

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS



FUNDAMENTOS DE BASES DE DATOS

Práctica 09

Equipo: eSosQLones

Estrada Garcia Luis Gerardo	319013832
Jiménez Hernández Allan	420003478
Mancera Quiroz Javier Alejandro	319274831
Mora Hernández Dulce Julieta	319236448
Peña Nuñez Axel Yael	318279754

1.- Los veterinarios y cuidadores que comparten el apellido.

```
SELECT P.*
FROM persona AS P
WHERE P.apellidopat IN (
    SELECT P.apellidopat
    FROM persona AS P
    WHERE P.id_persona IN (
        SELECT T.id_persona
        FROM trabajador AS T
        WHERE T.rfc IN (
            SELECT rfc
            FROM trabajador
            INTERSECT
            SELECT rfc
            FROM veterinario
        )
    )
)
INTERSECT
SELECT P.apellidopat
FROM persona AS P
WHERE P.id_persona IN (
    SELECT T.id_persona
    FROM trabajador AS T
    WHERE T.rfc IN (
        SELECT rfc
        FROM trabajador
        INTERSECT
        SELECT rfc
        FROM cuidador
    )
)
);
```

Explicación: Selecciona todas las personas cuyos apellidos paternos están presentes en ambos grupos: trabajadores que son veterinarios y trabajadores que son cuidadores.

1. Selecciona los *apellidos paternos* de las personas.
2. Verifica si estas personas también están en la tabla *trabajadores*.
3. De estos trabajadores, filtra aquellos cuyo *RFC* también aparece en los veterinarios.
4. Realiza un procedimiento similar para identificar trabajadores que son cuidadores.
5. Usa *INTERSECT* para encontrar apellidos paternos comunes a ambos grupos.
6. Finalmente, selecciona todas las entradas en la tabla *persona* con estos apellidos paternos.

2.- Los animales herbívoros que pertenezcan al bioma bosque templado.

```
SELECT A.*
FROM animal AS A
WHERE A.id_animal IN (
    SELECT id_animal
    FROM animal
    WHERE alimentacion ILIKE 'herbivoro'
    INTERSECT
    SELECT id_animal
    FROM bioma
    WHERE tipo ILIKE 'bosque templado'
);
```

Explicación: Esta consulta SQL tiene como objetivo seleccionar todos los animales que son herbívoros y que además pertenecen al bioma 'bosque templado'. La consulta se realiza en dos pasos:

1. Selecciona los *id_animal* de la tabla *animal* donde la *alimentación* es herbívora (usando *ILIKE* para una coincidencia de patrón insensible a mayúsculas y minúsculas).
2. Selecciona los *id_animal* de la tabla *bioma* donde el *tipo* es 'bosque templado', aplicando la misma lógica de coincidencia de patrón.
3. Utiliza *INTERSECT* para encontrar los *id_animal* que están presentes en ambos conjuntos, es decir, animales que son herbívoros y que están en el bioma 'bosque templado'.
4. Finalmente, selecciona todas las entradas en la tabla *animal* con estos *id_animal*.

3.- Alimentos de tipo semilla, que son distribuidos en el aviario.

```
SELECT A.*
FROM alimento AS A
WHERE A.id_insumo IN (
    SELECT id_insumo
    FROM alimento
    INTERSECT
    SELECT D.id_insumo
    FROM distribuir AS D
    WHERE D.id_insumo IN (
        SELECT id_insumo
        FROM Alimento
        WHERE tipoAlimento ILIKE 'semilla%'
        INTERSECT
        SELECT id_insumo FROM distribuir
    )
    AND D.id_bioma IN (
        SELECT id_bioma
        FROM bioma
```

```

        WHERE tipo ILIKE 'aviario'
        INTERSECT
        SELECT id_bioma FROM distribuir
    )
)
INTERSECT
SELECT *
FROM Alimento
WHERE tipoAlimento ILIKE 'semilla%';

```

Explicación: Esta consulta SQL busca seleccionar todos los alimentos de tipo 'semilla' que son distribuidos en el aviario. La consulta sigue varios pasos:

1. Identifica los *id_insumo* de la tabla *alimento* que también están presentes en la tabla *distribuir*.
2. Dentro de la tabla *distribuir*, filtra aquellos *id_insumo* que corresponden a alimentos tipo 'semilla' y aquellos que están asociados con el bioma 'aviario'.
3. Utiliza *INTERSECT* para encontrar los *id_insumo* comunes entre estos dos conjuntos.
4. Se realiza una última intersección para asegurarse de que los alimentos seleccionados sean efectivamente de tipo 'semilla'.
5. Finalmente, selecciona todas las entradas en la tabla *alimento* que cumplen con estas condiciones.

4.- Los cuidadores y proveedores que vivan en el mismo estado.

```

SELECT P.*
FROM persona AS P
WHERE P.estado IN (
    SELECT P.estado
    FROM persona AS P
    WHERE P.id_persona IN (
        SELECT T.id_persona
        FROM trabajador AS T
        WHERE T.rfc IN (
            SELECT rfc
            FROM trabajador
            INTERSECT
            SELECT rfc
            FROM proveedor
        )
    )
)
INTERSECT
SELECT P.estado
FROM persona AS P
WHERE P.id_persona IN (

```

```

SELECT T.id_persona
FROM trabajador AS T
WHERE T.rfc IN (
    SELECT rfc
    FROM trabajador
    INTERSECT
    SELECT rfc
    FROM cuidador
)
);

```

Explicación: Esta consulta SQL se enfoca en seleccionar todas las personas que tienen estados asociados tanto a trabajadores que son proveedores como a trabajadores que son cuidadores. Los pasos son:

1. Selecciona los *estados* de personas en la tabla *persona* donde el *id_persona* corresponde a trabajadores que también son proveedores.
2. Realiza un proceso similar para seleccionar *estados* de personas donde el *id_persona* corresponde a trabajadores que también son cuidadores.
3. Utiliza *INTERSECT* para encontrar los *estados* comunes entre estos dos grupos.
4. Finalmente, selecciona todas las entradas en la tabla *persona* que tienen estos estados.

5.- Biomas que posean al menos 10 animales.

```

SELECT tipo, COUNT(*) as num_animales
FROM bioma
GROUP BY tipo
HAVING COUNT(*) >= 10;

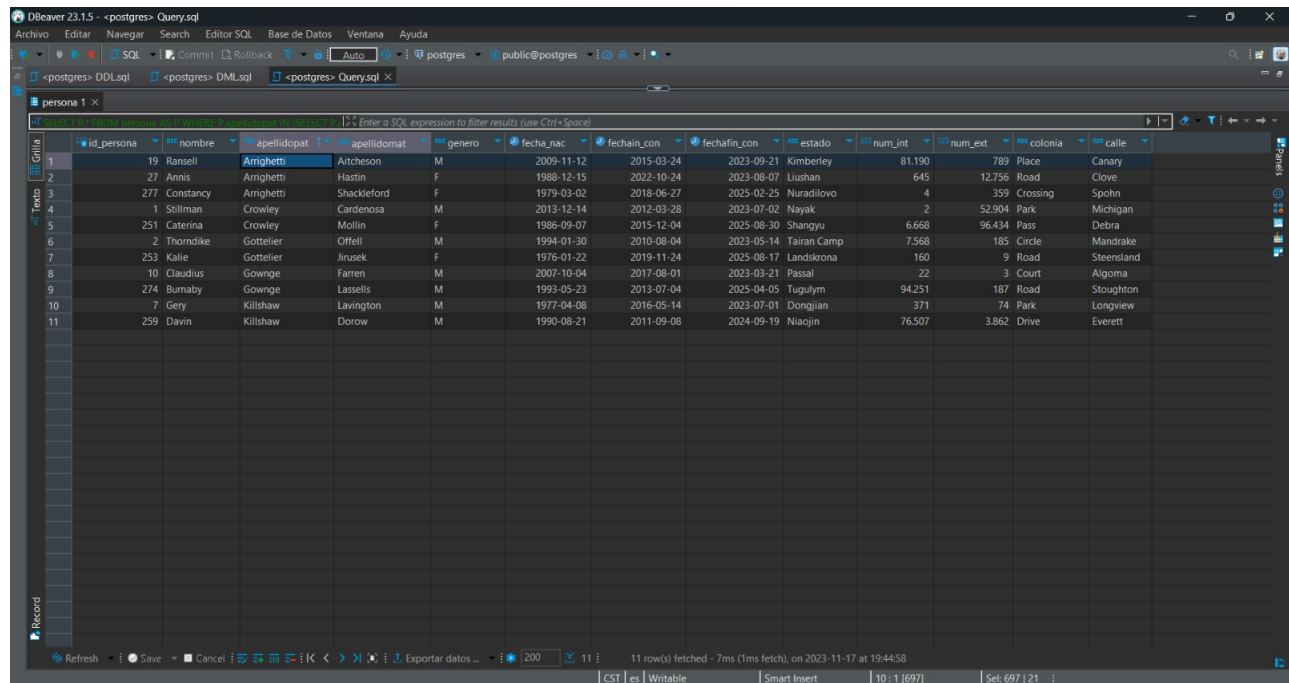
```

Explicación: Esta consulta SQL está diseñada para identificar los biomas que tienen al menos 10 animales. La consulta se realiza de la siguiente manera:

1. Selecciona el *tipo* de bioma de la tabla *bioma*.
2. Utiliza la función *COUNT(*)* para contar el número de animales asociados a cada tipo de bioma.
3. Agrupa los resultados por el *tipo* de bioma usando *GROUP BY*.
4. Aplica la cláusula *HAVING* para filtrar solo aquellos tipos de biomas que tienen 10 o más animales (*COUNT(*) >= 10*).
5. El resultado muestra el tipo de bioma y el número total de animales asociados a cada tipo que cumple con el criterio de tener al menos 10 animales.

Evidencias

```
8
9-- 1) Los veterinarios y cuidadores que comparten el apellido.
10 SELECT P.*
11 FROM persona AS P
12 WHERE P.apellidopat IN (SELECT P.apellidopat
13                          from persona as P
14                          where P.id_persona in (select T.id_persona
15                                                  from trabajador as T
16                                                  where T.rfc in (select rfc
17                                                            from trabajador
18                                                            intersect
19                                                            select rfc
20                                                            from veterinario)))
21
22 intersect
23 SELECT P.apellidopat
24 from persona as P
25 where P.id_persona in (select T.id_persona
26                       from trabajador as T
27                       where T.rfc in (select rfc
28                                       from trabajador
29                                       intersect
30                                       select rfc
31                                       from cuidador)))
```



The screenshot shows the DBeaver 23.1.5 interface with a PostgreSQL database. The query results are displayed in a table with 11 rows and 13 columns. The columns are: id_persona, nombre, apellido, apellido_m, genero, fecha_nac, fecha_nac_con, fecha_fin_con, estado, num_int, num_ext, colonia, and calle. The data is as follows:

	id_persona	nombre	apellido	apellido_m	genero	fecha_nac	fecha_nac_con	fecha_fin_con	estado	num_int	num_ext	colonia	calle
1	19	Ransell	Arrighetti	Aitcheson	M	2009-11-12	2015-03-24	2023-09-21	Kimberley	81.190	789	Place	Canary
2	27	Annis	Arrighetti	Hastin	F	1988-12-15	2022-10-24	2023-08-07	Liushan	645	12.756	Road	Clove
3	277	Constancy	Arrighetti	Shackleford	F	1979-03-02	2018-06-27	2025-02-25	Nuradilovo	4	359	Crossing	Spohn
4	1	Stillman	Crowley	Cardenosa	M	2013-12-14	2012-03-28	2023-07-02	Nayak	2	52.904	Park	Michigan
5	251	Caterina	Crowley	Mollin	F	1986-09-07	2015-12-04	2025-08-30	Shangyu	6.668	96.434	Pass	Debra
6	2	Thorndike	Gottelier	Offell	M	1994-01-30	2010-08-04	2023-05-14	Tairan Camp	7.568	185	Circle	Mandrake
7	253	Kalie	Gottelier	Jirusek	F	1976-01-22	2019-11-24	2025-08-17	Landskrona	160	9	Road	Steensland
8	10	Claudius	Gowngie	Farren	M	2007-10-04	2017-08-01	2023-03-21	Passal	22	3	Court	Algoma
9	274	Burnaby	Gowngie	Lassells	M	1993-05-23	2013-07-04	2025-04-05	Tugulym	94.251	187	Road	Stoughton
10	7	Gery	Killshaw	Lavington	M	1977-04-08	2016-05-14	2023-07-01	Dongjian	371	74	Park	Longview
11	259	Davin	Killshaw	Dorow	M	1990-08-21	2011-09-08	2024-09-19	Niaojin	76.507	3.862	Drive	Everett

```

101-- 2) Los animales herbivoros que pertenezcan al bioma bosque templado.
102 SELECT A.*
103 FROM animal as A
104 WHERE A.id_animal in (select id_animal
105                        from animal
106                        where alimentacion ilike 'herb_voro'
107                        intersect
108                        select id_animal
109                        from bioma
110                        where tipo ilike 'bosque templado');
111

```

DBaiver 23.1.5 - <postgres> Query.sql

Archivo Editar Navegar Search Editor SQL Base de Datos Ventana Ayuda

SQL Commit Rollback Auto postgres public@postgres

<postgres> DDL.sql <postgres> DML.sql <postgres> Query.sql

animal 1 x

Enter a SQL expression to filter results (use Ctrl+Space)

	id_animal	rfc	num_jaula	nombre	alimentacion	sexo	altura	peso	especie
1	3	P6xF2N07vsPo	3	Charissa	Herbivoro	F	21.2	5.011	León
2	58	HQ2XtB0U8vSv	58	Maniquilla	Herbivoro	F	1.8	3.123	Rinoceronte Blanco
3	88	VqPIIPYwBZoI	88	Benita	Herbivoro	F	22.7	2.559	Nutria
4	93	513Fcgt2h6C	93	Vinson	Herbivoro	M	27.5	3.227	Lagarto
5	101	P6xF2N07vsPo	3	Charissa	Herbivoro	F	21.2	5.011	León

Refresh Save Cancel Exportar datos 200 5 5 row(s) fetched - 5ms, on 2023-11-17 at 19:48:13

CST es Writable Smart Insert 102:1 [244] Sel: 244 | 9

```

135
136-- 3) Alimentos de tipo semilla, que son distribuidos en el aviario.
137 select A.*
138 from alimento as A
139 where A.id_insumo in (select id_insumo
140                       from alimento
141                       intersect
142                       select D.id_insumo
143                       from distribuir as D
144                       where D.id_insumo in (SELECT id_insumo
145                                           FROM Alimento
146                                           WHERE tipoAlimento ilike 'semilla%'
147                                           intersect
148                                           SELECT id_insumo FROM distribuir)
149                       and
150                       D.id_bioma in (SELECT id_bioma
151                                     FROM bioma
152                                     WHERE tipo ilike 'aviario'
153                                     intersect
154                                     SELECT id_bioma FROM distribuir))
155 intersect
156 SELECT *
157 FROM Alimento
158 WHERE tipoAlimento ilike 'semilla%';
159

```

DBBeaver 23.1.5 -- <postgres> Query.sql

Archivo Editar Navegar Search Editor SQL Base de Datos Ventana Ayuda

SQL Commit Rollback Auto postgres public@postgres

Resultados 1 x

id_insumo tipoalimento

1	43	semilla
2	5	semillas de calabaza
3	5	semilla
4	22	semilla
5	20	semilla
6	81	semilla
7	60	semilla
8	62	semilla

8 row(s) fetched - 4ms, on 2023-11-17 at 19:54:33

CS | es | Writable | Smart Insert | 135 : 1 : 3881 | Set: 0 | 0


```

200
201 -- 4) Los cuidadores y proveedores que vivan en el mismo estado.
202 select P.*
203 from persona as P
204 where P.estado in (SELECT P.estado
205                    from persona as P
206                    where P.id_persona in (select T.id_persona
207                                           from trabajador as T
208                                           where T.rfc in (select rfc
209                                                         from trabajador
210                                                         intersect
211                                                         select rfc
212                                                         from proveedor)))
213                    intersect
214                    SELECT P.estado
215                    from persona as P
216                    where P.id_persona in (select T.id_persona
217                                           from trabajador as T
218                                           where T.rfc in (select rfc
219                                                         from trabajador
220                                                         intersect
221                                                         select rfc
222                                                         from cuidador)))));
223

```

DBaiver 23.1.5 - <postgres> Query.sql

Archivo Editar Navegar Search Editor SQL Base de Datos Ventana Ayuda

SQL Commit Rollback Auto postgres public@postgres

personas > DDL.sql <postgres> DML.sql <postgres> Query.sql

persona 1 x

Enter a SQL expression to filter results (use Ctrl+Space)

	id_persona	nombre	apellido1	apellido2	genero	fecha_nac	fecha_in_con	fecha_fin_con	estado	num_int	num_ext	colonia	calle
1	163	Denis	Deble	Cuncliffe	M	1977-06-26	2022-01-15	2024-02-12	Chixi	58	8.538	Way	Sheridan
2	167	Arabel	Oughtright	De Roos	F	1991-01-06	2014-04-23	2025-01-24	Chixi	33.754	806	Center	Holmberg
3	243	Cory	Daughton	Coate	M	1998-01-30	2022-01-01	2024-06-26	Chixi	5	25	Circle	Tomscoot
4	258	Ermina	Hancom	Faro	F	1977-04-17	2018-03-09	2024-04-30	Chixi	457	321	Center	Garrison
5	263	Tann	Antsyhev	Kamall	M	1991-12-31	2014-06-06	2025-02-16	Chixi	2	280	Park	Scott
6	291	Benito	Harriman	Hurley	M	1990-02-05	2011-09-10	2023-07-18	Chixi	7	15	Hill	Quincy
7	171	Felisha	Sommerling	Cumbers	F	1988-10-29	2019-05-18	2024-04-29	Dasuk	15	948	Trail	Holmberg
8	255	Malorie	Rolling	Sawdy	F	2002-07-25	2016-05-07	2023-07-28	Dasuk	83.615	83.371	Drive	Heffernan
9	256	Tallor	Cicciotti	Eades	M	1970-08-23	2020-03-21	2023-07-29	Dasuk	7	517	Center	Jana
10	176	Kasey	Heaselgrave	Gladding	F	1981-09-13	2012-08-24	2025-06-26	Niaojin	85	1	Park	Stone Corner
11	178	Sigfried	Agglio	Mountfort	M	1998-08-15	2014-07-20	2025-06-09	Niaojin	813	3.302	Plaza	Dorton
12	232	Velma	Chidwick	Kerslake	F	1976-06-23	2016-10-29	2023-08-18	Niaojin	905	16	Plaza	Chinook
13	236	Bliss	Hars	Lowers	F	2001-01-15	2016-10-27	2023-02-15	Niaojin	3	39.193	Terrace	Vidon
14	259	Davin	Killshaw	Dorow	M	1990-08-21	2011-09-08	2024-09-19	Niaojin	76.507	3.862	Drive	Everett
15	151	Halsey	Westcarr	Vaulkhard	M	2001-08-24	2021-06-14	2023-08-09	Ourozinho	6	651	Point	Cardinal
16	155	Catiee	McEvay	Puleque	F	1981-12-15	2019-09-26	2025-06-09	Ourozinho	177	2	Parkway	Redwing
17	250	Aaren	Mcfeen	Skocroft	F	2001-02-05	2019-08-13	2023-06-17	Ourozinho	22.926	826	Crossing	Charing Cross
18	252	Wriatt	Akrigg	Clibbery	M	1971-09-13	2020-12-18	2023-01-31	Ourozinho	72	766	Point	Old Shore
19	153	Walsh	Chilton	Hand	M	1976-02-25	2013-10-31	2025-10-23	Shangyu	4	98	Junction	Holy Cross
20	251	Caterina	Crowley	Mollin	F	1986-09-07	2015-12-04	2025-08-30	Shangyu	6.668	96.434	Pass	Debra

20 row(s) fetched - 8ms (2ms fetch), on 2023-11-17 at 19:56:39

Refresh Save Cancel Exportar datos... 200 CST es Writable Smart Insert 202:1 [698] Set: 698 | 21

```
256
257-- 5) Biomas que posean al menos 10 animales.
258 SELECT tipo, COUNT(*) as num_animales
259 FROM bioma
260 GROUP by tipo having COUNT(*)>=10
261
```

bioma 1 x

	tipo	num_animales
1	Selva	12
2	Pradera	13
3	Tundra ártica	15
4	Bosque templado	10
5	Desierto	15

5 row(s) fetched - 4ms, on 2023-11-17 at 19:58:06