SQL Exercises

Javier Ortín Rodenas

May 17, 2025

1 Database system concepts chapter 3

1.1 Exercise

Find the name of all instructors with a higher salary than all the instructors in the 'Biology' department.

```
SELECT DISTINCT name
FROM instructor
WHERE salary > all(
    select distinct salary
    from instructor
    where dept_name = 'Biology');
```

1.2 Exercise

Find all courses taught in both the Fall 2017 semester and in the Spring 2018 semester.

```
SELECT course_id
FROM section s
WHERE semester = 'Fall' and year = 2017
and EXISTS (
    select course_id from section
    where semester = 'Spring' and year = 2018
    and course_id = s.course_id);
```

1.3 Exercise

Find all students who have taken all courses offered in the Biology department.

```
SELECT *
FROM student s
WHERE NOT EXISTS (
    SELECT course_id
    FROM course c
    WHERE dept_name = 'Biology'
    and NOT EXISTS (
        SELECT *
        FROM takes
        WHERE student_id = s.student_id
        and course_id = c.course_id
    )
)
```

We could write an equivalent query as follows:

```
SELECT *
FROM student s
WHERE NOT EXISTS (
    (SELECT course_id
    FROM course c
    WHERE dept_name = ’Biology’)
    EXCEPT
    (SELECT course_id
    FROM takes
    WHERE student_id = s.student_id)
)
```

This query will return the same result as the one above it.

1.4 Exercise

Find all courses that were offered at most once in 2017.

```
SELECT *
FROM course c
WHERE UNIQUE(
    SELECT course_id
    FROM section
    WHERE year = 2017 and course_id = c.course_id
);
```

1.5 Exercise

Find the average instructors’ salaries of those departments where the average salary is greater than $42,000.

```
SELECT AVG(salary) as avg_sal
FROM instructor
WHERE dept_name in (
    SELECT distinct dept_name
    FROM instructor
    GROUP BY dept_name
    HAVING AVG(salary) > 42000
);
```

We must find out which departments have an average salary greater than $42.000 and then calculate the average salary of the instructors from those departments.

1.6 Exercise

Find all departments where the total salary is greater than the average of the total salary at all departments.

```
SELECT dept_name, SUM(salary) AS total_salary
FROM instructor
GROUP BY dept_name
HAVING SUM(salary) > (
    SELECT AVG(dept_total)
    FROM (
        SELECT SUM(salary) AS dept_total
        FROM instructor
        GROUP BY dept_name
    ) AS dept_totals
);
```

1.7 Exercise

List all departments along with the number of instructors in each department

```
SELECT dept_name, COUNT(*) as n_instructors
FROM instructor
GROUP BY dept_name;
```

2 SQL 2019 Exam

2.1 Exercise

Cambiar todas las medidas con estado ”incompleta” de los sensores del módulo ”MA” tomadas en la calle Uría después de las 17:00, para que estado sea ”ok”.

```
UPDATE MEDIDAS
SET estado = 'ok'
WHERE id_ubicacion in (select id_ubicacion
    from ubicacion where calle = 'Uria')
and id_modulo in (select id_modulo
    from modulo where nombre_modulo = 'MA')
and estado = 'incompleta'
and hora > '17:00';
```

2.2 Exercise

Identificación y descripción de los sensores que tengan el mayor número de medidas realisades dentiro de cada tipo de sensores (imprimir también el tipo).

```
SELECT id_modulo, id_sensor, descripcion_sensor, tipo
FROM medida as m NATURAL JOIN sensor as s
GROUP BY m.id_modulo, m.id_sensor, s.descripcion_sensor, s.tipo
HAVING COUNT(*) >= ALL (
    SELECT COUNT(*)
    FROM medida NATURAL JOIN sensor as s2
    WHERE s2.tipo = s.tipo
    GROUP BY id_modulo, id_sensor
);
```

2.3 Exercise

Calles en las que nunca se ha realizado una medida de tipo”temperatura” con resultads ”fallo”. También hacer en álgebre relacional

```
(SELECT calle
FROM ubicacion)
EXCEPT
(SELECT calle
FROM ubicacion
WHERE id_ubicacion in(
    SELECT id_ubicacion
    FROM medida as m NATURAL JOIN sensor as s
    WHERE m.estado = "fallo"
    and s.tipo = "temperatura"
));
```

Veamos cómoo hacer la consulta en álgebra relacional:

$$\Pi_{\text{calle}}\,(\text{ubicacion})$$

$$-$$

$$\Pi_{\text{calle}}\Big(\sigma_{\text{estado}\,=\,\text{"fallo"}}\,(\text{medida})\;\bowtie\;\sigma_{\text{tipo}\,=\,\text{"temperatura"}}\,(\text{sensor})\;\bowtie\;\text{ubicacion}\Big)$$

2.4 Exercise

Identificación del módulo y nombre del responsable de aquellos módulos tales que todas las medidas que han realizado sus sensores del tipo "humedad" han tenido valor mayor a 100.

```
SELECT id_modulo, nombre_responsable
FROM modulo mod
WHERE NOT EXISTS (
    SELECT *
    FROM sensor as s NATURAL JOIN medida as med
    WHERE s.id_modulo = mod.id_modulo
    and s.tipo = 'humedad'
    and med.valor <= 100
);
```