

## Registro de resultados en DBeaver

### Desafío - Definición de tabla.

Estudiante: *Valeria Fariña R.*

### Requerimientos

1. Crear una base de datos llamada películas.
2. Cargar ambos archivos a su tabla correspondiente. Nombre de las tablas: reparto y películas.
3. Obtener el ID de la película "Titanic".

The screenshot shows the DBeaver 24.0.4 interface. On the left, the 'Database Navigator' pane shows a tree structure with 'películas' selected, and 'reparto' and 'películas' tables listed under the 'public' schema. The 'SQL Editor' pane on the right contains the following SQL query:

```
SELECT id, "Pelicula"
FROM public.películas
WHERE "Pelicula" = 'Titanic'
```

Below the SQL editor, the 'Results' pane shows a table with the following data:

id	Pelicula
1	Titanic

4. Listar a todos los actores que aparecen en la película "Titanic".  
Considerar id\_película = película\_id.

DBeaver 24.0.4 - <localhost> public.peliculas

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto localh

Database Navig... Projects

Enter a part of object name here

- películas
  - Schemas
    - public
      - Tables
        - películas 8K
        - reparto 88K
        - Foreign Tables
        - Views
        - Materialized Views
        - Indexes
        - Functions
        - Sequences
        - Data types
        - Aggregate functions
      - Event Triggers
      - Extensions
      - Storage
      - System Info
      - Roles
- postgres

películas \*<localhost> publi... public public

```
SELECT p."Película" AS nombre_película, r.actor
FROM películas p
INNER JOIN reparto r ON p.id = r.id_película
WHERE p."Película" = 'Titanic';
```

películas(+) 1

```
SELECT p."Película" AS nombre_película, r.actor FROM películas p INNER
```

	ABC nombre_película	ABC actor
1	Titanic	Leonardo DiCaprio
2	Titanic	Kate Winslet
3	Titanic	Billy Zane
4	Titanic	Kathy Bates
5	Titanic	Frances Fisher
6	Titanic	Bernard Hill
7	Titanic	Jonathan Hyde
8	Titanic	Danny Nucci
9	Titanic	David Warner
10	Titanic	Bill Paxton
11	Titanic	Gloria Stuart
12	Titanic	Victor Garber
13	Titanic	Suzy Amis

Project - General

Name DataSource

- Bookmarks
- Dashboards
- Diagrams
- Scripts

5. Consultar en cuántas películas del top 100 participa Harrison Ford.  
Considerar id\_pelicula = pelicula\_id

DBaver 24.0.4 - <localhost> public.reparto

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto localhost public@películas

Database Navig... Projects

Enter a part of object name here

- películas
  - Schemas
    - public
      - Tables
        - películas 8K
        - reparto 88K
      - Foreign Tables
      - Views
      - Materialized Views
      - Indexes
      - Functions
      - Sequences
      - Data types
      - Aggregate functions
      - Event Triggers
      - Extensions
      - Storage
      - System Info
      - Roles
- postgres

Project - General

Name DataSource

Bookmarks Dashboards Diagrams Scripts

```
WITH Total_Peliculas AS (
  SELECT
    p."Pelicula" AS nombre_pelicula,
    COUNT(*) OVER () AS total_peliculas
  FROM
    public.reparto r
  INNER JOIN
    public.películas p ON r.id_pelicula = p.id
  WHERE
    r.actor = 'Harrison Ford'
  GROUP BY
    p."Pelicula"
)
SELECT
  ROW_NUMBER() OVER (ORDER BY nombre_pelicula) AS n°_del_total,
  tp.total_peliculas,
  tp.nombre_pelicula
FROM
  Total_Peliculas tp;
```

películas 1

WITH Total\_Peliculas AS ( SELECT p."Pelicula" AS nombre\_pelicula, COUNT(\*) OVER () AS total\_peliculas

	123 n°_del_total	123 total_peliculas	ABC nombre_pelicula
1	1	8	Apocalypse Now
2	2	8	Blade Runner
3	3	8	En busca del arca perdida
4	4	8	Indiana Jones y el templo maldito
5	5	8	Indiana Jones y la última cruzada
6	6	8	Star Wars. Episodio IV: Una nueva esperanza
7	7	8	Star Wars. Episodio V: El imperio contraataca
8	8	8	Star Wars. Episodio VI: El retorno del Jedi

6. Indicar las películas estrenadas entre los años 1990 y 1999 ordenadas por título de manera ascendente.

DBEaver 24.0.4 - <localhost> public.peliculas

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto localhost

Database Navig... Projects

Enter a part of object name here

- películas
  - Schemas
    - public
      - Tables
        - películas 8K
        - reparto 88K
      - Foreign Tables
      - Views
      - Materialized Views
      - Indexes
      - Functions
      - Sequences
      - Data types
      - Aggregate functions
    - Event Triggers
    - Extensions
    - Storage
    - System Info
    - Roles
  - postgres

Project - General

Name DataSource

- Bookmarks
- Dashboards
- Diagrams
- Scripts

películas public public películas reparto

```
SELECT "Película", "Año estreno"
FROM public.peliculas
WHERE "Año estreno" >= 1990 and "Año estreno" < 1999
ORDER BY "Película";
```

películas 1

```
SELECT "Película", "Año estreno" FROM public.peliculas WHERE "Año estreno" >= 1990 and "Año estreno" < 1999
ORDER BY "Película";
```

	ABC Película	123 Año estreno
1	American History X	1,998
2	Braveheart	1,995
3	Cadena perpetua	1,994
4	Eduardo Manostijeras	1,990
5	El Padrino. Parte III	1,990
6	El profesional	1,994
7	El show de Truman	1,998
8	El silencio de los corderos	1,991
9	Entrevista con el vampiro	1,994
10	Forest Gump	1,994
11	Hombres de negro	1,997
12	La lista de Schindler	1,993
13	Mejor... imposible	1,997
14	Parque Jurásico	1,993
15	Pesadilla antes de navidad	1,993
16	Pulp Fiction	1,994
17	Salvar al soldado Ryan	1,998
18	Seven	1,995
19	Terminator 2: el juicio final	1,991
20	Titanic	1,997
21	Toy Story	1,995
22	Trainspotting	1,996
23	Uno de los nuestros	1,990

7. Hacer una consulta SQL que muestre los títulos con su longitud, la longitud debe ser nombrado para la consulta como "longitud\_titulo".

WITH Peliculas\_longitud AS (

```

SELECT
    "Pelicula",
    LENGTH("Pelicula") AS longitud_titulo
FROM
    public.peliculas
)
SELECT
    "Pelicula",
    longitud_titulo
FROM
    Peliculas_longitud
ORDER BY
    longitud_titulo ASC;

```

películas 1

WITH Peliculas\_longitud AS ( SELECT "Pelicula", LENGTH("Pelicula") AS longitud\_titulo )

id	Pelicula	longitud_titulo
1	E.T	3
2	300	3
3	Saw	3
4	Seven	5
5	Rocky	5
6	Alien	5
7	Rambo	5
8	Grease	6
9	Wall-E	6
10	Amélie	6
11	Matrix	6
12	Titanic	7
13	X-Men 2	7
14	Tiburón	7
15	Ben-Hur	7
16	Iron Man	8
17	Psicosis	8
18	Predator	8
19	King Kong	9
20	Toy Story	9
21	Gladiator	9
22	Casablanca	10
23	Terminator	10
24	Braveheart	10
25	Spider-Man	10
26	El Padrino	10
27	taxi driver	10
28	Infiltrados	11
29	Ratatouille	11
30	Toy Story 2	11
31	Forest Gump	11
32	El exorcista	12
33	Transformers	12
34	Blade Runner	12
35	Pulp Fiction	12
36	Spider-Man 2	12
37	Mouling Rouge	13
38	Trainspotting	13
39	El resplandor	13
40	Batman Begins	13
41	V de Vendetta	13
42	El profesional	14
43	Apocalypse Now	14
44	Los increíbles	14
45	Ocean's Eleven	14
46	La milla verde	14
47	Monstruos S.A.	14
48	American Beauty	15
49	Parque Jurásico	15
50	Escuela de Rock	15
51	Cadena perpetua	15
52	Buscando a Nemo	15
53	Kill Bill Vol. 1	16
54	El sexto sentido	16
55	Hombres de negro	16
56	Kill Bill Vol. 2	16
57	La vida de Brian	16
58	Crash	17
59	El show de Truman	17
60	Los cazafantasmas	17
61	Regreso al futuro	17
62	Mejor... imposible	18
63	American History X	18
64	El viaje de Chihiro	19
65	El club de la pelea	19
66	El precio del poder	19
67	Uno de los nuestros	19
68	El caballero oscuro	19
69	Eduardo Manostijie	20
70	El libro de la selva	20
71	El Padrino. Parte II	20
72	Regreso al futuro II	20
73	El Padrino. Parte III	21
74	La lista de Schindler	21
75	El Laberinto del Faun	22
76	Salvar al soldado Ryan	22
77	El planeta de los simios	24
78	Lo que el viento se llevó	25
79	Entrevista con el vampiro	25
80	En busca del arca perdida	25
81	El bueno, el feo y el malo	25
82	Pesadilla antes de Navidad	26
83	El silencio de los corderos	27
84	Terminator 2: el juicio final	29
85	Indiana Jones y el templo maldito	33
86	Indiana Jones y la última cruzada	33
87	Blade Runner	33
88	Alguien voló sobre un nido de cucarachas	35
89	El Señor de los anillos: Los Anillos del Poder	39
90	Harry Potter y el Príncipe Mestizo	39
91	Star Wars. Episodio I: La Fuerza despierta	42
92	Star Wars. Episodio II: El Ataque de los Clones	43
93	El Señor de los anillos: Los Dos Reinos	43
94	Star Wars. Episodio III: La Rebelión de las Damas	43
95	Star Wars. Episodio IV: Un Nuevo Horizonte	45
96	Piratas del Caribe: El Cofre de la Muerte	46
97	Star Wars. Episodio V: El Imperio Contraataca	48
98	El Señor de los anillos: El Retorno de los Reyes	48
99	Piratas del Caribe: El Baile de las Brujas	50
100	Sweeney Todd: El Barbero de Flota	52

8. Consultar cuál es la longitud más grande entre todos los títulos de las películas.

DBeaver 24.0.4 - <localhost> public.reparto

File Edit Navigate Search SQL Editor Database Window Help

SQL Commit Rollback Auto localhost public@películas

Database Navig... Projects películas public películas reparto \*localhost... \*localhost...

Enter a part of object name here

- películas
  - Schemas
    - public
      - Tables
        - películas 8K
        - reparto 88K
      - Foreign Tables
      - Views
      - Materialized Views
      - Indexes
      - Functions
      - Sequences
      - Data types
      - Aggregate functions
    - Event Triggers
    - Extensions
    - Storage
    - System Info
    - Roles
  - postgres

```
WITH Películas_Longitud AS (  
  SELECT  
    "Película",  
    LENGTH("Película") AS longitud_titulo  
  FROM  
    public.películas  
)  
SELECT  
  "Película",  
  longitud_titulo  
FROM  
  Películas_Longitud  
WHERE  
  longitud_titulo = (SELECT MAX(longitud_titulo) FROM Películas_Longitud);
```

películas 1

WITH Películas\_Longitud AS ( SELECT "Película", LENGTH("Película") AS I | Enter a SQL expression to filter result

	ABC Película	123 longitud_titulo
1	Sweeney Todd: El barbero diabólico de la calle Fleet	52