

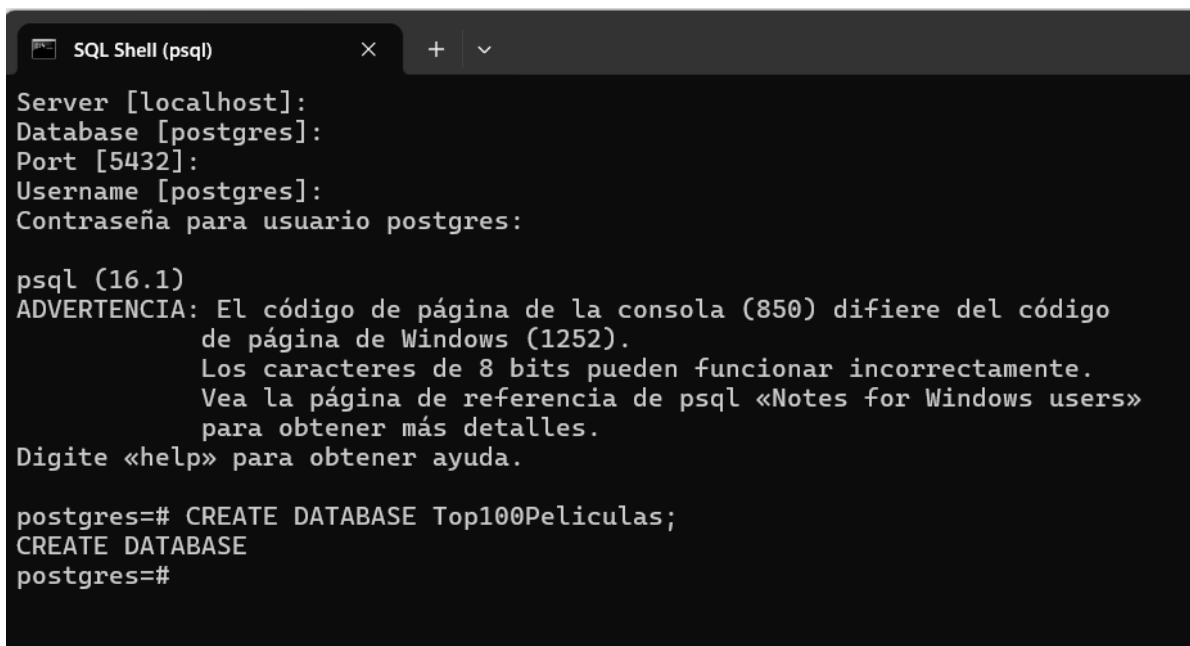
M5 Desafío3 – Definición de Tablas

Fecha: 18-04-2024

1. Crear una base de datos llamada películas.

Para este desafío se crea una base de datos llamada top100películas.

```
CREATE DATABASE top100películas;
```



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Contraseña para usuario postgres:

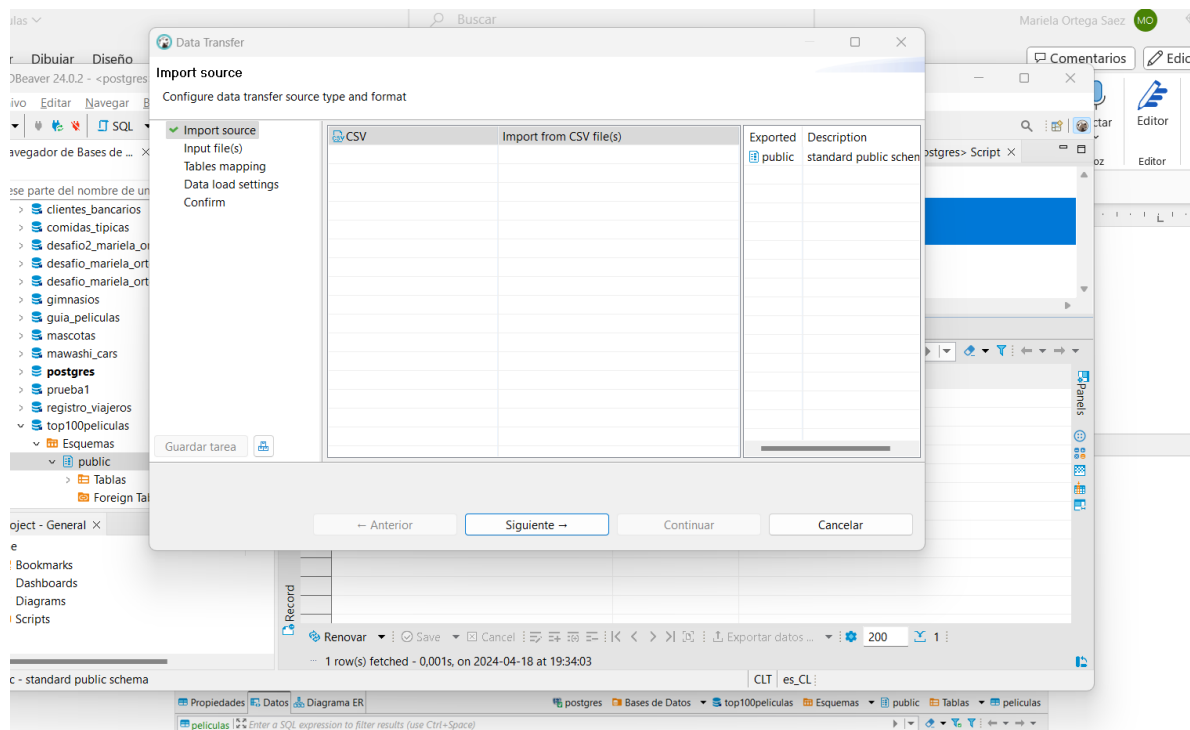
psql (16.1)
ADVERTENCIA: El código de página de la consola (850) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencia de psql «Notes for Windows users»
para obtener más detalles.
Digite «help» para obtener ayuda.

postgres=# CREATE DATABASE Top100Películas;
CREATE DATABASE
postgres=#
```

2. Cargar ambos archivos a su tabla correspondiente

Aplicando los conceptos y herramientas aprendidas hasta ahora y utilizando DBeaver, se importan los datos de los archivos csv películas y reparto.

Evidencia importación de datos para ambas tablas

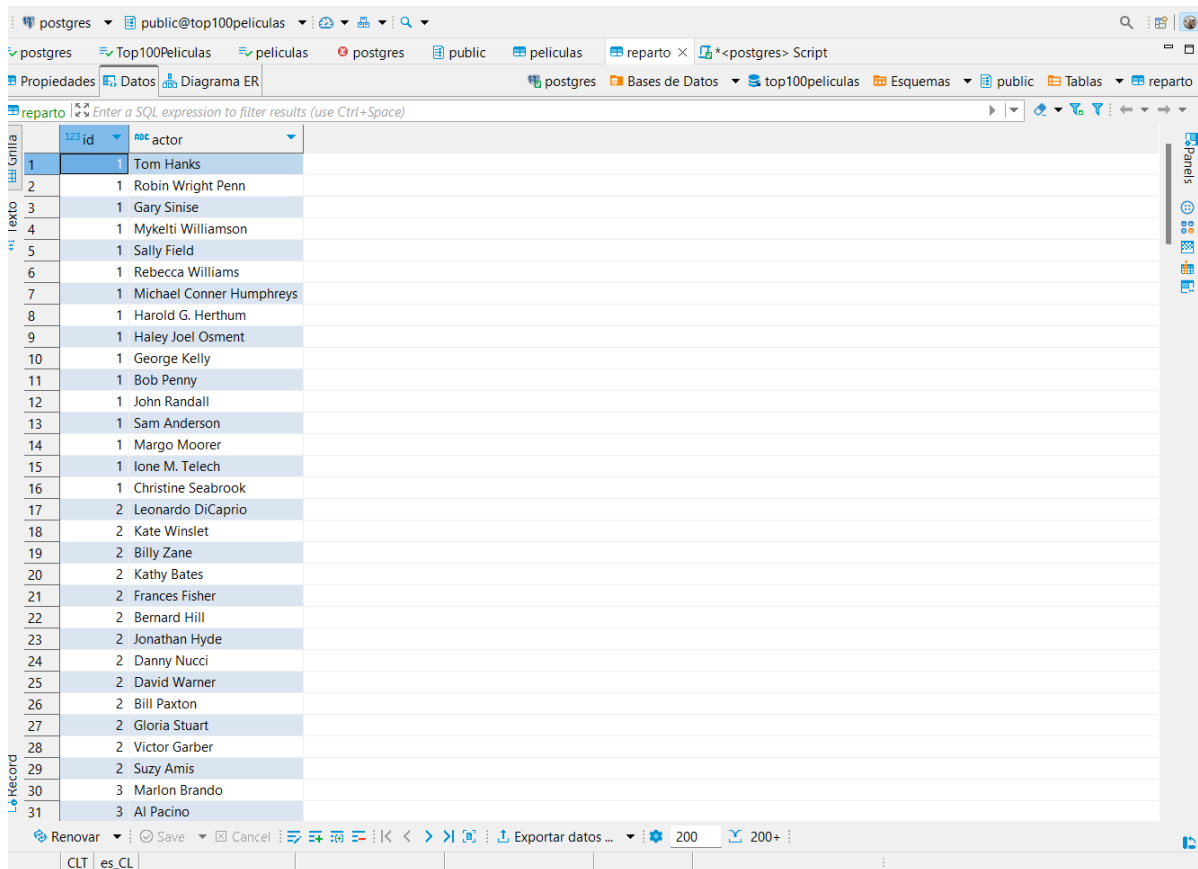


Evidencia de la carga de datos a la tabla película.

The screenshot shows the 'películas' table in DBeaver. The table is displayed in a grid view with the following columns: 'id', 'película', 'año_estreno', and 'Director'. The table contains 31 rows of data.

id	película	año_estreno	Director
1	Forest Gump	1994	Robert Zemeckis
2	Titanic	1997	James Cameron
3	El Padrino	1972	Francis Ford Coppola
4	Gladiator	2000	Ridley Scott
5	El Señor de los anillos: El retorno del rey	2003	Peter Jackson
6	El caballero oscuro	2008	Christopher Nolan
7	Cadena perpetua	1994	Frank Darabont
8	Piratas del Caribe: La maldición de la Perla Negra	2003	Gore Verbinski
9	Braveheart	1995	Mel Gibson
10	La lista de Schindler	1993	Steven Spielberg
11	Toy Story	1995	John Lasseter
12	Eduardo Manostijeras	1990	Tim Burton
13	El Señor de los anillos: La comunidad del anillo	2001	Peter Jackson
14	Salvar al soldado Ryan	1998	Steven Spielberg
15	Regreso al futuro	1985	Robert Zemeckis
16	Monstruos S.A.	2001	Pete Docter
17	Buscando a Nemo	2003	Andrew Stanton
18	El Señor de los anillos: Las dos torres	2002	Peter Jackson
19	Harry Potter y el Prisionero de Azkaban	2004	Alfonso Cuarón
20	American History X	1998	Tony Kaye
21	300	2007	Zack Snyder
22	El sexto sentido	1999	M. Night Shyamalan
23	Pulp Fiction	1994	Quentin Tarantino
24	V de Vendetta	2005	James McTeigue
25	El silencio de los corderos	1991	Jonathan Demme
26	Rocky	1976	John G. Avildsen
27	El club de la pelea	1999	David Fincher
28	E.T.	1982	Steven Spielberg
29	Parque Jurásico	1993	Steven Spielberg
30	Matrix	1999	Andy y Lana Wachowski
31	La milla verde	1999	Frank Darabont

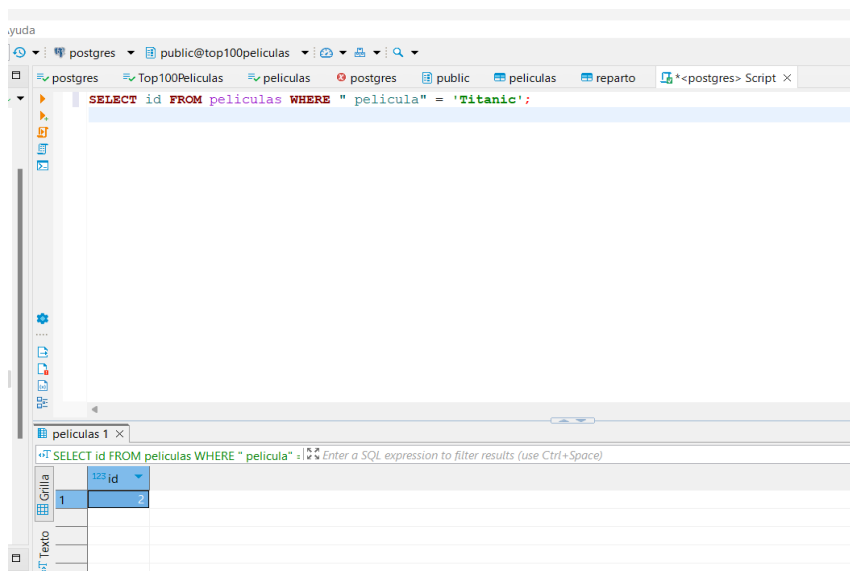
Evidencia de la carga de datos a la tabla reparto.



	id	actor
1	1	Tom Hanks
2	1	Robin Wright Penn
3	1	Gary Sinise
4	1	Mykelti Williamson
5	1	Sally Field
6	1	Rebecca Williams
7	1	Michael Conner Humphreys
8	1	Harold G. Herthum
9	1	Haley Joel Osment
10	1	George Kelly
11	1	Bob Penny
12	1	John Randall
13	1	Sam Anderson
14	1	Margo Moorer
15	1	Ione M. Telech
16	1	Christine Seabrook
17	2	Leonardo DiCaprio
18	2	Kate Winslet
19	2	Billy Zane
20	2	Kathy Bates
21	2	Frances Fisher
22	2	Bernard Hill
23	2	Jonathan Hyde
24	2	Danny Nucci
25	2	David Warner
26	2	Bill Paxton
27	2	Gloria Stuart
28	2	Victor Garber
29	2	Suzy Amis
30	3	Marlon Brando
31	3	Al Pacino

3. Obtener el ID de la película “Titanic”

```
SELECT id FROM peliculas WHERE " pelicula" = 'Titanic';
```

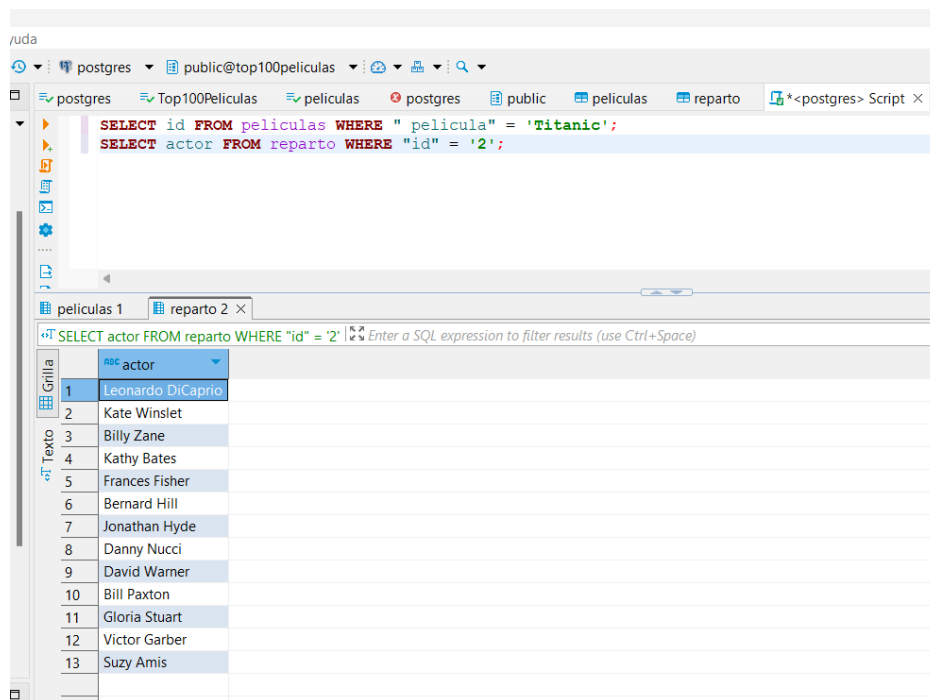


```
SELECT id FROM peliculas WHERE " pelicula" = 'Titanic';
```

	id
1	2

4. Listar a todos los actores que aparecen en la película "Titanic".

```
SELECT actor FROM reparto WHERE "id" = '2';
```

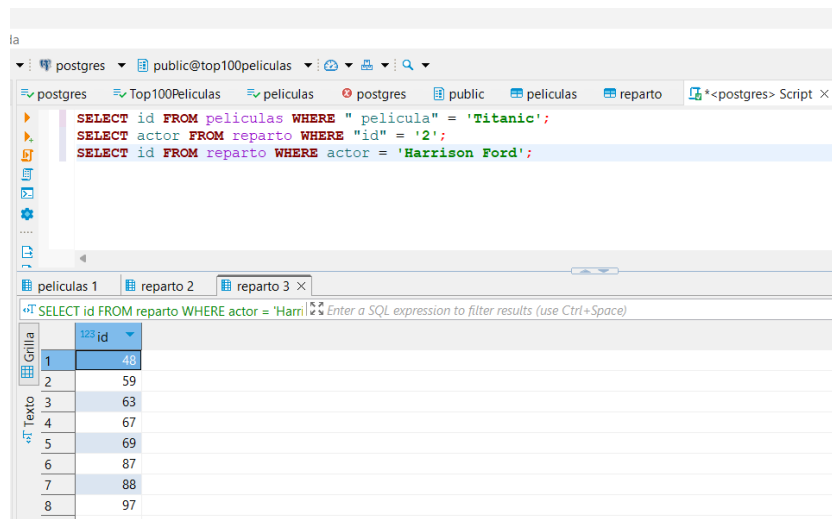


Grilla

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

5. Consultar en cuántas películas del top 100 participa Harrison Ford.

```
SELECT id FROM reparto WHERE actor = 'Harrison Ford';
```

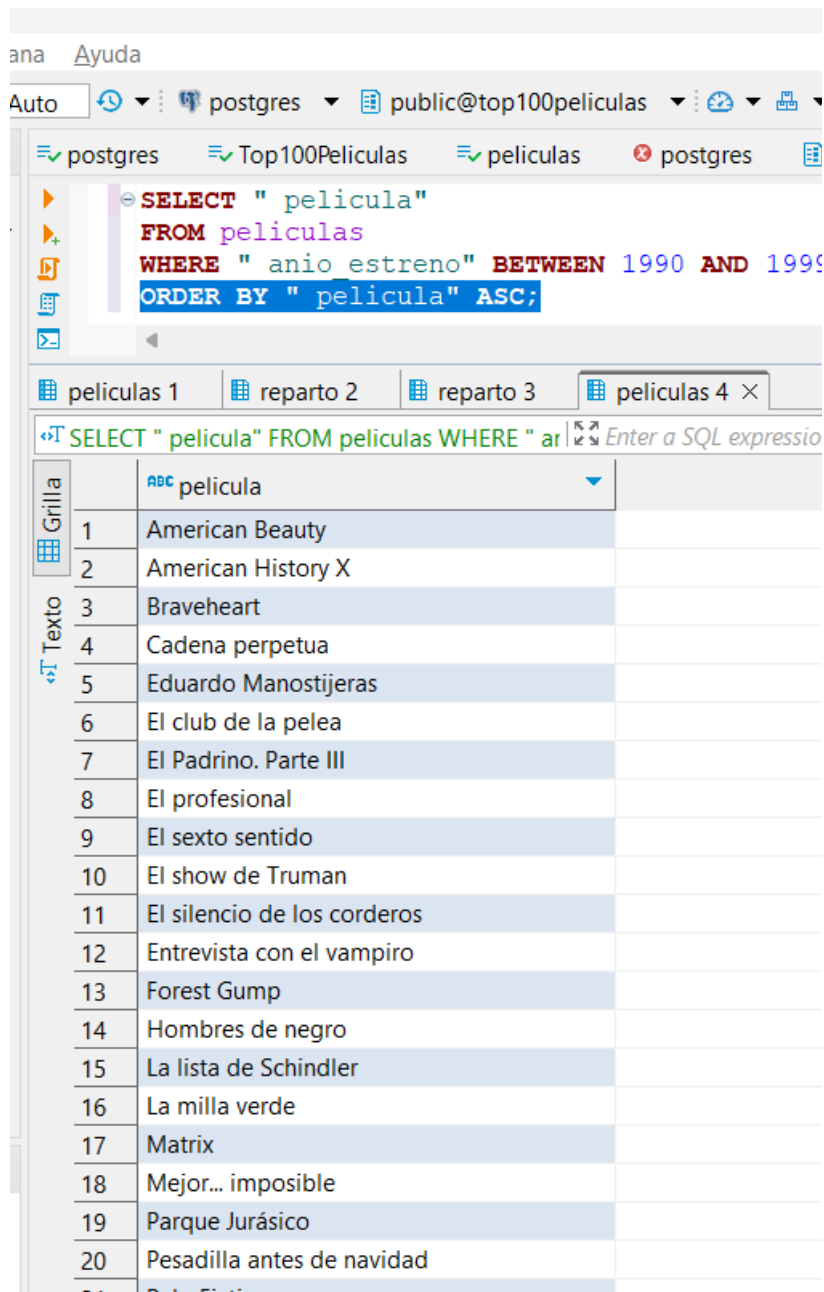


Grilla

	id
1	48
2	59
3	63
4	67
5	69
6	87
7	88
8	97

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

```
SELECT " pelicula" FROM películas WHERE " anio_estreno" BETWEEN 1990 AND 1999 ORDER BY " pelicula" ASC;
```



The screenshot shows a PostgreSQL query editor interface. The query is: `SELECT " pelicula" FROM películas WHERE " anio_estreno" BETWEEN 1990 AND 1999 ORDER BY " pelicula" ASC;`. The results are displayed in a table with the following data:

	abc pelicula
1	American Beauty
2	American History X
3	Braveheart
4	Cadena perpetua
5	Eduardo Manostijeras
6	El club de la pelea
7	El Padrino. Parte III
8	El profesional
9	El sexto sentido
10	El show de Truman
11	El silencio de los corderos
12	Entrevista con el vampiro
13	Forest Gump
14	Hombres de negro
15	La lista de Schindler
16	La milla verde
17	Matrix
18	Mejor... imposible
19	Parque Jurásico
20	Pesadilla antes de navidad
21	Rule Fiction

ana Ayuda

Auto postgres public@top100peliculas

postgres Top100Películas × películas postgres

```
SELECT " pelicula"
FROM películas
WHERE " año_estreno" BETWEEN 1990 AND 1999
ORDER BY " pelicula" ASC;
```

películas 1 reparto 2 reparto 3 películas 4 ×

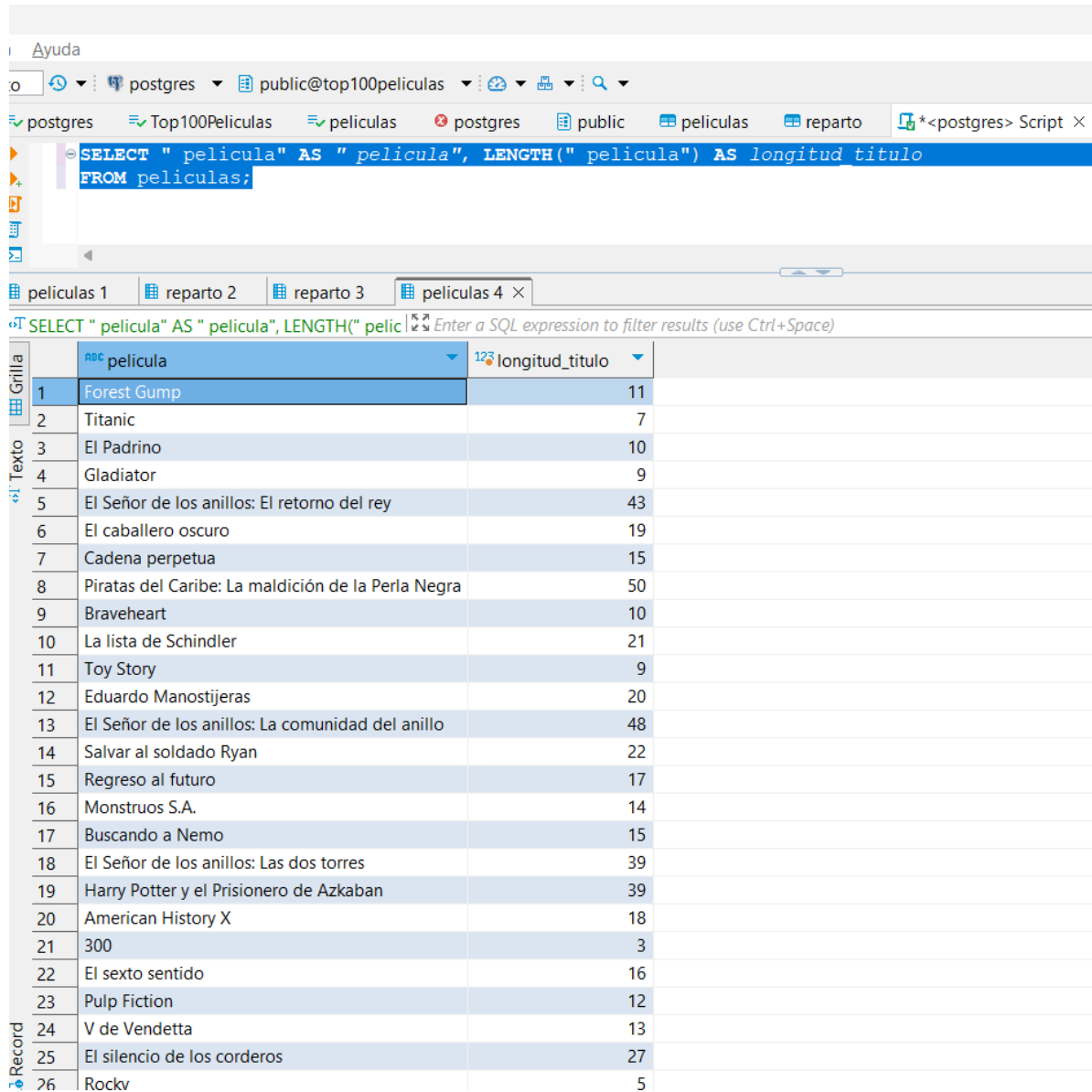
SELECT " pelicula" FROM películas WHERE " ar Enter a SQL expression

Grilla	abc película
6	El club de la pelea
7	El Padrino. Parte III
8	El profesional
9	El sexto sentido
10	El show de Truman
11	El silencio de los corderos
12	Entrevista con el vampiro
13	Forest Gump
14	Hombres de negro
15	La lista de Schindler
16	La milla verde
17	Matrix
18	Mejor... imposible
19	Parque Jurásico
20	Pesadilla antes de navidad
21	Pulp Fiction
22	Salvar al soldado Ryan
23	Seven
24	Star Wars. Episodio I: La amenaza fantasma
25	Terminator 2: el juicio final
26	Titanic
27	Toy Story
28	Toy Story 2
29	Trainspotting
30	Uno de los nuestros

Record

7. Hacer una consulta SQL que muestre los títulos con su longitud, la longitud debe ser nombrado para la consulta como “longitud_titulo”

```
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM peliculas;
```



The screenshot shows a PostgreSQL client interface. The top panel displays the SQL query: `SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo FROM peliculas;`. The bottom panel shows the results in a table grid. The table has two columns: " pelicula" and longitud_titulo. The results are as follows:

	" pelicula"	longitud_titulo
1	Forest Gump	11
2	Titanic	7
3	El Padrino	10
4	Gladiator	9
5	El Señor de los anillos: El retorno del rey	43
6	El caballero oscuro	19
7	Cadena perpetua	15
8	Piratas del Caribe: La maldición de la Perla Negra	50
9	Braveheart	10
10	La lista de Schindler	21
11	Toy Story	9
12	Eduardo Manostijeras	20
13	El Señor de los anillos: La comunidad del anillo	48
14	Salvar al soldado Ryan	22
15	Regreso al futuro	17
16	Monstruos S.A.	14
17	Buscando a Nemo	15
18	El Señor de los anillos: Las dos torres	39
19	Harry Potter y el Prisionero de Azkaban	39
20	American History X	18
21	300	3
22	El sexto sentido	16
23	Pulp Fiction	12
24	V de Vendetta	13
25	El silencio de los corderos	27
26	Rocky	5

Ayuda

postgres public@top100peliculas

postgres Top100Peliculas peliculas postgres public peliculas reparto *<postgres> Script X

```
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM peliculas;
```

peliculas 1 reparto 2 reparto 3 peliculas 4 X

SELECT " pelicula" AS " pelicula", LENGTH(" pelic

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

	asc pelicula	123 longitud_titulo
27	El club de la pelea	19
28	E.T	3
29	Parque Jurásico	15
30	Matrix	6
31	La milla verde	14
32	Ratatouille	11
33	Grease	6
34	El Padrino. Parte II	20
35	Wall-E	6
36	Iron Man	8
37	El exorcista	12
38	Piratas del Caribe: El cofre del hombre muerto	46
39	Seven	5
40	Terminator 2: el juicio final	29
41	El resplandor	13
42	Hombres de negro	16
43	Spider-Man	10
44	Regreso al futuro II	20
45	El show de Truman	17
46	Toy Story 2	11
47	Pesadilla antes de navidad	26
48	Star Wars. Episodio IV: Una nueva esperanza	43
49	Saw	3
50	Terminator	10
51	Kill Bill Vol. 1	16
52	Lo que el viento se llevó	25

<div> <div>postgres</div> <div>Top100Peliculas</div> <div>✓ peliculas</div> <div>✗ postgres</div> <div>public</div> <div>peliculas</div> <div>reparto</div> <div>*<postgres> Script</div> </div>			
<div> <div> <div>SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo</div> <div>FROM peliculas;</div> </div> </div>			
<div> <div> <div>peliculas 1</div> <div>reparto 2</div> <div>reparto 3</div> <div>peliculas 4</div> </div> </div>			
<div> <div> <div>SELECT " pelicula" AS " pelicula", LENGTH(" pelic</div> <div>Enter a SQL expression to filter results (use Ctrl+Space)</div> </div> </div>			
<div> <div>Grilla</div> <div>Texto</div> <div>Record</div> </div>		asc pelicula	desc longitud_titulo
	53	El Laberinto del Fauno	22
	54	Los increíbles	14
	55	El viaje de Chihiro	19
	56	El precio del poder	19
	57	King Kong	9
	58	Ben-Hur	7
	59	Indiana Jones y la última cruzada	33
	60	Infiltrados	11
	61	Entrevista con el vampiro	25
	62	Batman Begins	13
	63	En busca del arca perdida	25
	64	Star Wars. Episodio III: La venganza de los Sith	48
	65	Alien	5
	66	El bueno el feo y el malo	25
	67	Star Wars. Episodio V: El imperio contraataca	45
	68	El Padrino. Parte III	21
	69	Star Wars. Episodio VI: El retorno del Jedi	43
	70	Escuela de Rock	15
	71	El planeta de los simios	24
	72	Rambo	5
	73	Full Metal Jacket	17
	74	Tiburón	7
	75	El profesional	14
	76	American Beauty	15
	77	Casablanca	10
	78	Amélie	6

postgres

Top100Peliculas

peliculas

postgres

public

peliculas

reparto

*<postgres> Script

```
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM peliculas;
```

peliculas 1

reparto 2

reparto 3

peliculas 4

SELECT " pelicula" AS " pelicula", LENGTH(" pelic

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

Grilla

asc

pelicula

123

longitud_titulo

75

El profesional

14

76

American Beauty

15

77

Casablanca

10

78

Amélie

6

79

Trainspotting

13

80

axi driver

10

81

Alguien voló sobre el nido del cuco

35

82

Kill Bill Vol. 2

16

83

Spider-Man 2

12

84

X-Men 2

7

85

Transformers

12

86

Star Wars. Episodio I: La amenaza fantasma

42

87

Blade Runner

12

88

Apocalypse Now

14

89

Mejor... imposible

18

90

La vida de Brian

16

91

El libro de la selva

20

92

Los cazafantasmas

17

93

Sweeney Todd: El barbero diabólico de la calle Fle

52

94

Ocean's Eleven

14

95

Blancanieves y los siete enanitos

33

96

Predator

8

97

Indiana Jones y el templo maldito

33

98

Uno de los nuestros

19

99

Mouling Rouge

13

100

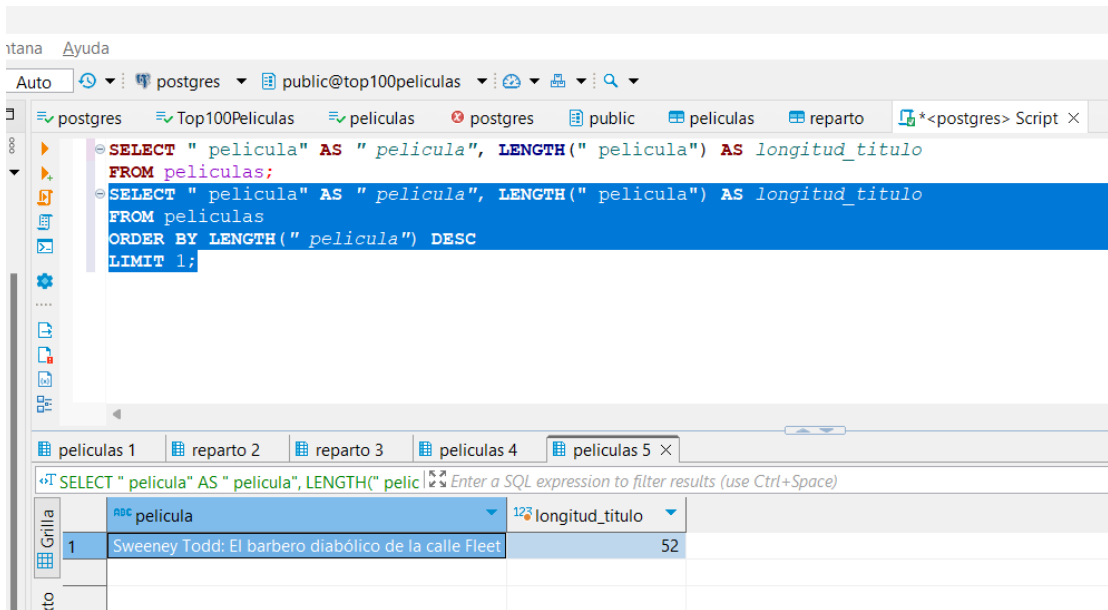
Psicosis

8

Record

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

```
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM películas ORDER BY LENGTH(" pelicula") DESC LIMIT 1;
```



The screenshot shows a PostgreSQL client interface with a SQL query editor and a results grid. The query is:

```
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM películas;
SELECT " pelicula" AS " pelicula", LENGTH(" pelicula") AS longitud_titulo
FROM películas
ORDER BY LENGTH(" pelicula") DESC
LIMIT 1;
```

The results grid shows the following data:

	película	longitud_titulo
1	Sweeney Todd: El barbero diabólico de la calle Fleet	52

¡¡¡Gracias!!!