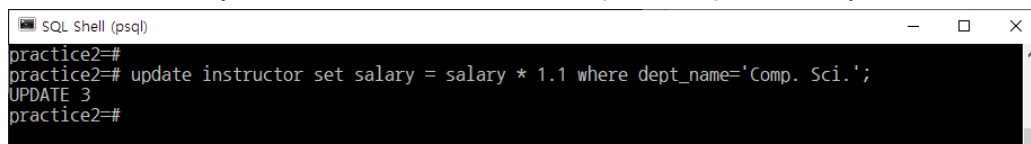
The first screenshot shows the result of the query 'select \* from instructor;'. It displays a table with 4 columns: id, name, dept\_name, and salary. There are 12 rows of data, and the output indicates '(12개 행)' (12 rows).

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

The second screenshot shows the result of the query 'select \* from course;'. It displays a table with 4 columns: course\_id, title, dept\_name, and credits. There are 13 rows of data, and the output indicates '(13개 행)' (13 rows).

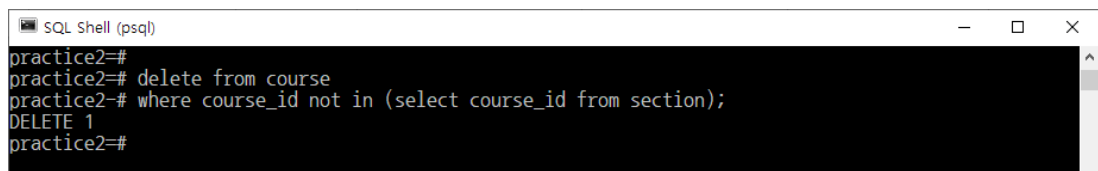
course_id	title	dept_name	credits
BIO-101	Intro. to Biology	Biology	4
BIO-301	Genetics	Biology	4
BIO-399	Computational Biology	Biology	3
CS-101	Intro. to Computer Science	Comp. Sci.	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3
CS-319	Image Processing	Comp. Sci.	3
CS-347	Database System 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

- b. Increase the salary of each instructor in the Comp. Sci. department by 10%.



The screenshot shows the execution of the SQL statement: 'update instructor set salary = salary \* 1.1 where dept\_name='Comp. Sci.';'. The output indicates 'UPDATE 3', meaning 3 rows were updated.

- c. Delete all courses that have never been offered (i.e., do not occur in the section relation).



The screenshot shows the execution of the SQL statement: 'delete from course where course\_id not in (select course\_id from section);'. The output indicates 'DELETE 1', meaning 1 row was deleted.

- d. Insert every student whose tot\_cred attribute is greater than 100 as an instructor in the same department, with a salary of \$10,000.

```
SQL Shell (psql)
practice2=#
practice2=# insert into instructor (
practice2(# select id, name, dept_name, 10000 from student
practice2(# where tot_cred > 100
practice2(# );
INSERT 0 3
practice2=#
```

- e. Execute 'select \* from instructor;' and 'select \* from course;'

```
SQL Shell (psql)
practice2=#
practice2=# select * from instructor;
 id      | name      | dept_name | salary
-----|-----|-----|-----
 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
 58583   | Califieri | History   | 62000.00
 76543   | Singh    | Finance   | 80000.00
 76766   | Crick    | Biology   | 72000.00
 98345   | Kim      | Elec. Eng. | 80000.00
 10101   | Srinivasan | Comp. Sci. | 71500.00
 45565   | Katz     | Comp. Sci. | 82500.00
 83821   | Brandt   | Comp. Sci. | 101200.00
 00128   | Zhang    | Comp. Sci. | 10000.00
 23121   | Chavez   | Finance   | 10000.00
 98988   | Tanaka   | Biology   | 10000.00
(15개 행)

practice2=#
```

```
SQL Shell (psql)
practice2=#
practice2=# select * from course;
 course_id | title                                | dept_name | credits
-----|-----|-----|-----
 BIO-101   | Intro. to Biology                   | Biology   | 4
 BIO-301   | Genetics                            | Biology   | 4
 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
(12개 행)

practice2=#
```

## Exercise 2.

- a. Find the ID, name, city, and street of each employee who works for "First Bank Corporation".

```
SQL Shell (psql)
practice2=#
practice2=# select employee.id, person_name, city, street from employee, works
practice2=# where employee.id=works.id and company_name='First Bank Corporation';
 id | person_name | city      | street
-----+-----+-----+-----
12345 | Adams       | Pittsfield | Spring
22222 | Curry       | Rye        | North
77777 | Jones       | Harrison   | Main
(37 행)
```

- b. Find the ID, name, city, and street of each employee who works for "First Bank Corporation" and earns more than \$10000.

```
SQL Shell (psql)
practice2=# select employee.id, person_name, city, street from employee, works
practice2=# where company_name='First Bank Corporation' and salary>10000
practice2=# and employee.id=works.id;
 id | person_name | city      | street
-----+-----+-----+-----
12345 | Adams       | Pittsfield | Spring
22222 | Curry       | Rye        | North
(27 행)
```

- c. Find the ID of each employee who does not work for "First Bank Corporation".

```
SQL Shell (psql)
practice2=# select id from works where company_name <> 'First Bank Corporation';
 id
-----
11111
33333
44444
55555
66666
88888
99999
00000
54321
(97 행)
```

- d. Find the ID of each employee who earns more than every employee of "Small Bank Corporation".

```
SQL Shell (psql)
practice2=#
practice2=# with max_salary(value) as
practice2=# (select max(salary) from works
practice2=# where company_name='Small Bank Corporation')
practice2=# select employee.id from employee, works, max_salary
practice2=# where employee.id=works.id and works.salary > max_salary.value;
   id
-----
11111
22222
33333
44444
55555
99999
54321
(7개 행)

practice2=#
```

- e. Find the name of each company that is in the city of "Small Bank Corporation" is located.

```
SQL Shell (psql)
practice2=#
practice2=# select company_name from company
practice2=# where city=(select city from company where company_name='Small Bank Corporation');
   company_name
-----
First Bank Corporation
Small Bank Corporation
(2개 행)

practice2=#
```

- f. Find the name of the company that has the most employees (or companies, in the case where there is a tie for the most).

```
SQL Shell (psql)
practice2=#
practice2=# with company_worker_count(company_name, count) as
practice2=# (select company_name, count(*) from works group by company_name)
practice2=# select company_name from company_worker_count
practice2=# where count=(select max(count) from company_worker_count);
   company_name
-----
Big Bank Corporation
(1개 행)
```

- g. Find the name of each company whose employees earn a higher salary, on average, than the average salary at "First Bank Corporation".

```
SQL Shell (psql)
practice2=#
practice2=# with company_avg_salary(company_name, avg_salary) as
practice2=# (select company_name, avg(salary) from works group by company_name)
practice2=# select company_name from company_avg_salary
practice2=# where avg_salary > (
practice2=# select avg_salary from company_avg_salary
practice2=# where company_name='First Bank Corporation'
practice2=# );
   company_name
-----
Second Bank Corporation
Big Bank Corporation
(2개 행)

practice2=#
```