

## **Consultas sql universidad**

Jhuliana Andrea González Torres

Servicios Nacional de Aprendizaje (SENA), Análisis y desarrollo de Sistemas de  
Información

2451627: Select Base de Datos Universidad

**Instructora.** Heidy Lizbeth Adarme Romero

23 de junio del 2022

## SINTAXIS EN SQL A DESARROLLAR

Realice las respectivas consultas sobre las tablas utilizando la herramienta POSTGRESQL, a partir de los siguientes enunciados:

### Consultas sobre una tabla

- Devuelve un listado con el primer apellido, segundo apellido y el nombre de todos los alumnos. El listado deberá estar ordenado alfabéticamente de menor a mayor por el primer apellido, segundo apellido y nombre.

Query    Query History

```
1 SELECT apellido1, apellido2, nombre FROM persona
2 WHERE tipo = 'alumno'
3 ORDER BY apellido1 ASC, apellido2 ASC, nombre ASC;
```

Data Output    Messages    Notifications

|    | apellido1<br>character varying (50) | apellido2<br>character varying (50) | nombre<br>character varying (25) |
|----|-------------------------------------|-------------------------------------|----------------------------------|
| 1  | Domínguez                           | Guerrero                            | Antonio                          |
| 2  | Gea                                 | Ruiz                                | Sonia                            |
| 3  | Gutiérrez                           | López                               | Juan                             |
| 4  | Heller                              | Pagac                               | Pedro                            |
| 5  | Herman                              | Pacocha                             | Daniel                           |
| 6  | Hernández                           | Martínez                            | Irene                            |
| 7  | Herzog                              | Tremblay                            | Ramón                            |
| 8  | Koss                                | Bayer                               | José                             |
| 9  | Lakin                               | Yundt                               | Inma                             |
| 10 | Saez                                | Vega                                | Juan                             |
| 11 | Sánchez                             | Pérez                               | Salvador                         |
| 12 | Strosin                             | Turcotte                            | Ismael                           |

Total rows: 12 of 12    Query complete 00:00:00.068

- Averigua el nombre y los dos apellidos de los alumnos que no han dado de alta su número de teléfono en la base de datos.

```
1 select nombre, apellido1, apellido2 FROM persona
2 WHERE tipo = 'alumno' AND telefono IS NULL;
```

Data Output    Messages    Notifications

|   | nombre<br>character varying (25) | apellido1<br>character varying (50) | apellido2<br>character varying (50) |
|---|----------------------------------|-------------------------------------|-------------------------------------|
| 1 | Pedro                            | Heller                              | Pagac                               |
| 2 | Ismael                           | Strosin                             | Turcotte                            |

- Devuelve el listado de los alumnos que nacieron en 1999.

Query    Query History

```

1: SELECT * FROM persona
2: WHERE EXTRACT(YEAR FROM fecha_nacimiento) = '1999' AND tipo = 'alumno';

```

Data Output    Messages    Notifications

|   | <b>id</b><br>[PK] integer | <b>nif</b><br>character varying (9) | <b>nombre</b><br>character varying (25) | <b>apellido1</b><br>character varying (50) | <b>apellido2</b><br>character varying (50) | <b>ciudad</b><br>character varying (25) |
|---|---------------------------|-------------------------------------|---|--|--|---|
| 1 |                           | 7 97258166K                         | Ismael                                  | Strosin                                    | Turcotte                                   | Almería                                 |
| 2 |                           | 22 41491230N                        | Antonio                                 | Domínguez                                  | Guerrero                                   | Almería                                 |

4. Devuelve el listado de profesores que no han dado de alta su número de teléfono en la base de datos y además su nif termina en K.

```

1: SELECT * FROM persona
2: WHERE telefono IS NULL AND RIGHT(nif, 1) = 'K'
3: AND tipo = 'profesor';

```

Data Output    Messages    Notifications

|   | <b>id</b><br>[PK] integer | <b>nif</b><br>character varying (9) | <b>nombre</b><br>character varying (25) | <b>apellido1</b><br>character varying (50) | <b>apellido2</b><br>character varying (50) | <b>ciudad</b><br>character varying (25) |
|---|---------------------------|-------------------------------------|---|--|--|---|
| 1 |                           | 16 10485008K                        | Antonio                                 | Fahey                                      | Considine                                  | Almería                                 |
| 2 |                           | 17 85869555K                        | Guillermo                               | Ruecker                                    | Upton                                      | Almería                                 |

5. Devuelve el listado de las asignaturas que se imparten en el primer cuatrimestre, en el tercer curso del grado que tiene el identificador 7.

```

1: SELECT * FROM asignatura
2: WHERE cuatrimestre = 1 AND curso = 3 AND id_grado = 7;

```

Data Output    Messages    Notifications

|   | <b>id</b><br>[PK] integer | <b>nombre</b><br>character varying (100)  | <b>creditos</b><br>double precision | <b>tipo</b><br>character varying (20) | <b>curso</b><br>smallint | <b>cuatrimestre</b><br>smallint | <b>id_grado</b><br>integer |
|---|---------------------------|---|-------------------------------------|---------------------------------------|--------------------------|---------------------------------|----------------------------|
| 1 | 72                        | Bases moleculares del desarrollo vegetal  | 4.5                                 | obligatoria                           | 3                        | 1                               |                            |
| 2 | 73                        | Fisiología animal                         | 4.5                                 | obligatoria                           | 3                        | 1                               |                            |
| 3 | 74                        | Metabolismo y biosíntesis de biomoléculas | 6                                   | obligatoria                           | 3                        | 1                               |                            |
| 4 | 75                        | Operaciones de separación                 | 6                                   | obligatoria                           | 3                        | 1                               |                            |
| 5 | 76                        | Patología molecular de plantas            | 4.5                                 | obligatoria                           | 3                        | 1                               |                            |
| 6 | 77                        | Técnicas instrumentales básicas           | 4.5                                 | obligatoria                           | 3                        | 1                               |                            |

## Consultas multitable (Composición interna)

1. Devuelve un listado con los datos de todas las alumnas que se han matriculado alguna vez en el Grado en Ingeniería Informática (Plan 2015).

```

SELECT DISTINCT per.* FROM persona AS per INNER JOIN alumno_se_matricula_asignatura
AS alum_asig ON per.id = alum_asig.id_alumno INNER JOIN asignatura ON
alum_asig.id_asignatura = asignatura.id INNER JOIN grado ON asignatura.id_grado =
grado.id INNER JOIN curso_escolar AS curso ON alum_asig.id_curso_escolar = curso.id
WHERE per.tipo = 'alumno' AND grado.nombre = 'Grado en Ingeniería Informática (Plan 2015)'
AND per.Sexo = 'M'; AND per.Sexo = 'M'; AND per.Sexo = 'M';

```

|   | id | nif       | nombre | apellido1 | apellido2 | ciudad  |
|---|----|-----------|--------|-----------|-----------|---------|
| 1 | 19 | 11575262G | Imma   | Lakin     | Yundt     | Almeria |
| 2 | 23 | 64753215G | Irene  | Hernández | Martinez  | Almeria |
| 3 | 24 | 85135690V | Stenia | Gea       | Ruiz      | Almeria |

2. Devuelve un listado con todas las asignaturas ofertadas en el Grado en Ingeniería Informática (Plan 2015).

**SELECT asignatura.\***

**FROM asignatura**

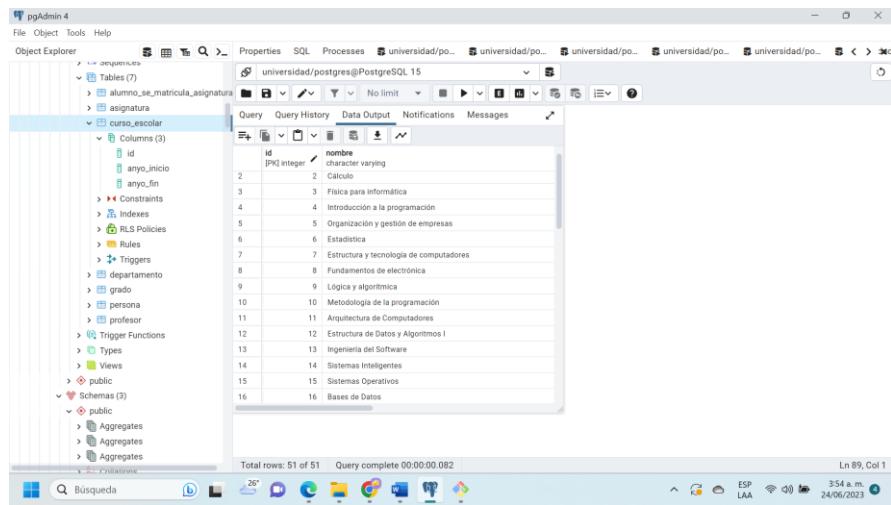
**INNER JOIN grado ON asignatura.id\_grado = grado.id**

**WHERE grado.nombre = 'Grado en Ingeniería Informática (Plan 2015)';**

```

81 SELECT DISTINCT dep.id, dep.nombre AS "Nombre del departamento"
82 FROM departamento AS dep
83 LEFT JOIN profesor AS prof ON dep.id = prof.id_departamento
84 LEFT JOIN asignatura AS asig ON prof.id_profesor = asig.id_profesor
85 LEFT JOIN alumno_se_matricula_asignatura AS alum_asig ON asig.id = alum_asig.id_asignatura LEFT JOIN curso_escolar AS
86 WHERE asig.id IS NULL;
87
88
89 SELECT asignatura.*
90 FROM asignatura
91 INNER JOIN grado ON asignatura.id_grado = grado.id
92 WHERE grado.nombre = 'Grado en Ingeniería Informática (Plan 2015)';
93
94
95
96 SELECT A.id, A.nombre
97 FROM asignatura AS A
98 JOIN grado AS G ON A.id_grado = G.id
99 WHERE G.nombre = 'Ingeniería Informática' AND A.tipo = 'ofertada';
100
101 SELECT COUNT(*) AS alumnas FROM persona
102 WHERE tipo = 'alumno' AND sexo = 'M';

```



3. Devuelve un listado de los profesores junto con el nombre del departamento al que están vinculados. El listado debe devolver cuatro columnas, primer apellido, segundo apellido, nombre y nombre del departamento. El resultado estará ordenado alfabéticamente de menor a mayor por los apellidos y el nombre.

```

SELECT P.apellido1, P.apellido2, P.nombre, D.nombre AS nombre_departamento
FROM profesor AS PR
JOIN persona AS P ON PR.id_profesor = P.id
LEFT JOIN departamento AS D ON PR.id_departamento = D.id
ORDER BY P.apellido1, P.apellido2, P.nombre;

```

The screenshot shows the pgAdmin 4 interface with a database connection to 'universidad/postgres@PostgreSQL 15\*'. The Object Explorer on the left lists various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables (with 7 entries), Triggers, Types, Views, Schemas (with 3 entries), and public objects. The central pane displays a table with 12 rows of data. The table has four columns: apellido1, apellido2, nombre, and nombre\_departamento. The data is as follows:

|    | apellido1  | apellido2  | nombre    | nombre_departamento |
|----|------------|------------|-----------|---------------------|
| 1  | Fahey      | Considine  | Antonio   | Economía y Empresa  |
| 2  | Hamill     | Kozey      | Mario     | Informática         |
| 3  | Kohler     | Schoen     | Alejandro | Matemáticas         |
| 4  | Lemke      | Rutherford | Cristina  | Economía y Empresa  |
| 5  | Monahan    | Murray     | Micaela   | Agronomía           |
| 6  | Ramirez    | Gea        | Zoe       | Informática         |
| 7  | Ruecker    | Upton      | Guillermo | Educación           |
| 8  | Schmidt    | Fisher     | David     | Matemáticas         |
| 9  | Schowalter | Muller     | Francesca | Química y Física    |
| 10 | Spencer    | Lakin      | Esther    | Educación           |
| 11 | Stiedemann | Morissette | Alfredo   | Química y Física    |
| 12 | Streich    | Hirth      | Carmen    | Educación           |

4. Devuelve un listado con el nombre de las asignaturas, año de inicio y año de fin del curso escolar del alumno con nif 26902806M.

```

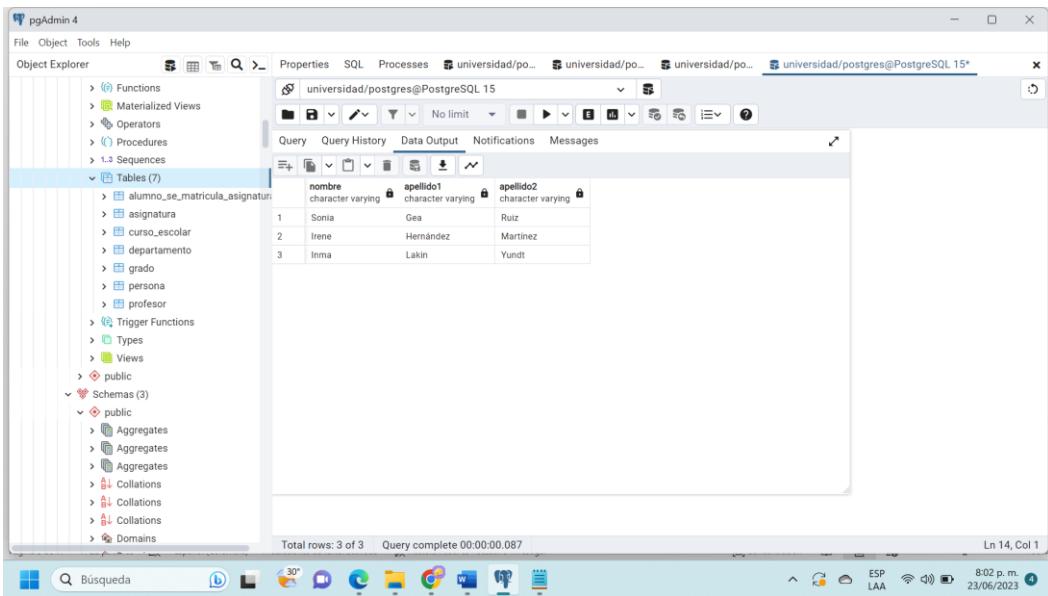
SELECT A.nombre AS nombre_asignatura, CE.anio_inicio, CE.anio_fin
FROM persona AS P
JOIN alumno_se_matricula_asignatura AS AMA ON P.id = AMA.id_alumno
JOIN asignatura AS A ON AMA.id_asignatura = A.id
JOIN curso_escolar AS CE ON AMA.id_curso_escolar = CE.id
WHERE P.nif = '26902806M';
    
```

|   | nombre_asignatura                    | anyo_inicio | anyo_fin |
|---|--------------------------------------|-------------|----------|
| 1 | Álgebra lineal y matemática discreta | 2014        | 2015     |
| 2 | Cálculo                              | 2014        | 2015     |
| 3 | Física para informática              | 2014        | 2015     |

6. Devuelve un listado con el nombre de todos los departamentos que tienen profesores que imparten alguna asignatura en el Grado en Ingeniería Informática (Plan 2015). 6. Devuelve un listado con todos los alumnos que se han matriculado en alguna asignatura durante el curso escolar 2018/2019.

```

SELECT DISTINCT P.nombre, P.apellido1, P.apellido2
FROM persona AS P
JOIN alumno_se_matricula_asignatura AS AMA ON P.id = AMA.id_alumno
JOIN curso_escolar AS CE ON AMA.id_curso_escolar = CE.id
WHERE CE.anyo_inicio = 2018 AND CE.anyo_fin = 2019;
    
```



## Consultas multitabla (Composición externa)

**Resuelva todas las consultas utilizando las cláusulas LEFT JOIN y RIGHT JOIN.**

- Devuelve un listado con los nombres de todos los profesores y los departamentos que tienen vinculados. El listado también debe mostrar aquellos profesores que no tienen ningún departamento asociado. El listado debe devolver cuatro columnas, nombre del departamento, primer apellido, segundo apellido y nombre del profesor. El resultado estará ordenado alfabéticamente de menor a mayor por el nombre del departamento, apellidos y el nombre.

```
SELECT D.nombre AS nombre_departamento, P.apellido1, P.apellido2, P.nombre AS nombre_profesor
```

```
FROM persona AS P
```

```
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
```

```
LEFT JOIN departamento AS D ON PR.id_departamento = D.id
```

```
ORDER BY D.nombre, P.apellido1, P.apellido2, P.nombre;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the Object Explorer with various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, and Tables. The Tables section shows seven tables: alumno\_se\_matricula\_asignatura, asignatura, curso\_escolar, departamento, grado, persona, and profesor. The main window contains a query editor with the following SQL code:

```

SELECT nombre_departamento, apellido1, apellido2, nombre_profesor
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
LEFT JOIN departamento AS D ON PR.id_departamento = D.id

```

The results grid shows the following data:

|    | nombre_departamento | apellido1  | apellido2  | nombre_profesor |
|----|---------------------|------------|------------|-----------------|
| 1  | Agronomía           | Monahan    | Murray     | Micaela         |
| 2  | Economía y Empresa  | Fahey      | Consideine | Antonio         |
| 3  | Economía y Empresa  | Lemke      | Rutherford | Cristina        |
| 4  | Educación           | Ruecker    | Upton      | Guillermo       |
| 5  | Educación           | Spencer    | Lakin      | Esther          |
| 6  | Educación           | Streich    | Hirth      | Carmen          |
| 7  | Informática         | Hamill     | Kozey      | Manolo          |
| 8  | Informática         | Ramirez    | Gea        | Zoe             |
| 9  | Matemáticas         | Kohler     | Schoen     | Alejandro       |
| 10 | Matemáticas         | Schmidt    | Fisher     | David           |
| 11 | Química y Física    | Schowalter | Muller     | Francesca       |
| 12 | Química y Física    | Stiedemann | Morissette | Alfredo         |
| 13 | [null]              | Dominguez  | Guerrero   | Antonio         |
| 14 | [null]              | Gea        | Ruiz       | Sonia           |
| 15 | [null]              | Gutiérrez  | López      | Juan            |

Below the results, a message bar indicates: "Successfully run. Total query runtime: 209 msec. 24 rows affected." and "Ln 20, Col 1".

2. Devuelve un listado con los profesores que no están asociados a un departamento.

```

SELECT D.nombre AS nombre_departamento, P.apellido1, P.apellido2, P.nombre AS
nombre_profesor
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
LEFT JOIN departamento AS D ON PR.id_departamento = D.id

```

ORDER BY D.nombre, P.apellido1, P.apellido2, P.nombre;

|    | nombre_departamento | apellido1  | apellido2  | nombre_profesor |
|----|---------------------|------------|------------|-----------------|
| 1  | Agronomia           | Monahan    | Murray     | Micaela         |
| 2  | Economía y Empresa  | Fahey      | Considine  | Antonio         |
| 3  | Economía y Empresa  | Lemke      | Rutherford | Cristina        |
| 4  | Educación           | Ruecker    | Upton      | Guillermo       |
| 5  | Educación           | Spencer    | Lakin      | Esther          |
| 6  | Educación           | Streich    | Hirthe     | Carmen          |
| 7  | Informática         | Hamill     | Kozev      | Manolo          |
| 8  | Informática         | Ramirez    | Gea        | Zoe             |
| 9  | Matemáticas         | Kohler     | Schoen     | Alejandro       |
| 10 | Matemáticas         | Schmidt    | Fisher     | David           |
| 11 | Química y Física    | Schowalter | Muller     | Francesca       |
| 12 | Química y Física    | Stiedemann | Morisette  | Alfredo         |
| 13 | [null]              | Dominguez  | Guerrero   | Antonio         |
| 14 | [null]              | Gea        | Ruiz       | Sonia           |
| 15 | [null]              | Gutiérrez  | López      | Juan            |

Total rows: 24 of 24 Query complete 00:00:00.209 Ln 20, Col 1

3. Devuelve un listado con los departamentos que no tienen profesores asociados.

```
SELECT D.nombre AS nombre_departamento
FROM departamento AS D
LEFT JOIN profesor AS PR ON D.id = PR.id_departamento
WHERE PR.id_profesor IS NULL;
```

|   |                     |
|---|---------------------|
| 1 | Derecho             |
| 2 | Biología y Geología |
| 3 | Filología           |

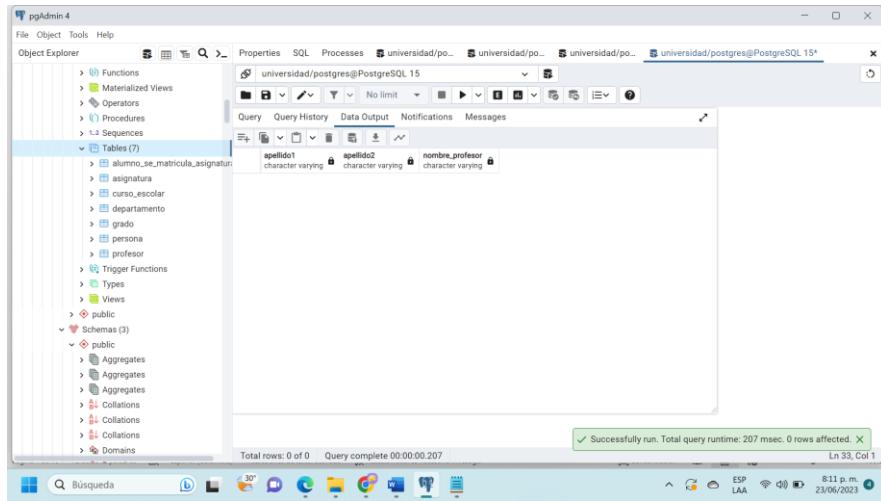
Successfully run. Total query runtime: 178 msec. 3 rows affected. Total rows: 3 of 3 Query complete 00:00:00.178 Ln 28, Col 1

4. Devuelve un listado con los profesores que no imparten ninguna asignatura.

```

SELECT P.apellido1, P.apellido2, P.nombre AS nombre_profesor
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
WHERE PR.id_profesor IS NOT NULL
AND PR.id_profesor NOT IN (SELECT DISTINCT id_profesor FROM asignatura);

```



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables (7), Triggers, Types, and Views. Under Tables, there are entries for alumno\_se\_matricula\_asignatura, asignatura, curso\_escolar, departamento, grado, persona, and profesor. On the right, a query editor window is open with the following SQL code:

```

SELECT P.apellido1, P.apellido2, P.nombre AS nombre_profesor
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
WHERE PR.id_profesor IS NOT NULL
AND PR.id_profesor NOT IN (SELECT DISTINCT id_profesor FROM asignatura);

```

The results pane shows three columns: apellido1, apellido2, and nombre\_profesor. Below the results, a status bar indicates "Successfully run. Total query runtime: 207 msec. 0 rows affected." The system tray at the bottom shows the date and time as 23/06/2023, 8:11 p.m.

5. Devuelve un listado con las asignaturas que no tienen un profesor asignado.

```

SELECT A.nombre AS nombre_asignatura
FROM asignatura AS A
LEFT JOIN profesor AS P ON A.id_profesor = P.id_profesor
WHERE A.id_profesor IS NULL;

```

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the Object Explorer with various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, and Tables. The 'Tables (7)' section is expanded, showing the 'asignatura' table. The main window contains a query results grid with 15 rows of data. A status bar at the bottom indicates 'Successfully run. Total query runtime: 232 msec. 62 rows affected.'

|    | nombre_asignatura  |
|----|--|
| 1  | Ingeniería de Requisitos   |
| 2  | Integración de las Tecnologías de la Información en las Organizaciones |
| 3  | Modelado y Diseño del Software 1                                       |
| 4  | Multiprocesadores  |
| 5  | Seguridad y cumplimiento normativo                                     |
| 6  | Sistema de Información para las Organizaciones                         |
| 7  | Tecnologías web  |
| 8  | Teoría de códigos y criptografía                                       |
| 9  | Administración de bases de datos                                       |
| 10 | Herramientas y Métodos de Ingeniería del Software                      |
| 11 | Informática industrial y robótica                                      |
| 12 | Ingeniería de Sistemas de Información                                  |
| 13 | Modelado y Diseño del Software 2                                       |
| 14 | Negocio Electrónico  |
| 15 | Periféricos e interfaces   |

- Devuelve un listado con todos los departamentos que tienen alguna asignatura que no se haya impartido en ningún curso escolar. El resultado debe mostrar el nombre del departamento y el nombre de la asignatura que no se haya impartido nunca.

```
SELECT D.nombre AS nombre_departamento, A.nombre AS nombre_asignatura
FROM departamento AS D
LEFT JOIN asignatura AS A ON D.id = A.id_grado
LEFT JOIN alumno_se_matricula_asignatura AS AMA ON A.id = AMA.id_asignatura
WHERE AMA.id_asignatura IS NULL;
```

The screenshot shows the pgAdmin 4 interface with the same setup as the previous one. The query results grid now contains 80 rows of data, showing combinations of departments and courses. A status bar at the bottom indicates 'Successfully run. Total query runtime: 65 msec. 80 rows affected.'

|    | nombre_departamento | nombre_asignatura  |
|----|---------------------|--|
| 1  | Educación           | Arquitectura de Computadores   |
| 2  | Educación           | Estructura de Datos y Algoritmos I                                     |
| 3  | Educación           | Ingeniería del Software  |
| 4  | Educación           | Sistemas Inteligentes  |
| 5  | Educación           | Sistemas Operativos  |
| 6  | Educación           | Bases de Datos   |
| 7  | Educación           | Estructura de Datos y Algoritmos II                                    |
| 8  | Educación           | Fundamentos de Redes de Computadores                                   |
| 9  | Educación           | Planificación y Gestión de Proyectos Informáticos                      |
| 10 | Educación           | Programación de Servicios Software                                     |
| 11 | Educación           | Desarrollo de interfaces de usuario                                    |
| 12 | Educación           | Ingeniería de Requisitos   |
| 13 | Educación           | Integración de las Tecnologías de la Información en las Organizaciones |
| 14 | Educación           | Modelado y Diseño del Software 1                                       |
| 15 | Educación           | Multiprocesadores  |

## Consultas resumen

1. Devuelve el número total de alumnas que hay.

Query    Query History

```
1 SELECT COUNT(*) AS alumnas FROM persona
2 WHERE tipo = 'alumno' AND sexo = 'M';
```

Data Output    Messages    Notifications

alumnas  
bigint

|   | alumnas |
|---|---------|
| 1 | 3       |

2. Calcula cuántos alumnos nacieron en 1999.

Query    Query History

```
1 SELECT COUNT(*) AS alumnos FROM persona
2 WHERE tipo = 'alumno' AND EXTRACT(YEAR FROM fecha_nacimiento) = 1999;
```

Data Output    Messages    Notifications

alumnos  
bigint

|   | alumnos |
|---|---------|
| 1 | 2       |

3. Calcula cuántos profesores hay en cada departamento. El resultado sólo debe mostrar dos columnas, una con el nombre del departamento y otra con el número de profesores que hay en ese departamento. El resultado sólo debe incluir los departamentos que tienen profesores asociados y deberá estar ordenado de mayor a menor por el número de profesores.

Query    Query History

```
1 SELECT dep.nombre AS nombre_departamento,
2 COUNT(profesor.id_profesor) AS numero_profesores
3 FROM departamento AS dep
4 INNER JOIN profesor ON dep.id = profesor.id_departamento
5 GROUP BY dep.nombre
6 ORDER BY numero_profesores DESC;
```

Data Output    Messages    Notifications

nombre\_departamento  
character varying (50)

numero\_profesores  
bigint

| nombre_departamento | numero_profesores |
|---------------------|-------------------|
| Educación           | 3                 |
| Química y Física    | 2                 |
| Matemáticas         | 2                 |
| Economía y Empresa  | 2                 |
| Informática         | 2                 |
| Agronomía           | 1                 |

4. Devuelve un listado con todos los departamentos y el número de profesores que hay en cada uno de ellos. Tenga en cuenta que pueden existir departamentos que no tienen profesores asociados. Estos departamentos también tienen que aparecer en el listado.

Query    Query History

```

1 SELECT departamento.nombre, COUNT(profesor.id_departamento) AS "Número de Profesores"
2 FROM departamento
3 LEFT JOIN profesor ON departamento.id = profesor.id_departamento
4 GROUP BY departamento.nombre
5 ORDER BY departamento.nombre;

```

Data Output    Messages    Notifications

| nombre              | Número de Profesores |
|---------------------|----------------------|
| Agronomía           | 1                    |
| Biología y Geología | 0                    |
| Derecho             | 0                    |
| Economía y Empresa  | 2                    |
| Educación           | 3                    |
| Filología           | 0                    |
| Informática         | 2                    |
| Matemáticas         | 2                    |
| Química y Física    | 2                    |

Total rows: 9 of 9    Query complete 00:00:00.070

5. Devuelve un listado con el nombre de todos los grados existentes en la base de datos y el número de asignaturas que tiene cada uno. Tenga en cuenta que pueden existir grados que no tienen asignaturas asociadas. Estos grados también tienen que aparecer en el listado. El resultado deberá estar ordenado de mayor a menor por el número de asignaturas.

```

1 SELECT grado.nombre, COUNT(asignatura.id_grado) AS "Número de Asignaturas"
2 FROM grado
3 LEFT JOIN asignatura ON grado.id = asignatura.id_grado
4 GROUP BY grado.nombre
5 ORDER BY "Número de Asignaturas" DESC;

```

Data Output    Messages    Notifications

| nombre  | Número de Asignaturas |
|---|-----------------------|
| Grado en Ingeniería Informática (Plan 2015)             | 51                    |
| Grado en Biotecnología (Plan 2015)                      | 32                    |
| Grado en Ingeniería Electrónica Industrial (Plan 201... | 0                     |
| Grado en Matemáticas (Plan 2010)                        | 0                     |
| Grado en Ingeniería Agrícola (Plan 2015)                | 0                     |
| Grado en Química (Plan 2009)                            | 0                     |
| Grado en Ingeniería Mecánica (Plan 2010)                | 0                     |
| Grado en Ciencias Ambientales (Plan 2009)               | 0                     |
| Grado en Ingeniería Química Industrial (Plan 2010)      | 0                     |
| Grado en Ingeniería Eléctrica (Plan 2014)               | 0                     |

Total rows: 10 of 10    Query complete 00:00:00.060

6. Devuelve un listado con el nombre de todos los grados existentes en la base de datos y el número de asignaturas que tiene cada uno, de los grados que tengan más de 40 asignaturas asociadas.

Query    Query History

```

1 SELECT grado.nombre, COUNT(asignatura.id_grado) AS "Número de Asignaturas"
2 FROM grado
3 INNER JOIN asignatura ON grado.id = asignatura.id_grado
4 GROUP BY grado.nombre
5 HAVING COUNT(asignatura.id_grado) > 40
6 ORDER BY "Número de Asignaturas" DESC;

```

Data Output    Messages    Notifications

| nombre                                      | Número de Asignaturas |
|---|-----------------------|
| Grado en Ingeniería Informática (Plan 2015) | 51                    |

7. Devuelve un listado que muestre el nombre de los grados y la suma del número total de créditos que hay para cada tipo de asignatura. El resultado debe tener tres columnas:

nombre del grado, tipo de asignatura y la suma de los créditos de todas las asignaturas que hay de ese tipo. Ordene el resultado de mayor a menor por el número total de créditos.

```

1 SELECT grado.nombre, asignatura.tipo, SUM(asignatura.creditos) AS "Total de Créditos"
2 FROM grado
3 INNER JOIN asignatura ON grado.id = asignatura.id_grado
4 GROUP BY grado.nombre, asignatura.tipo
5 ORDER BY "Total de Créditos" DESC;
6

```

Data Output Messages Notifications

|   | nombre<br>character varying (100)           | tipo<br>character varying (20) | Total de Créditos<br>double precision |
|---|---|--------------------------------|---------------------------------------|
| 1 | Grado en Ingeniería Informática (Plan 2015) | optativa                       | 180                                   |
| 2 | Grado en Biotecnología (Plan 2015)          | obligatoria                    | 120                                   |
| 3 | Grado en Ingeniería Informática (Plan 2015) | básica                         | 72                                    |
| 4 | Grado en Biotecnología (Plan 2015)          | básica                         | 60                                    |
| 5 | Grado en Ingeniería Informática (Plan 2015) | obligatoria                    | 54                                    |

8. Devuelve un listado que muestre cuántos alumnos se han matriculado de alguna asignatura en cada uno de los cursos escolares. El resultado deberá mostrar dos columnas, una columna con el año de inicio del curso escolar y otra con el número de alumnos matriculados.

```

1 SELECT anyo_inicio, COUNT(DISTINCT id_alumno) AS "Número de Alumnos Matriculados"
2 FROM curso_escolar
3 INNER JOIN alumno_se_matricula_asignatura
4 ON curso_escolar.id = alumno_se_matricula_asignatura.id_curso_escolar
5 GROUP BY anyo_inicio;
6

```

Data Output Messages Notifications

|   | anyo_inicio<br>integer | Número de Alumnos Matriculados<br>bigint |
|---|------------------------|--|
| 1 | 2014                   | 3  |
| 2 | 2018                   | 3  |

9. Devuelve un listado con el número de asignaturas que imparte cada profesor. El listado debe tener en cuenta aquellos profesores que no imparten ninguna asignatura. El resultado mostrará cinco columnas: id, nombre, primer apellido, segundo apellido y número de asignaturas. El resultado estará ordenado de mayor a menor por el número de asignaturas.

SELECT P.id\_profesor, Pe.nombre, Pe.apellido1, Pe.apellido2, COUNT(A.id) AS numero\_asignaturas

FROM profesor AS P

LEFT JOIN asignatura AS A ON P.id\_profesor = A.id\_profesor

JOIN persona AS Pe ON P.id\_profesor = Pe.id

GROUP BY P.id\_profesor, Pe.nombre, Pe.apellido1, Pe.apellido2

ORDER BY numero\_asignaturas DESC;

```

File Object Tools Help
Object Explorer Properties SQL Processes universidad/postgres universidad/postgres universidad/postgres universidad/postgres universidad/postgres universidad/postgres universidad/postgres
universidad/postgres@PostgreSQL 15
Query History Data Output Notifications Messages
Query
11 LEFT JOIN asignatura AS A ON D.id = A.id_departamento
12 WHERE A.id IS NULL;
13
14 SELECT D.nombre AS nombre_departamento, A.nombre AS nombre_asignatura
15 FROM departamento AS D
16 LEFT JOIN asignatura AS A ON D.id = A.id_grado
17 LEFT JOIN alumno_se_matricula_asignatura AS AMA ON A.id = AMA.id_asignatura
18 WHERE AMA.id_asignatura IS NULL;
19
20 SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS numero_ex
21 FROM profesor AS P
22 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
23 GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
24 ORDER BY numero_asignaturas DESC;
25
26 SELECT P.id_profesor, Pe.nombre, Pe.apellido1, Pe.apellido2, COUNT(A.id) AS numero_ex
27 FROM profesor AS P
28 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
29 JOIN persona AS Pe ON P.id_profesor = Pe.id
30 GROUP BY P.id_profesor, Pe.nombre, Pe.apellido1, Pe.apellido2
31 ORDER BY numero_asignaturas DESC;
32
33

```

Total rows: 12 of 12 Query complete 00:00:00.072 Ln 26, Col 1

|    | id_profesor | nombre    | apellido1  | apellido2  | numero_asignaturas |
|----|-------------|-----------|------------|------------|--------------------|
| 1  | 14          | Manolo    | Hamil      | Kozey      | 11                 |
| 2  | 3           | Zoe       | Ramirez    | Gea        | 10                 |
| 3  | 20          | Francesca | Schowalter | Muller     | 0                  |
| 4  | 5           | David     | Schmidt    | Fisher     | 0                  |
| 5  | 12          | Carmen    | Streich    | Hirth      | 0                  |
| 6  | 16          | Antonio   | Fahay      | Considine  | 0                  |
| 7  | 13          | Alfredo   | Stiedemann | Morissette | 0                  |
| 8  | 18          | Micaela   | Monahan    | Murray     | 0                  |
| 9  | 15          | Alejandro | Kohler     | Schoen     | 0                  |
| 10 | 8           | Cristina  | Lemke      | Rutherford | 0                  |
| 11 | 17          | Guillermo | Ruecker    | Upton      | 0                  |
| 12 | 10          | Esther    | Spencer    | Lakin      | 0                  |

Successfully run. Total query runtime: 72 msec. 12 rows affected. X Ln 26, Col 1

## Subconsultas

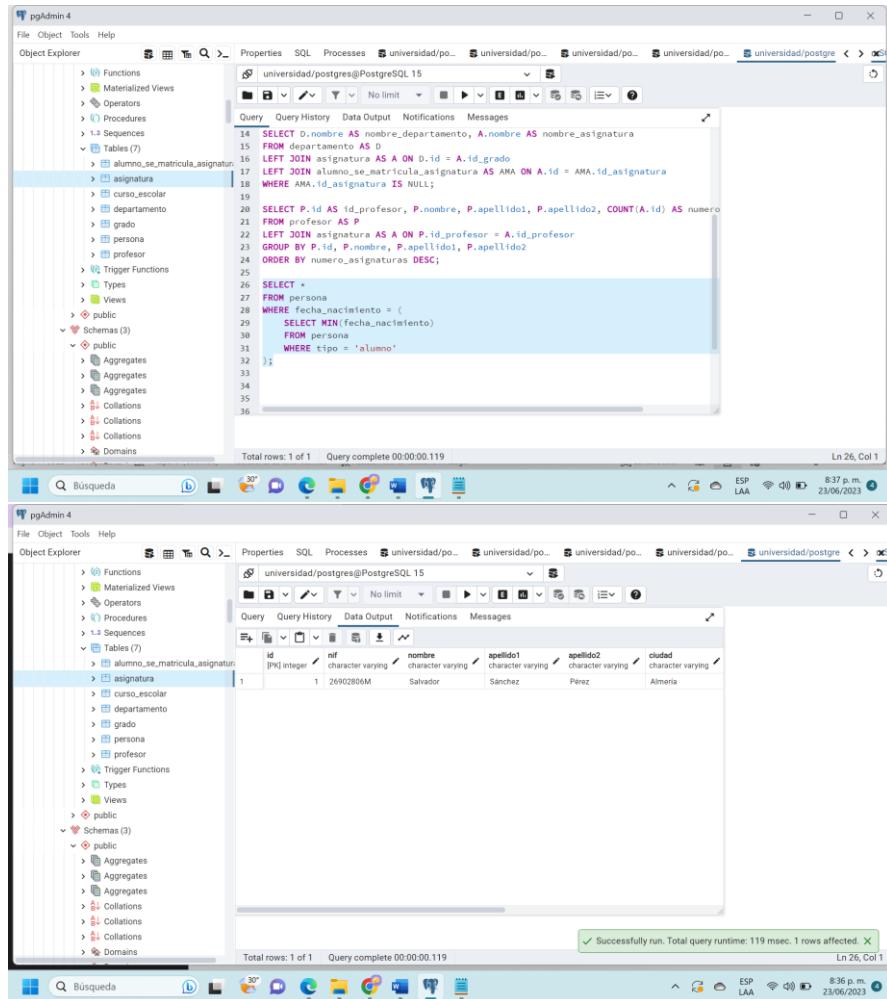
- Devuelve todos los datos del alumno más joven.

```

SELECT *
FROM persona
WHERE fecha_nacimiento =
    SELECT MIN(fecha_nacimiento)
FROM persona
WHERE tipo = 'alumno'

```

);



The screenshot shows two pgAdmin 4 windows side-by-side. The left window's Object Explorer shows various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables, Types, and Views. The right window's Query Editor contains the following SQL code:

```
14 SELECT D.nombre AS nombre_departamento, A.nombre AS nombre_asignatura
15 FROM departamento AS D
16 LEFT JOIN asignatura AS A ON D.id = A.id_grado
17 LEFT JOIN alumno_se_matricula_asignatura AS AMA ON A.id = AMA.id_asignatura
18 WHERE AMA.id_asignatura IS NULL;
19
20 SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS numero
21 FROM profesor AS P
22 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
23 GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
24 ORDER BY numero_asignaturas DESC;
25
26 SELECT *
27 FROM persona
28 WHERE fecha_nacimiento = (
29     SELECT MIN(fecha_nacimiento)
30     FROM persona
31     WHERE tipo = 'alumno'
32 );
33
34
35
36
```

The status bar at the bottom of the right window indicates "Total rows: 1 of 1 Query complete 00:00:00.119".

The bottom window's Object Explorer is identical to the top one. Its Query Editor contains a single query:

```
1 SELECT * FROM persona WHERE tipo = 'alumno'
```

The results grid shows one row of data from the 'asignatura' table:

| id | nif       | nombre   | apellido1 | apellido2 | ciudad  |
|----|-----------|----------|-----------|-----------|---------|
| 1  | 26902806M | Salvador | Sánchez   | Pérez     | Almería |

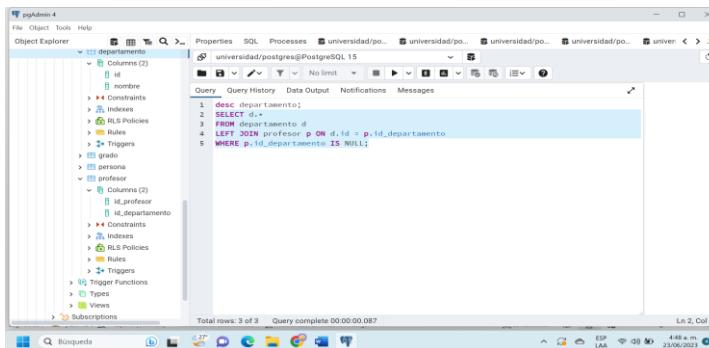
The status bar at the bottom of the bottom window indicates "Successfully run. Total query runtime: 119 msec. 1 rows affected. X" and "Total rows: 1 of 1 Query complete 00:00:00.119".

## 2. Devuelve un listado con los profesores que no están asociados a un departamento.

```
SELECT per.nombre, per.apellido1, per.apellido2
FROM persona AS per
LEFT JOIN profesor AS profe ON per.id = profe.id_profesor
LEFT JOIN departamento AS dep ON profe.id_departamento = dep.id
WHERE dep.id IS NULL;
```

### 3. Devuelve un listado con los departamentos que no tienen profesores asociados.

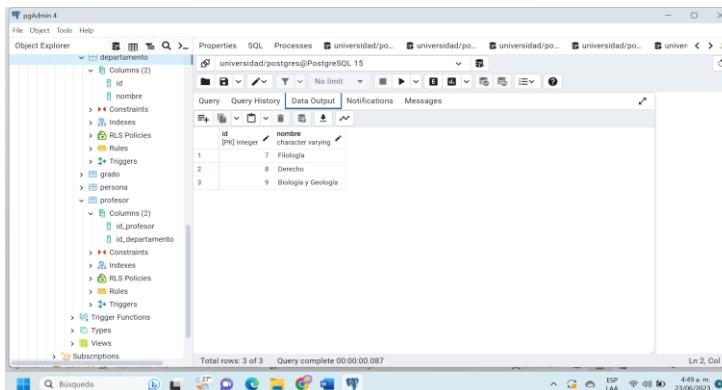
```
SELECT d.*  
  
FROM departamento d  
  
LEFT JOIN profesor p ON d.id = p.id_departamento  
  
WHERE p.id_departamento IS NULL;
```



The screenshot shows the pgAdmin 4 interface with the SQL tab selected. The query window contains the following code:

```
1 desc departamento;
2 SELECT d.* 
3 FROM departamento d
4 LEFT JOIN profesor p ON d.id = p.id_departamento
5 WHERE p.id_departamento IS NULL;
```

The status bar at the bottom indicates "Total rows: 3 of 3" and "Query complete 00:00:00.087".



The screenshot shows the pgAdmin 4 interface with the Data Output tab selected. The results table displays the following data:

| id | nombre                |
|----|-----------------------|
| 1  | Filosofía             |
| 2  | Derecho               |
| 3  | Biolología y Geología |

The status bar at the bottom indicates "Total rows: 3 of 3" and "Query complete 00:00:00.087".

### 4. Devuelve un listado con los profesores que tienen un departamento asociado y que no imparten ninguna asignatura.

```
SELECT per.nombre, per.apellido1, per.apellido2  
  
FROM persona AS per  
  
LEFT JOIN profesor AS profe ON per.id = profe.id_profesor  
  
LEFT JOIN asignatura AS asig ON profe.id_profesor = asig.id_profesor  
  
WHERE asig.id_profesor IS NULL;
```

pgAdmin 4

File Object Tools Help

Object Explorer

- Tables (7)
  - alumno\_se\_matricula\_asignatura
  - asignatura
  - curso\_escolar
  - departamento
  - grado
  - persona
  - profesor
- Constraints
- Indexes
- RLS Policies
- Rules
- Triggers
- departamento
- grado
- persona
- profesor
- Trigger Functions
- Types
- Views
- Schemas (3)
- public
- Aggregates
- Aggregates
- Aggregates

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#) No limit

Query History Data Output Notifications Messages

```

101 SELECT COUNT(*) AS alumnas FROM persona
102 WHERE tipo = 'alumno' AND sexo = 'M';
103
104 SELECT dep.nombre AS "Nombre del departamento", asig.nombre AS "Nombre de la asignatura"
105 FROM departamento AS dep
106 INNER JOIN profesor AS profe ON dep.id = profe.id_departamento
107 INNER JOIN asignatura AS asig ON profe.id_profesor = asig.id_profesor
108 LEFT JOIN alumno_se_matricula_asignatura AS alum_asig ON asig.id = alum_asig.id_asignatura
109 WHERE alum_asig.id_asignatura IS NULL;
110
111 SELECT per.nombre, per.apellido1, per.apellido2
112 FROM persona AS per
113 LEFT JOIN profesor AS profe ON per.id = profe.id_profesor
114 LEFT JOIN asignatura AS asig ON profe.id_profesor = asig.id_profesor
115 WHERE asig.id_profesor IS NULL;
116
117
118
119
120
121

```

Total rows: 22 of 22 Query complete 00:00:00.078 Ln 111, Col 1

Búsqueda 26° ESP LAA 3:47 a.m. 24/06/2023

pgAdmin 4

File Object Tools Help

Object Explorer

- Tables (7)
  - alumno\_se\_matricula\_asignatura
  - asignatura
  - curso\_escolar
  - departamento
  - grado
  - persona
  - profesor
- Constraints
- Indexes
- RLS Policies
- Rules
- Triggers
- departamento
- grado
- persona
- profesor
- Trigger Functions
- Types
- Views
- Schemas (3)
- public
- Aggregates
- Aggregates
- Aggregates

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#) No limit

Query History Data Output Notifications Messages

|    | nombre    | apellido1  | apellido2  |
|----|-----------|------------|------------|
| 1  | Salvador  | Sánchez    | Pérez      |
| 2  | Juan      | Saez       | Vega       |
| 3  | Pedro     | Heller     | Pagac      |
| 4  | David     | Schmidt    | Fisher     |
| 5  | José      | Koss       | Bayer      |
| 6  | Ismael    | Strosin    | Turcotte   |
| 7  | Cristina  | Lemke      | Rutherford |
| 8  | Ramiro    | Herzog     | Tremblay   |
| 9  | Esther    | Spencer    | Lakin      |
| 10 | Daniel    | Herman     | Pacocha    |
| 11 | Carmen    | Streich    | Hirthe     |
| 12 | Alfredo   | Steidemann | Monsette   |
| 13 | Alejandro | Kohler     | Schoen     |
| 14 | Antonio   | Fahney     | Conradine  |
| 15 | Guillermo | Ruecker    | Upton      |

✓ Successfully run. Total query runtime: 78 msec. 22 rows affected Ln 111, Col 1

Búsqueda 26° ESP LAA 3:46 a.m. 24/06/2023

5.Devuelve un listado con las asignaturas que no tienen un profesor asignado.

```

SELECT a.*
FROM asignatura a
LEFT JOIN profesor p ON a.id_profesor = p.id_profesor

```

WHERE a.id\_profesor IS NULL;

|    | id | nombre   | creditos | tipo     | curso | cuatrimestre | id_profesor | id_grado |
|----|----|--|----------|----------|-------|--------------|-------------|----------|
| 1  | 22 | Ingeniería de Requisitos   | 6        | optativa |       |              |             |          |
| 2  | 23 | Integración de las Tecnologías de la Información en las Organizaciones | 6        | optativa |       |              |             |          |
| 3  | 24 | Modelado y Diseño del Software 1                                       | 6        | optativa |       |              |             |          |
| 4  | 25 | Multiprocesadores  | 6        | optativa |       |              |             |          |
| 5  | 26 | Seguridad y cumplimiento normativo                                     | 6        | optativa |       |              |             |          |
| 6  | 27 | Sistema de Información para las Organizaciones                         | 6        | optativa |       |              |             |          |
| 7  | 28 | Tecnologías web  | 6        | optativa |       |              |             |          |
| 8  | 29 | Teoría de códigos y criptografía                                       | 6        | optativa |       |              |             |          |
| 9  | 30 | Administración de bases de datos                                       | 6        | optativa |       |              |             |          |
| 10 | 31 | Herramientas y Métodos de Ingeniería del Software                      | 6        | optativa |       |              |             |          |
| 11 | 32 | Informática industrial y robótica                                      | 6        | optativa |       |              |             |          |
| 12 | 33 | Ingeniería de Sistemas de Información                                  | 6        | optativa |       |              |             |          |
| 13 | 34 | Modelado y Diseño del Software 2                                       | 6        | optativa |       |              |             |          |
| 14 | 35 | Negocio Electrónico  | 6        | optativa |       |              |             |          |
| 15 | 36 | Periféricos e interfaces   | 6        | optativa |       |              |             |          |

|    | id | nombre   | creditos | tipo        | curso | cuatrimestre | id_profesor | id_grado |
|----|----|--|----------|-------------|-------|--------------|-------------|----------|
| 48 | 69 | ingeniería genética                            | 6        | obligatoria |       |              |             |          |
| 49 | 70 | Immunología                                    | 6        | obligatoria |       |              |             |          |
| 50 | 71 | Virología                                      | 6        | obligatoria |       |              |             |          |
| 51 | 72 | Bases moleculares del desarrollo vegetal       | 4.5      | obligatoria |       |              |             |          |
| 52 | 73 | Fisiología animal                              | 4.5      | obligatoria |       |              |             |          |
| 53 | 74 | Metabolismo y biosíntesis de biomoléculas      | 6        | obligatoria |       |              |             |          |
| 54 | 75 | Operaciones de separación                      | 6        | obligatoria |       |              |             |          |
| 55 | 76 | Patología molecular de plantas                 | 4.5      | obligatoria |       |              |             |          |
| 56 | 77 | Técnicas instrumentales básicas                | 4.5      | obligatoria |       |              |             |          |
| 57 | 78 | Bioinformática                                 | 4.5      | obligatoria |       |              |             |          |
| 58 | 79 | Biotecnología de los productos hortofruticolas | 4.5      | obligatoria |       |              |             |          |
| 59 | 80 | Biotecnología vegetal                          | 6        | obligatoria |       |              |             |          |
| 60 | 81 | Genómica y proteómica                          | 4.5      | obligatoria |       |              |             |          |
| 61 | 82 | Procesos biotecnológicos                       | 6        | obligatoria |       |              |             |          |
| 62 | 83 | Técnicas instrumentales avanzadas              | 4.5      | obligatoria |       |              |             |          |

The screenshot shows the pgAdmin 4 interface. The left pane, 'Object Explorer', lists several tables: 'alumno\_se\_matricula\_asignatura', 'asignatura', 'departamento', 'curso\_escolar', and 'profesor'. The 'asignatura' table is expanded, showing columns: id, nombre, creditos, tipo, curso, cuatrimestre, id\_profesor, and id\_grado. The right pane is a 'Query' window titled 'universidad/postgres@PostgreSQL 15' with the following data:

| id | nombre   | creditos | tipo        | curso | cuatrimestre |
|----|--|----------|-------------|-------|--------------|
| 48 | ingeniería genética                            | 6        | obligatoria |       | 2            |
| 49 | Inmunología                                    | 6        | obligatoria |       | 2            |
| 50 | Virología                                      | 6        | obligatoria |       | 2            |
| 51 | Bases moleculares del desarrollo vegetal       | 4.5      | obligatoria |       | 3            |
| 52 | Fisiología animal                              | 4.5      | obligatoria |       | 3            |
| 53 | Metabolismo y biosíntesis de biomoléculas      | 6        | obligatoria |       | 3            |
| 54 | Operaciones de separación                      | 6        | obligatoria |       | 3            |
| 55 | Patología molecular de plantas                 | 4.5      | obligatoria |       | 3            |
| 56 | Técnicas instrumentales básicas                | 4.5      | obligatoria |       | 3            |
| 57 | Bioinformática                                 | 4.5      | obligatoria |       | 3            |
| 58 | Biotecnología de los productos hortofrutícolas | 4.5      | obligatoria |       | 3            |
| 59 | Biotecnología vegetal                          | 6        | obligatoria |       | 3            |
| 60 | Genómica y proteómica                          | 4.5      | obligatoria |       | 3            |
| 61 | Procesos biotecnológicos                       | 6        | obligatoria |       | 3            |
| 62 | Técnicas instrumentales avanzadas              | 4.5      | obligatoria |       | 3            |

Total rows: 62 of 62 Query complete 00:00:00.123 Ln 13, Col 1

6.Devuelve un listado con todos los departamentos que tienen alguna asignatura que no se haya impartido en ningún curso escolar. El resultado debe mostrar el nombre del departamento y el nombre de la asignatura que no se haya impartido nunca.

```

SELECT dep.nombre AS "Nombre del departamento", asig.nombre AS "Nombre de la
asignatura"
FROM departamento AS dep
INNER JOIN profesor AS profe ON dep.id = profe.id_departamento
INNER JOIN asignatura AS asig ON profe.id_profesor = asig.id_profesor
LEFT JOIN alumno_se_matricula_asignatura AS alum_asig ON asig.id =
alum_asig.id_asignatura
WHERE alum_asig.id_asignatura IS NULL;

```

The screenshot shows the pgAdmin 4 interface. The left pane, 'Object Explorer', lists tables: 'alumno\_se\_matricula\_asignatura', 'asignatura', 'curso\_escolar', and 'departamento'. The 'asignatura' table is expanded, showing columns: id, nombre, anyo\_inicio, and anyo\_fin. The right pane is a 'Query' window with the following complex query:

```

96 SELECT A.id, A.nombre
97 FROM asignatura AS A
98 JOIN grado G ON A.id_grado = G.id
99 WHERE G.nombre = 'Ingeniería Informática' AND A.tipo = 'ofertada';
100
101 SELECT COUNT(*) AS alumnos FROM persona
102 WHERE tipo = 'alumno' AND sexo = 'M'
103
104 SELECT dep.nombre AS "Nombre del departamento", asig.nombre AS "Nombre de la asignatura"
105 FROM departamento AS dep
106 INNER JOIN profesor AS profe ON dep.id = profe.id_departamento
107 INNER JOIN asignatura AS asig ON profe.id_profesor = asig.id_profesor
108 LEFT JOIN alumno_se_matricula_asignatura AS alum_asig ON asig.id = alum_asig.id_asignatura
109 WHERE alum_asig.id_asignatura IS NULL;
110
111
112
113
114
115
116

```

Total rows: 11 of 11 Query complete 00:00:00.057 Ln 104, Col 1

Object Explorer

Properties SQL Processes

| Nombre del departamento | Nombre de la asignatura                           |
|-------------------------|---|
| Informática             | Arquitectura de Computadores                      |
| Informática             | Estructura de Datos y Algoritmos I                |
| Informática             | Ingierencia del Software                          |
| Informática             | Sistemas Inteligentes                             |
| Informática             | Sistemas Operativos                               |
| Informática             | Bases de Datos                                    |
| Informática             | Estructura de Datos y Algoritmos II               |
| Informática             | Fundamentos de Redes de Computadores              |
| Informática             | Planificación y Gestión de Proyectos Informáticos |
| Informática             | Programación de Servicios Software                |
| Informática             | Desarrollo de Interfaces de usuarios              |

## Consultas resumen

1. Devuelve el número total de alumnas que hay.

```

SELECT COUNT(*) AS alumnas
FROM asignatura
INNER JOIN grado ON asignatura.id_grado = grado.id
WHERE grado.nombre = 'Grado en Ingeniería(plan 2015)';

SELECT A.id, A.nombre
FROM asignatura AS A
JOIN grado AS G ON A.id_grado = G.id
WHERE G.nombre = 'Ingeniería Informática' AND A.tipo = 'ofertada';

SELECT COUNT(*) AS alumnas FROM persona
WHERE tipo = 'alumno' AND sexo = 'M';
    
```

Object Explorer

Properties SQL Processes

| alumnas |
|---------|
| 3       |

2.Calcula cuántos alumnos nacieron en 1999.

`SELECT COUNT(*) AS total_alumnos`

```

FROM persona
WHERE fecha_nacimiento >= '1999-01-01' AND fecha_nacimiento <= '1999-12-31' AND tipo
= 'alumno';

```

```

16 LEFT JOIN asignatura AS A ON D.id = A.id_grado
17 LEFT JOIN alumno_se_matricula_asignatura AS AMA ON A.id = AMA.id_asignatura
18 WHERE AMA.id_asignatura IS NULL;
19
20 SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS numero
21 FROM profesor AS P
22 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
23 GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
24 ORDER BY numero_asignaturas DESC;
25
26 SELECT *
27 FROM persona
28 WHERE fecha_nacimiento = (
29     SELECT MIN(fecha_nacimiento)
30     FROM persona
31     WHERE tipo = 'alumno'
32 );
33
34 SELECT COUNT(*) AS total_alumnos
35 FROM persona
36 WHERE fecha_nacimiento >= '1999-01-01' AND fecha_nacimiento <= '1999-12-31' AND tipo
37
38

```

Total rows: 1 of 1    Query complete 00:00:00.210    Ln 34, Col 1

| total_alumnos | bignum |
|---------------|--------|
| 1             | 2      |

Total rows: 1 of 1    Query complete 00:00:00.210    Ln 34, Col 1

3.Calcula cuántos profesores hay en cada departamento. El resultado sólo debe mostrar dos columnas, una con el nombre del departamento y otra con el número de profesores que hay en ese departamento. El resultado sólo debe incluir los departamentos que tienen profesores asociados y deberá estar ordenado de mayor a menor por el número de profesores.

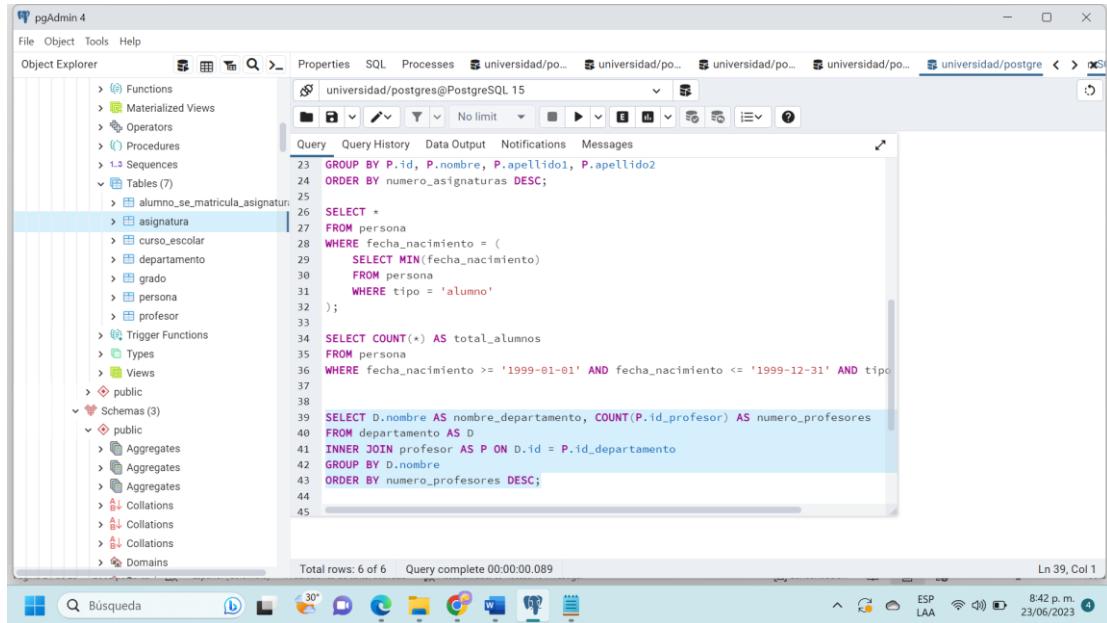
```

SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
FROM departamento AS D
INNER JOIN profesor AS P ON D.id = P.id_departamento

```

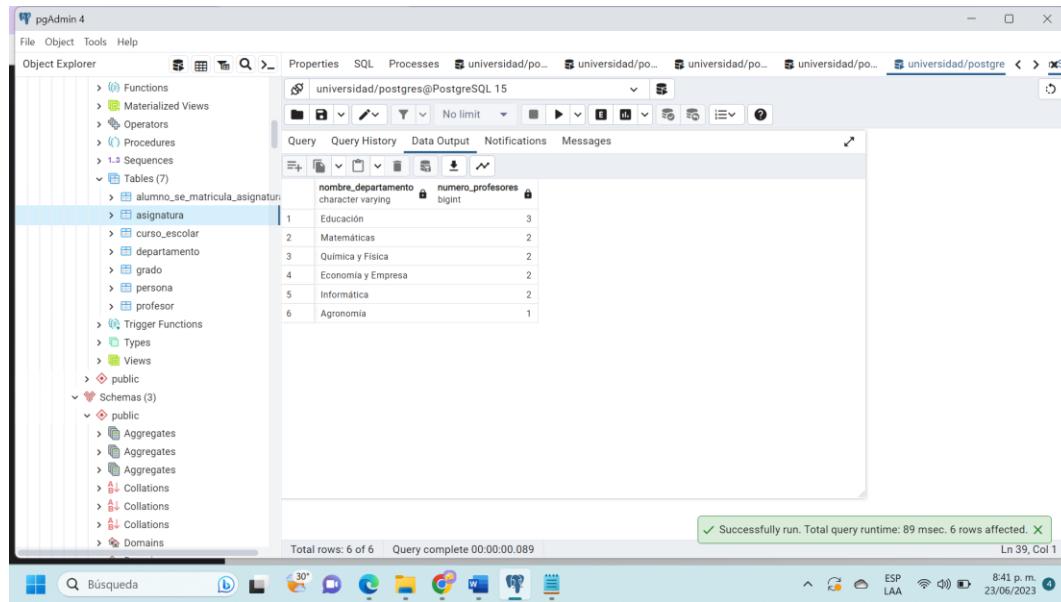
GROUP BY D.nombre

ORDER BY numero\_profesores DESC;



```
23 GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
24 ORDER BY numero_asignaturas DESC;
25
26 SELECT *
27 FROM persona
28 WHERE fecha_nacimiento = (
29     SELECT MIN(fecha_nacimiento)
30     FROM persona
31     WHERE tipo = 'alumno'
32 );
33
34 SELECT COUNT(*) AS total_alumnos
35 FROM persona
36 WHERE fecha_nacimiento >= '1999-01-01' AND fecha_nacimiento <= '1999-12-31' AND tipo
37
38
39 SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
40 FROM departamento AS D
41 INNER JOIN profesor AS P ON D.id = P.id_departamento
42 GROUP BY D.nombre
43 ORDER BY numero_profesores DESC;
44
45
```

Total rows: 6 of 6    Query complete 00:00:00.089    Ln 39, Col 1



| nombre_departamento | numero_profesores |
|---------------------|-------------------|
| Educación           | 3                 |
| Matemáticas         | 2                 |
| Química y Física    | 2                 |
| Economía y Empresa  | 2                 |
| Informática         | 2                 |
| Agronomía           | 1                 |

Successfully run. Total query runtime: 89 msec. 6 rows affected.    Ln 39, Col 1

- 4.Devuelve un listado con todos los departamentos y el número de profesores que hay en cada uno de ellos. Tenga en cuenta que pueden existir departamentos que no tienen profesores asociados. Estos departamentos también tienen que aparecer en el listado.

```

SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
FROM departamento AS D
LEFT JOIN profesor AS P ON D.id = P.id_departamento
GROUP BY D.nombre
ORDER BY numero_profesores DESC;

```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, it shows the database schema with various objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables (7), Triggers, Types, and Views.
- Properties:** A tab at the top of the main window.
- SQL:** A tab at the top of the main window.
- Processes:** A tab at the top of the main window.
- Query:** The main area where the SQL query is entered. The query is as follows:

```

29  SELECT MIN(fecha_nacimiento)
30    FROM persona
31   WHERE tipo = 'alumno'
32 );
33
34  SELECT COUNT(*) AS total_alumnos
35    FROM persona
36   WHERE fecha_nacimiento >= '1999-01-01' AND fecha_nacimiento <= '1999-12-31' AND tipo
37
38
39  SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
40    FROM departamento AS D
41   INNER JOIN profesor AS P ON D.id = P.id_departamento
42
43 GROUP BY D.nombre
44
45  SELECT * FROM consulta WHERE id = 1
46
47
48
49
50
51

```

- Data Output:** A tab at the top of the main window.
- Notifications:** A tab at the top of the main window.
- Messages:** A tab at the top of the main window.
- Status Bar:** Shows "Ln 44, Col 1" and the system tray with icons for battery, signal, and date/time.

The screenshot shows the pgAdmin 4 interface with the following details:

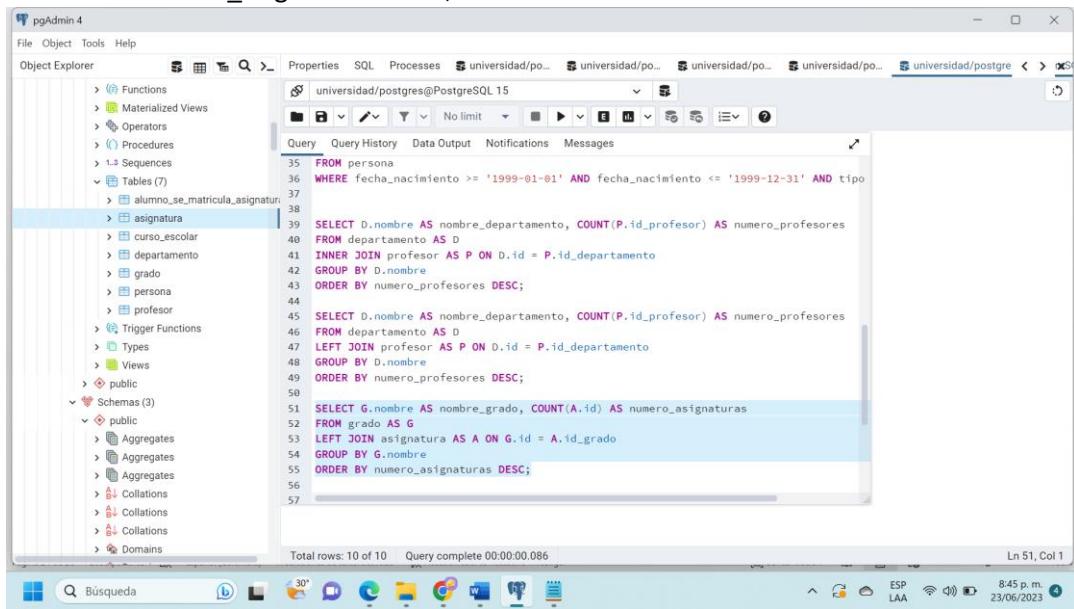
- Object Explorer:** On the left, it shows the database schema with various objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables (7), Triggers, Types, and Views.
- Properties:** A tab at the top of the main window.
- SQL:** A tab at the top of the main window.
- Processes:** A tab at the top of the main window.
- Tables:** A tab at the top of the main window.
- Data Output:** The active tab at the top of the main window, showing the results of the executed query. The results are:

| nombre_departamento   | numero_profesores |
|-----------------------|-------------------|
| Educación             | 3                 |
| Economía y Empresa    | 2                 |
| Química y Física      | 2                 |
| Matemáticas           | 2                 |
| Informática           | 2                 |
| Agronomía             | 1                 |
| Biotología y Geología | 0                 |
| Derecho               | 0                 |
| Filología             | 0                 |

- Notifications:** A tab at the top of the main window.
- Messages:** A tab at the top of the main window.
- Status Bar:** Shows "Successfully run. Total query runtime: 114 msec. 9 rows affected." and the system tray with icons for battery, signal, and date/time.

5.Devuelve un listado con el nombre de todos los grados existentes en la base de datos y el número de asignaturas que tiene cada uno. Tenga en cuenta que pueden existir grados que no tienen asignaturas asociadas. Estos grados también tienen que aparecer en el listado. El resultado deberá estar ordenado de mayor a menor por el número de asignaturas.

```
SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
FROM grado AS G
LEFT JOIN asignatura AS A ON G.id = A.id_grado
GROUP BY G.nombre
ORDER BY numero_asignaturas DESC;
```



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane displays various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, Tables (7), Triggers, Types, and Views. The Tables node is expanded, showing tables such as alumno\_se\_matricula\_asignatura, asignatura, curso\_escolar, departamento, grado, persona, profesor, and trigger functions. Below the Object Explorer is the main window containing the SQL query editor. The query is:

```
35 FROM persona
36 WHERE fecha_nacimiento >= '1999-01-01' AND fecha_nacimiento <= '1999-12-31' AND tipo
37
38
39 SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
40 FROM departamento AS D
41 INNER JOIN profesor AS P ON D.id = P.id_departamento
42 GROUP BY D.nombre
43 ORDER BY numero_profesores DESC;
44
45 SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
46 FROM departamento AS D
47 LEFT JOIN profesor AS P ON D.id = P.id_departamento
48 GROUP BY D.nombre
49 ORDER BY numero_profesores DESC;
50
51 SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
52 FROM grado AS G
53 LEFT JOIN asignatura AS A ON G.id = A.id_grado
54 GROUP BY G.nombre
55 ORDER BY numero_asignaturas DESC;
```

The status bar at the bottom indicates "Total rows: 10 of 10" and "Query complete 00:00:00.086". The bottom right corner shows the date and time: "8:45 p. m. 23/06/2023".

The screenshot shows the pgAdmin 4 interface with a query results window. The table contains the following data:

| nombre_grado  | numero_asignaturas |
|---|--------------------|
| Grado en Ingeniería Informática (Plan 2015)             | 51                 |
| Grado en Biotecnología (Plan 2015)                      | 32                 |
| Grado en Ingeniería Mecánica (Plan 2010)                | 0                  |
| Grado en Ingeniería Química Industrial (Plan 2010)      | 0                  |
| Grado en Matemáticas (Plan 2010)                        | 0                  |
| Grado en Química (Plan 2009)                            | 0                  |
| Grado en Ingeniería Agrícola (Plan 2015)                | 0                  |
| Grado en Ingeniería Electrónica Industrial (Plan 201... | 0                  |
| Grado en Ingeniería Eléctrica (Plan 2014)               | 0                  |
| Grado en Ciencias Ambientales (Plan 2009)               | 0                  |

6. Devuelve un listado con el nombre de todos los grados existentes en la base de datos y el número de asignaturas que tiene cada uno, de los grados que tengan más de 40 asignaturas asociadas.

```
SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
FROM grado AS G
LEFT JOIN asignatura AS A ON G.id = A.id_grado
GROUP BY G.nombre
HAVING COUNT(A.id) > 40
ORDER BY numero_asignaturas DESC;
```

```

File Object Tools Help
Object Explorer Properties SQL Processes
universidad/postgres@PostgreSQL 15
Query History Data Output Notifications Messages
No limit
Query
SELECT D.nombre AS nombre_departamento, COUNT(P.id_profesor) AS numero_profesores
FROM departamento AS D
LEFT JOIN profesor AS P ON D.id = P.id_departamento
GROUP BY D.nombre
ORDER BY numero_profesores DESC;

SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
FROM grado AS G
LEFT JOIN asignatura AS A ON G.id = A.id_grado
GROUP BY G.nombre
ORDER BY numero_asignaturas DESC;

SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
FROM grado AS G
LEFT JOIN asignatura AS A ON G.id = A.id_grado
GROUP BY G.nombre
HAVING COUNT(A.id) > 40
ORDER BY numero_asignaturas DESC;

```

Total rows: 1 of 1    Query complete 00:00:00.081    Ln 57, Col 1

| nombre_grado                                | numero_asignaturas |
|---|--------------------|
| Grado en Ingeniería Informática (Plan 2015) | 51                 |

Successfully run. Total query runtime: 81 msec. 1 rows affected.    Ln 57, Col 1

7. Devuelve un listado que muestre el nombre de los grados y la suma del número total de créditos que hay para cada tipo de asignatura. El resultado debe tener tres columnas: nombre del grado, tipo de asignatura y la suma de los créditos de todas las asignaturas que hay de ese tipo. Ordene el resultado de mayor a menor por el número total de créditos.

```

SELECT G.nombre AS nombre_grado, A.tipo, SUM(A.creditos) AS total_creditos
FROM grado AS G
LEFT JOIN asignatura AS A ON G.id = A.id_grado
GROUP BY G.nombre, A.tipo
ORDER BY total_creditos DESC;

```

pgAdmin 4

File Object Tools Help

Object Explorer

- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (7)
  - > alumno\_se\_matricula\_asignatura
  - > asignatura
  - > curso\_escolar
  - > departamento
  - > grado
  - > persona
  - > profesor
- > Trigger Functions
- > Types
- > Views
- > public
- > Schemas (3)
  - > public
  - > Aggregates
  - > Aggregates
  - > Aggregates
  - > Collations
  - > Collations
  - > Collations
  - > Domains

Properties SQL Processes [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#)

Query Query History Data Output Notifications Messages

```

56
57 SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
58 FROM grado AS G
59 LEFT JOIN asignatura AS A ON G.id = A.id_grado
60 GROUP BY G.nombre
61 HAVING COUNT(A.id) > 40
62 ORDER BY numero_asignaturas DESC;
63
64 SELECT G.nombre AS nombre_grado, A.tipo, SUM(A.creditos) AS total_creditos
65 FROM grado AS G
66 LEFT JOIN asignatura AS A ON G.id = A.id_grado
67 GROUP BY G.nombre, A.tipo
68 ORDER BY total_creditos DESC;
69
70
71
72
73
74
75
76
77
78

```

Total rows: 13 of 13 Query complete 00:00:00.074 Ln 64, Col 1

Búsqueda 8:47 p. m. 23/06/2023

pgAdmin 4

File Object Tools Help

Object Explorer

- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (7)
  - > alumno\_se\_matricula\_asignatura
  - > asignatura
  - > curso\_escolar
  - > departamento
  - > grado
  - > persona
  - > profesor
- > Trigger Functions
- > Types
- > Views
- > public
- > Schemas (3)
  - > public
  - > Aggregates
  - > Aggregates
  - > Aggregates
  - > Collations
  - > Collations
  - > Collations
  - > Domains

Properties SQL Processes [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#) [universidad/postre](#)

Query Query History Data Output Notifications Messages

| nombre_grado  | tipo        | total_creditos |
|---|-------------|----------------|
| 1 Grado en Ingeniería Agrícola (Plan 2015)                | [null]      | [null]         |
| 2 Grado en Ingeniería Mecánica (Plan 2010)                | [null]      | [null]         |
| 3 Grado en Matemáticas (Plan 2010)                        | [null]      | [null]         |
| 4 Grado en Ciencias Ambientales (Plan 2009)               | [null]      | [null]         |
| 5 Grado en Química (Plan 2009)                            | [null]      | [null]         |
| 6 Grado en Ingeniería Química Industrial (Plan 2010)      | [null]      | [null]         |
| 7 Grado en Ingeniería Electrónica Industrial (Plan 201... | [null]      | [null]         |
| 8 Grado en Ingeniería Eléctrica (Plan 2014)               | [null]      | [null]         |
| 9 Grado en Ingeniería Informática (Plan 2015)             | optativa    | 180            |
| 10 Grado en Biotecnología (Plan 2015)                     | obligatoria | 120            |
| 11 Grado en Ingeniería Informática (Plan 2015)            | básica      | 72             |
| 12 Grado en Biotecnología (Plan 2015)                     | básica      | 60             |
| 13 Grado en Ingeniería Informática (Plan 2015)            | obligatoria | 54             |

Total rows: 13 of 13 Query complete 00:00:00.074 Successfully run. Total query runtime: 74 msec. 13 rows affected. Ln 64, Col 1

Búsqueda 8:47 p. m. 23/06/2023

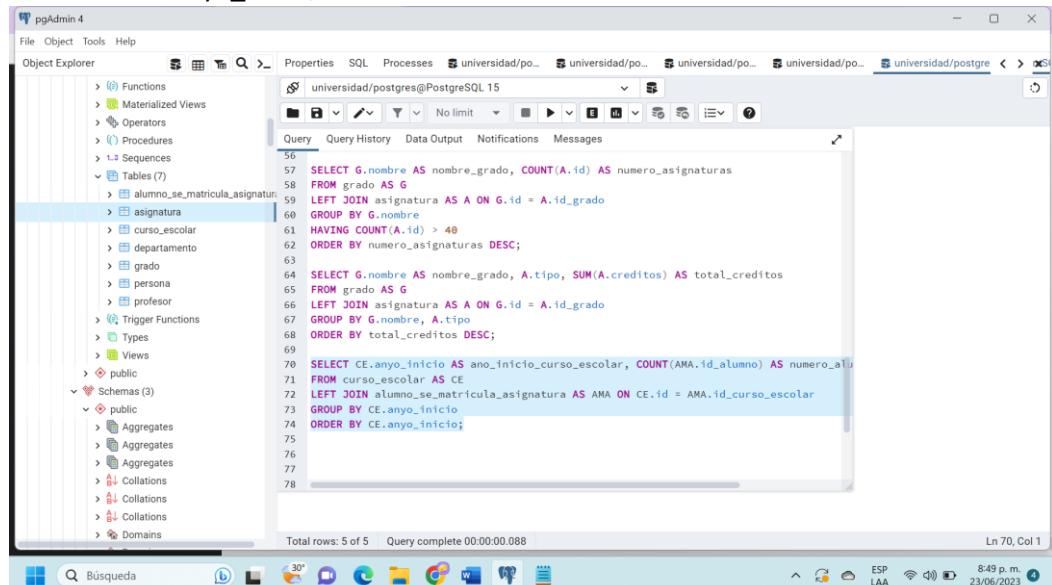
8. Devuelve un listado que muestre cuántos alumnos se han matriculado de alguna asignatura en cada uno de los cursos escolares. El resultado deberá mostrar dos columnas, una columna con el año de inicio del curso escolar y otra con el número de alumnos matriculados.

```

SELECT CE.anho_inicio AS ano_inicio_curso_escolar, COUNT(AMA.id_alumno) AS
numero_alumnos_matriculados
FROM curso_escolar AS CE
LEFT JOIN alumno_se_matricula_asignatura AS AMA ON CE.id = AMA.id_curso_escolar

```

ORDER BY CE.anexo\_inicio;

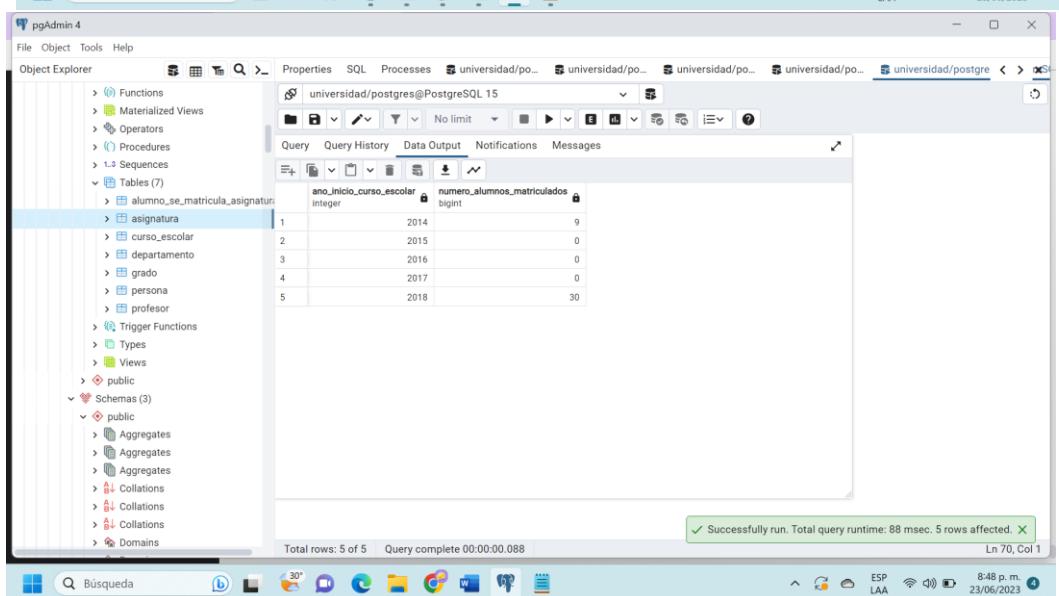


```

56
57 SELECT G.nombre AS nombre_grado, COUNT(A.id) AS numero_asignaturas
58 FROM grado AS G
59 LEFT JOIN asignatura AS A ON G.id = A.id_grado
60 GROUP BY G.nombre
61 HAVING COUNT(A.id) > 40
62 ORDER BY numero_asignaturas DESC;
63
64 SELECT G.nombre AS nombre_grado, A.tipo, SUM(A.creditos) AS total_creditos
65 FROM grado AS G
66 LEFT JOIN asignatura AS A ON G.id = A.id_grado
67 GROUP BY G.nombre, A.tipo
68 ORDER BY total_creditos DESC;
69
70 SELECT CE.anexo_inicio AS anexo_inicio_curso_escolar, COUNT(AMA.id_alumno) AS numero_alumnos
71 FROM curso_escolar AS CE
72 LEFT JOIN alumno_se_matricula_asignatura AS AMA ON CE.id = AMA.id_curso_escolar
73 GROUP BY CE.anexo_inicio
74 ORDER BY CE.anexo_inicio;
75
76
77

```

Total rows: 5 of 5    Query complete 00:00:00.088    Ln 70, Col 1



| anexo_inicio_curso_escolar | numero_alumnos_matriculados |
|----------------------------|-----------------------------|
| 1                          | 2014                        |
| 2                          | 2015                        |
| 3                          | 2016                        |
| 4                          | 2017                        |
| 5                          | 2018                        |

✓ Successfully run. Total query runtime: 88 msec. 5 rows affected.    Ln 70, Col 1

- Devuelve un listado con el número de asignaturas que imparte cada profesor. El listado debe tener en cuenta aquellos profesores que no imparten ninguna asignatura. El resultado mostrará cinco columnas: id, nombre, primer apellido, segundo apellido y número de asignaturas. El resultado estará ordenado de mayor a menor por el número de asignaturas.

```

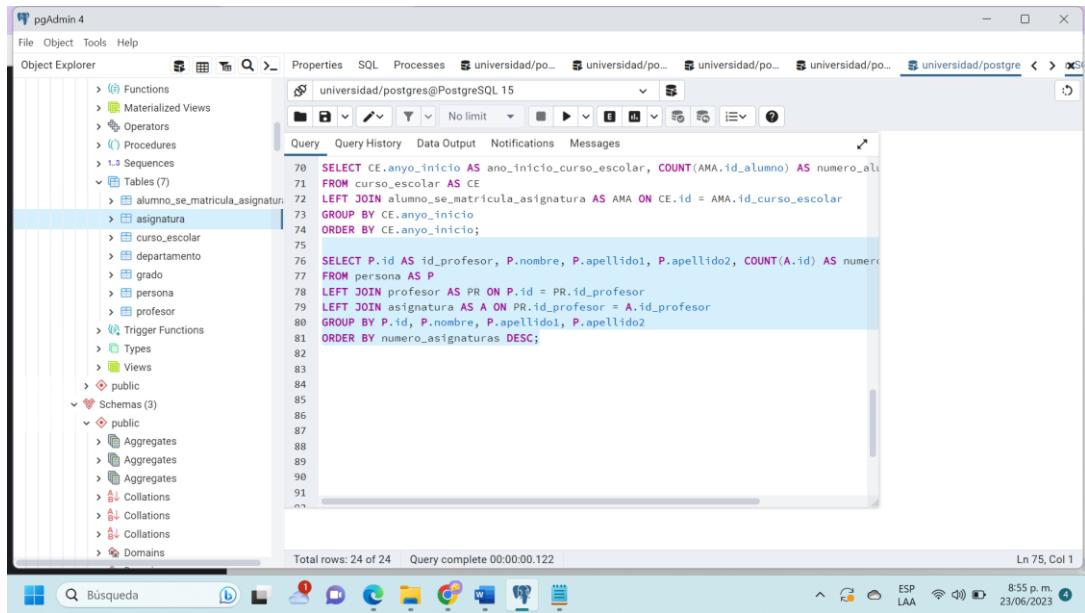
SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS
numero_asignaturas
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor

```

```

LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
ORDER BY numero_asignaturas DESC;

```



The screenshot shows the pgAdmin 4 interface with the SQL tab active. The query window contains the following SQL code:

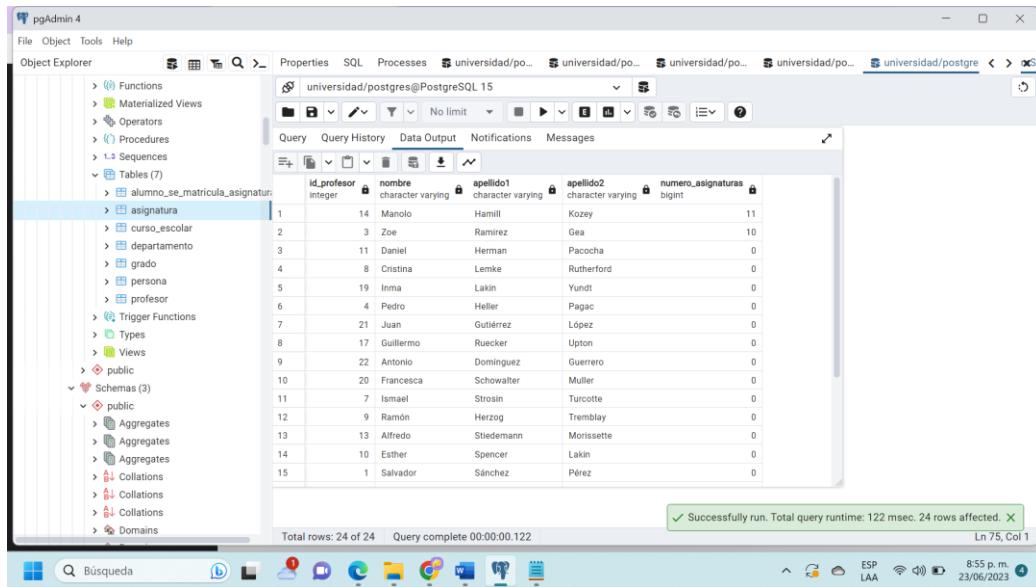
```

SELECT CE.anexo_inicio AS anexo_inicio_curso_escolar, COUNT(AMA.id_alumno) AS numero_alumnos
FROM curso_escolar AS CE
LEFT JOIN alumno_se_matricula_asignatura AS AMA ON CE.id = AMA.id_curso_escolar
GROUP BY CE.anexo_inicio
ORDER BY CE.anexo_inicio;

SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS numero_asignaturas
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
ORDER BY numero_asignaturas DESC;

```

The status bar at the bottom indicates "Total rows: 24 of 24" and "Query complete 00:00:00.122".



The screenshot shows the pgAdmin 4 interface with the Data Output tab active. The results of the query are displayed in a table:

|    | id_profesor | nombre    | apellido1  | apellido2  | numero_asignaturas |
|----|-------------|-----------|------------|------------|--------------------|
| 1  | 14          | Manolo    | Hamil      | Kozey      | 11                 |
| 2  | 3           | Zoe       | Ramirez    | Gea        | 10                 |
| 3  | 11          | Daniel    | Herman     | Pacocha    | 0                  |
| 4  | 8           | Cristina  | Lemke      | Rutherford | 0                  |
| 5  | 19          | Irma      | Lakin      | Yundt      | 0                  |
| 6  | 4           | Pedro     | Heller     | Pagac      | 0                  |
| 7  | 21          | Juan      | Gutiérrez  | López      | 0                  |
| 8  | 17          | Guillermo | Ruecker    | Upton      | 0                  |
| 9  | 22          | Antonio   | Dominguez  | Guerrero   | 0                  |
| 10 | 20          | Francesca | Schowalter | Muller     | 0                  |
| 11 | 7           | Ismael    | Stros      | Turcotte   | 0                  |
| 12 | 9           | Ramón     | Herzog     | Tremblay   | 0                  |
| 13 | 13          | Alfredo   | Stiedemann | Morisette  | 0                  |
| 14 | 10          | Ester     | Spencer    | Lakin      | 0                  |
| 15 | 1           | Salvador  | Sánchez    | Pérez      | 0                  |

The status bar at the bottom indicates "Successfully run. Total query runtime: 122 msec. 24 rows affected." and "Total rows: 24 of 24".

## Subconsultas

- Devuelve todos los datos del alumno más joven.

```
SELECT *
```

```

FROM persona
WHERE tipo = 'alumno' AND fecha_nacimiento = (
    SELECT MIN(fecha_nacimiento)
    FROM persona
    WHERE tipo = 'alumno'
);

```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer with the following tree structure:

- grado
- persona
  - Columns (11)
  - Constraints
  - Indexes
  - RLS Policies
  - Rules
  - Triggers
- profesor
  - Columns (2)
  - Constraints

The persona table's columns are listed as follows:

|   | <b>id</b><br>[PK] integer | <b>nif</b><br>character varying | <b>nombre</b><br>character varying | <b>apellido1</b><br>character varying | <b>apellido2</b><br>character varying | <b>ciudad</b><br>character varying | <b>direccion</b><br>character varying | <b>telefono</b><br>char |
|---|---------------------------|---------------------------------|------------------------------------|---------------------------------------|---------------------------------------|------------------------------------|---------------------------------------|-------------------------|
| 1 | 1                         | 26902806M                       | Salvador                           | Sánchez                               | Pérez                                 | Almería                            | C/ Real del barrio alto               | 950                     |

At the bottom of the pgAdmin window, it says "Total rows: 1 of 1" and "Query complete 00:00:00.316".

- Devuelve un listado con los profesores que no están asociados a un departamento.

```

SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
WHERE PR.id_departamento IS NULL;

```

```

SELECT CE.anho_inicio AS año_inicio_curso_escolar, COUNT(AMA.id_aLumno) AS numero_alumnos
FROM curso_escolar AS CE
LEFT JOIN alumno_se_matricula_asignatura AS AMA ON CE.id = AMA.id_curso_escolar
GROUP BY CE.anho_inicio
ORDER BY CE.anho_inicio;

SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2, COUNT(A.id) AS numero_asignaturas
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
GROUP BY P.id, P.nombre, P.apellido1, P.apellido2
ORDER BY numero_asignaturas DESC;

SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2
FROM persona AS P
LEFT JOIN profesor AS PR ON P.id = PR.id_profesor
WHERE PR.id_departamento IS NULL;

```

Total rows: 12 of 12    Query complete 00:00:00.096    Ln 83, Col 1

|    | <b>id_profesor</b> | <b>nombre</b> | <b>apellido1</b> | <b>apellido2</b> |
|----|--------------------|---------------|------------------|------------------|
| 1  | 1                  | Salvador      | Sánchez          | Pérez            |
| 2  | 2                  | Juan          | Sáez             | Vega             |
| 3  | 4                  | Pedro         | Heller           | Pagac            |
| 4  | 6                  | José          | Koss             | Bayer            |
| 5  | 7                  | Ismael        | Strosin          | Turcotte         |
| 6  | 9                  | Ramón         | Herzog           | Tremblay         |
| 7  | 11                 | Daniel        | Herman           | Pacocha          |
| 8  | 19                 | Inma          | Lakin            | Yundt            |
| 9  | 21                 | Juan          | Gutiérrez        | López            |
| 10 | 22                 | Antonio       | Domínguez        | Guerrero         |
| 11 | 23                 | Irene         | Hernández        | Martínez         |
| 12 | 24                 | Sonia         | Gea              | Ruiz             |

Total rows: 12 of 12    Query complete 00:00:00.096    Successfully run. Total query runtime: 96 msec. 12 rows affected.    Ln 83, Col 1

3. Devuelve un listado con los departamentos que no tienen profesores asociados.

```

SELECT D.id AS id_departamento, D.nombre
FROM departamento AS D
LEFT JOIN profesor AS PR ON D.id = PR.id_departamento
WHERE PR.id_profesor IS NULL;

```

The screenshot shows the pgAdmin 4 interface with the 'universidad/postgres@PostgreSQL 15' connection selected. The Object Explorer on the left lists various database objects like Functions, Materialized Views, Operators, Procedures, Sequences, and Tables. The 'Tables (7)' section is expanded, showing 'alumno\_se\_matricula\_asignatura', 'asignatura', 'curso\_escolar', 'departamento', 'grado', 'persona', and 'profesor'. The central pane displays the results of a query on the 'departamento' table:

|   | id_departamento | nombre              |
|---|-----------------|---------------------|
| 1 | 8               | Derecho             |
| 2 | 9               | Biología y Geología |
| 3 | 7               | Filología           |

Below the table, a message indicates: "Successfully run. Total query runtime: 98 msec. 3 rows affected." The status bar at the bottom right shows "Ln 88, Col 1" and the date "23/06/2023".

This screenshot is nearly identical to the one above, showing the pgAdmin 4 interface with the 'universidad/postgres@PostgreSQL 15' connection selected. The Object Explorer and the results for the 'departamento' table are the same. The message at the bottom of the central pane is "Successfully run. Total query runtime: 98 msec. 3 rows affected." The status bar at the bottom right shows "Ln 88, Col 1" and the date "23/06/2023".

4. Devuelve un listado con los profesores que tienen un departamento asociado y que no imparten ninguna asignatura.

```

SELECT PR.id_profesor, P.nombre, P.apellido1, P.apellido2
FROM profesor AS PR
INNER JOIN persona AS P ON PR.id_profesor = P.id
LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
WHERE PR.id_departamento IS NOT NULL
AND A.id IS NULL;
    
```

pgAdmin 4

File Object Tools Help

Object Explorer

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#)

Query History Data Output Notifications Messages

```

82
83 SELECT P.id AS id_profesor, P.nombre, P.apellido1, P.apellido2
84 FROM profesor AS P
85 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
86 WHERE PR.id_departamento IS NULL;
87
88 SELECT D.id AS id_departamento, D.nombre
89 FROM departamento AS D
90 LEFT JOIN profesor AS PR ON D.id = PR.id_departamento
91 WHERE PR.id_profesor IS NULL;
92
93 SELECT PR.id_profesor, P.nombre, P.apellido1, P.apellido2
94 FROM profesor AS PR
95 INNER JOIN persona AS P ON PR.id_profesor = P.id
96 LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
97 WHERE PR.id_departamento IS NOT NULL
98 AND A.id IS NULL;
99
100
101
102
103
104

```

Total rows: 10 of 10 Query complete 00:00:00.145 Ln 93, Col 1

pgAdmin 4

File Object Tools Help

Object Explorer

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#)

Query History Data Output Notifications Messages

|    | id_profesor | nombre    | apellido1  | apellido2  |
|----|-------------|-----------|------------|------------|
| 1  | 20          | Francesca | Schowalter | Muller     |
| 2  | 17          | Guillermo | Ruecker    | Upton      |
| 3  | 12          | Carmen    | Streich    | Hirthe     |
| 4  | 10          | Esther    | Spencer    | Lakin      |
| 5  | 18          | Micaela   | Monahan    | Murray     |
| 6  | 15          | Alejandro | Kohler     | Schoen     |
| 7  | 13          | Alfredo   | Stiedemann | Morisette  |
| 8  | 5           | David     | Schmidt    | Fisher     |
| 9  | 8           | Cristina  | Lemke      | Rutherford |
| 10 | 16          | Antonio   | Fahey      | Considine  |

Total rows: 10 of 10 Query complete 00:00:00.145 Successfully run. Total query runtime: 145 msec. 10 rows affected. Ln 93, Col 1

5. Devuelve un listado con las asignaturas que no tienen un profesor asignado.

```

SELECT A.id AS id_asignatura, A.nombre AS nombre_asignatura
FROM asignatura AS A
LEFT JOIN profesor AS P ON A.id_profesor = P.id_profesor
WHERE P.id_profesor IS NULL;

```

pgAdmin 4

Object Explorer

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#)

Query Query History Data Output Notifications Messages

Columns (8)

|    | id_asignatura | nombre_asignatura  |
|----|---------------|--|
| 1  | 22            | Ingeniería de Requisitos   |
| 2  | 23            | Integración de las Tecnologías de la Información en las Organizaciones.. |
| 3  | 24            | Modelado y Diseño del Software 1   |
| 4  | 25            | Multiprocesadores  |
| 5  | 26            | Seguridad y cumplimiento normativo                                       |
| 6  | 27            | Sistema de Información para las Organizaciones                           |
| 7  | 28            | Tecnologías web  |
| 8  | 29            | Teoría de códigos y criptografía   |
| 9  | 30            | Administración de bases de datos   |
| 10 | 31            | Herramientas y Métodos de Ingeniería del Software                        |
| 11 | 32            | Informática industrial y robótica  |
| 12 | 33            | Ingeniería de Sistemas de Información                                    |
| 13 | 34            | Modelado y Diseño del Software 2   |
| 14 | 35            | Negocio Electrónico  |
| 15 | 36            | Periféricos e interfaces   |

Total rows: 62 of 62 Query complete 00:00:00.218

Búsqueda ESP LAA 9:03 p. m. 23/06/2023 Ln 101, Col 1

pgAdmin 4

Object Explorer

Properties SQL Processes [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#) [universidad/postgres@PostgreSQL 15](#)

Query Query History Data Output Notifications Messages

Columns (8)

|    | id_asignatura | nombre_asignatura  |
|----|---------------|--|
| 1  | 22            | Ingeniería de Requisitos   |
| 2  | 23            | Integración de las Tecnologías de la Información en las Organizaciones.. |
| 3  | 24            | Modelado y Diseño del Software 1   |
| 4  | 25            | Multiprocesadores  |
| 5  | 26            | Seguridad y cumplimiento normativo                                       |
| 6  | 27            | Sistema de Información para las Organizaciones                           |
| 7  | 28            | Tecnologías web  |
| 8  | 29            | Teoría de códigos y criptografía   |
| 9  | 30            | Administración de bases de datos   |
| 10 | 31            | Herramientas y Métodos de Ingeniería del Software                        |
| 11 | 32            | Informática industrial y robótica  |
| 12 | 33            | Ingeniería de Sistemas de Información                                    |
| 13 | 34            | Modelado y Diseño del Software 2   |
| 14 | 35            | Negocio Electrónico  |
| 15 | 36            | Periféricos e interfaces   |

Total rows: 62 of 62 Query complete 00:00:00.218

Búsqueda ESP LAA 9:03 p. m. 23/06/2023 Ln 101, Col 1

- Devuelve un listado con todos los departamentos que no han impartido asignaturas en ningún curso escolar.

```

SELECT D.id AS id_departamento, D.nombre AS nombre_departamento
FROM departamento AS D
LEFT JOIN profesor AS P ON D.id = P.id_departamento
LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
GROUP BY D.id, D.nombre
    
```

HAVING COUNT(A.id) = 0;

The screenshot shows the pgAdmin 4 interface with the 'Query' tab selected. The query window contains the following SQL code:

```
91 WHERE PR.id_profesor IS NULL;
92
93 SELECT PR.id_profesor, P.nombre, P.apellido1, P.apellido2
94 FROM profesor AS PR
95 LEFT JOIN asignatura AS A ON PR.id_profesor = A.id_profesor
96 WHERE PR.id_departamento IS NOT NULL
97 AND A.id IS NULL;
98
99
100
101 SELECT A.id AS id_asignatura, A.nombre AS nombre_asignatura
102 FROM asignatura AS A
103 LEFT JOIN profesor AS P ON A.id_profesor = P.id_profesor
104 WHERE P.id_profesor IS NULL;
105
106 SELECT D.id AS id_departamento, D.nombre AS nombre_departamento
107 FROM departamento AS D
108 LEFT JOIN profesor AS P ON D.id = P.id_departamento
109 LEFT JOIN asignatura AS A ON P.id_profesor = A.id_profesor
110 GROUP BY D.id, D.nombre
111 HAVING COUNT(A.id) = 0;
112
113
```

Total rows: 8 of 8    Query complete 00:00:00.292    Ln 106, Col 1

The screenshot shows the pgAdmin 4 interface with the 'Data Output' tab selected. The results pane displays the following data from the 'departamento' table:

|   | id_departamento | nombre_departamento |
|---|-----------------|---------------------|
| 1 | 2               | Derecho             |
| 2 | 7               | Filología           |
| 3 | 9               | Biología y Geología |
| 4 | 5               | Agronomía           |
| 5 | 2               | Matemáticas         |
| 6 | 4               | Educación           |
| 7 | 6               | Química y Física    |
| 8 | 3               | Economía y Empresa  |

Total rows: 8 of 8    Query complete 00:00:00.292    Successfully run. Total query runtime: 292 msec. 8 rows affected.    Ln 106, Col 1