

# Assignment 1: DVDRental Queries

## DVD-Rental Assignment 1:

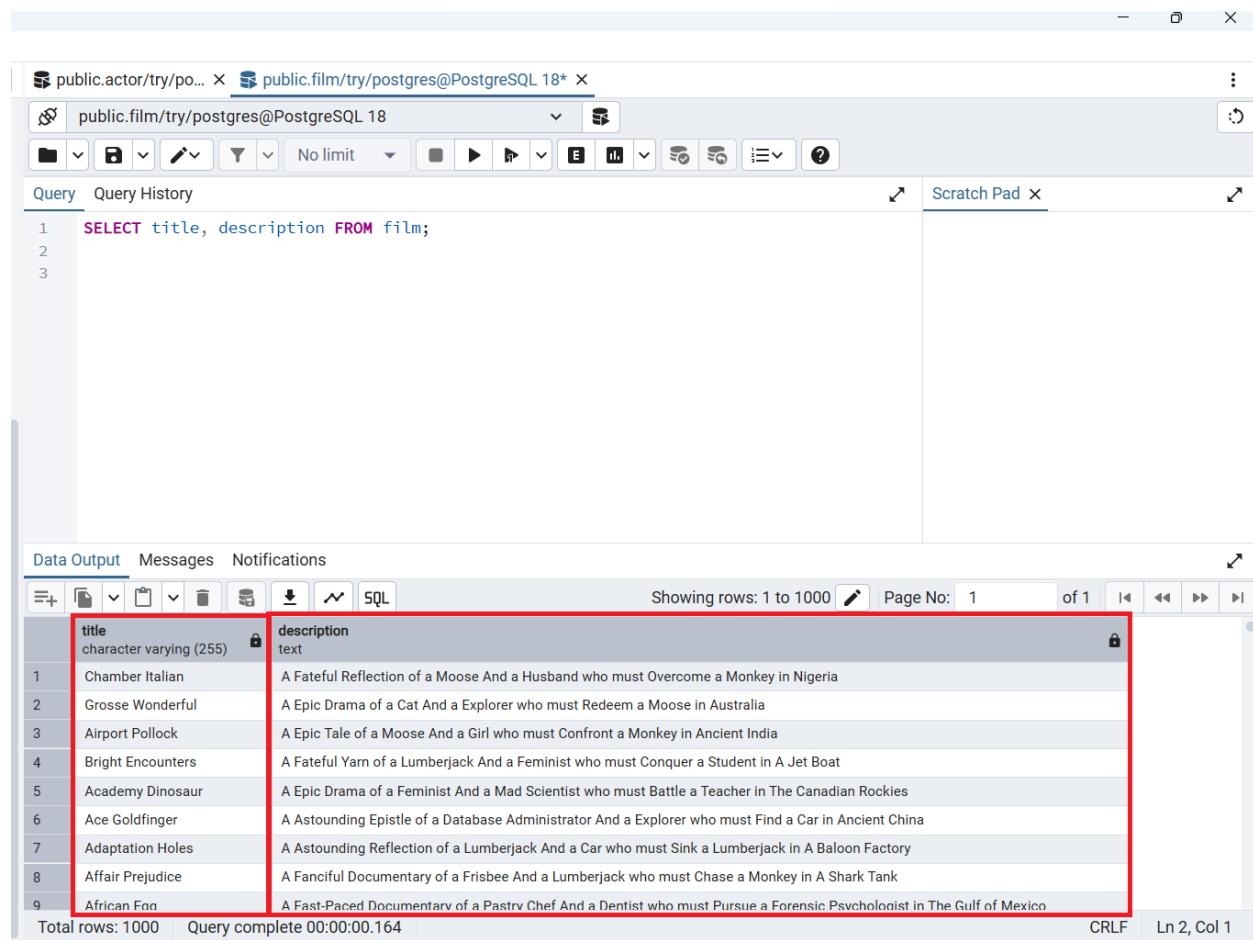
This assignment includes basic PostgreSQL queries on the DVDRental database.

The queries demonstrate how to retrieve information from the film and customer tables, applying different conditions and filters.

Screenshots of the query results are included below each query.

Query 1:

**Description:** Retrieve the title and description columns from the film table.



The screenshot shows the pgAdmin 4 interface with two tabs open: 'public.actor/try/po...' and 'public.film/try/postgres@PostgreSQL 18\*'. The SQL tab contains the following query:

```
1  SELECT title, description FROM film;
```

The Data Output tab displays the results of the query:

	title	description
1	Chamber Italian	A Fateful Reflection of a Moose And a Husband who must Overcome a Monkey in Nigeria
2	Grosse Wonderful	A Epic Drama of a Cat And a Explorer who must Redeem a Moose in Australia
3	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India
4	Bright Encounters	A Fateful Yarn of a Lumberjack And a Feminist who must Conquer a Student in A Jet Boat
5	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies
6	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China
7	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory
8	Affair Prejudice	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank
9	African Foo	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico

Total rows: 1000 Query complete 00:00:00.164 CRLF Ln 2, Col 1

## Query 2:

**Description:** Retrieve all columns from the film table for films with length > 60 and < 75.

The screenshot shows the pgAdmin 4 interface with two tabs open: 'public.actor/try/postgres@PostgreSQL 18\*' and 'public.film/try/postgres@PostgreSQL 18\*'. The 'public.film' tab is active. The 'Query' tab contains the following SQL code:

```
1 SELECT * FROM film
2 WHERE length > 60 AND length < 75;
3
4
```

The 'Data Output' tab displays the results of the query. A red box highlights the 'length' column for rows 1 through 8. The results are as follows:

	length smallint	replacement_cost numeric (5,2)	rating mpaa_rating	last_update timestamp without time zone	special_features text[]	fulltext tsvector
1	73	12.99	PG-13	2013-05-26 14:50:58.951	{Trailers}	'boat':20 'bright':1 'conquer':14 'er
2	62	28.99	PG-13	2013-05-26 14:50:58.951	{Trailers,'Deleted Scenes'}	'airplan':1 'boat':20 'butler':11,16 '
3	63	24.99	NC-17	2013-05-26 14:50:58.951	{Trailers,'Deleted Scenes'}	'action':5 'action-pack':4 'aladdin':
4	74	15.99	G	2013-05-26 14:50:58.951	{Trailers}	'angel':1 'astronaut':11 'battl':14 't
5	62	29.99	NC-17	2013-05-26 14:50:58.951	{Commentaries,'Deleted Scenes'}	'arabia':1 'cow':12 'defeat':15 'dog
6	68	25.99	NC-17	2013-05-26 14:50:58.951	{Trailers,Commentaries,'Deleted Scenes','Behind the Scen...	'ark':1 'beauti':4 'desert':20 'explor
7	65	27.99	G	2013-05-26 14:50:58.951	{"Behind the Scenes"}	'awe':5 'awe-inspir':4 'barbarella':1
8	73	21.99	PG	2013-05-26 14:50:58.951	{Trailers,'Deleted Scenes','Behind the Scenes'}	'abandon':21 'astound':4 'bedazzl'

Total rows: 104 | Query complete 00:00:00.236 | CRLF | Ln 2, Col 35

### Query 3:

**Description:** Retrieve all columns from the film table where rental\_rate = 0.99 **and** replacement\_cost = 12.99 or 28.99.

The screenshot shows the pgAdmin 4 interface with two tabs open: 'public.actor/try/po...' and 'public.film/try/postgres@PostgreSQL 18\*'. The current tab is 'public.film/try/postgres@PostgreSQL 18\*'. The query window contains the following SQL code:

```
1 SELECT * FROM film
2 WHERE rental_rate = 0.99
3 AND (replacement_cost = 12.99 OR replacement_cost = 28.99);
4
5
```

The results window displays the output of the query, showing 37 rows. The columns are: rental\_rate, length, replacement\_cost, rating, mpaa\_rating, last\_update, special\_features, and fulltext\_tsvector. The first two columns (rental\_rate and replacement\_cost) are highlighted with red boxes. The data is as follows:

	rental_rate	length	replacement_cost	rating	mpaa_rating	last_update	special_features	fulltext_tsvector
1	0.99	179	12.99	NC-17	2013-05-26 14:50:58.951	{"Deleted Scenes","Behind the Scenes"}	'administr':9.18	
2	0.99	127	12.99	PG-13	2013-05-26 14:50:58.951	{Trailers,Commentaries}	'abandon':20 'ar	
3	0.99	65	12.99	PG-13	2013-05-26 14:50:58.951	{Trailers,"Deleted Scenes"}	'boat':21 'club':1	
4	0.99	177	28.99	R	2013-05-26 14:50:58.951	{Trailers,"Deleted Scenes","Behind the Scenes"}	'challeng':14 'dc	
5	0.99	104	12.99	PG-13	2013-05-26 14:50:58.951	{Trailers,"Deleted Scenes"}	'australia':18 'de	
6	0.99	178	28.99	PG-13	2013-05-26 14:50:58.951	{"Deleted Scenes"}	'boat':8 'fight':1	
7	0.99	54	12.99	R	2013-05-26 14:50:58.951	{Trailers}	'administr':14 't	
8	0.99	108	28.99	R	2013-05-26 14:50:58.951	{Commentaries,"Deleted Scenes","Behind the Scenes"}	'beauti':4 'boon'	

Total rows: 37 | Query complete 00:00:00.129 | CRLF | Ln 4, Col 1

Query 4:

**Description:** Retrieve the last\_name of the customer whose first\_name = 'Mary'.

**Answer:** Smith

The screenshot shows the pgAdmin 4 interface with two tabs open: 'public.actor/try/po...' and 'public.film/try/postgres@PostgreSQL 18\*'. The active tab is 'public.film/try/postgres@PostgreSQL 18'. The 'Query' tab contains the following SQL code:

```
1 SELECT last_name FROM customer
2 WHERE first_name = 'Mary';
```

The 'Data Output' tab displays the results of the query:

last_name
Smith

The entire table row is highlighted with a red box. The status bar at the bottom indicates 'Total rows: 1' and 'Query complete 00:00:00.126'.

### Query 5:

**Description:** Retrieve all columns from the film table where length  $\leq 50$  and rental\_rate is not 2.99 or 4.99.

The screenshot shows the pgAdmin 4 interface with two tabs open: 'public.actor/try/po...' and 'public.film/try/postgres@PostgreSQL 18\*'. The current tab is 'public.film/try/postgres@PostgreSQL 18\*'. The query window contains the following SQL code:

```
1 SELECT * FROM film
2 WHERE length <= 50 AND (NOT rental_rate = 2.99 AND NOT rental_rate = 4.99);
```

The results window displays the output of the query, showing 13 rows of film data. The first two columns, 'rental\_rate' and 'length', are highlighted with a red border. The data is as follows:

	rental_rate	length	replacement_cost	rating	last_update	special_features	fulltext
1	0.99	47	19.99	G	2013-05-26 14:50:58.951	{Trailers,Commentaries,"Deleted Scenes"}	'chef':9 'desert':22 'dow'
2	0.99	47	12.99	PG-13	2013-05-26 14:50:58.951	{"Behind the Scenes"}	'action':5 'action-pack':
3	0.99	49	23.99	G	2013-05-26 14:50:58.951	{Trailers,Commentaries,"Behind the Scenes"}	'australia':18 'boy':8,16
4	0.99	46	25.99	PG-13	2013-05-26 14:50:58.951	{Trailers,Commentaries}	'amaz':4 'car':8,16 'dran'
5	0.99	48	20.99	PG	2013-05-26 14:50:58.951	{"Deleted Scenes"}	'australia':18 'challeng':
6	0.99	50	13.99	PG	2013-05-26 14:50:58.951	{Trailers,"Deleted Scenes"}	'ancient':20 'cat':12 'chi'
7	0.99	50	24.99	PG-13	2013-05-26 14:50:58.951	{Commentaries,"Deleted Scenes"}	'baloon':22 'car':19 'def'
8	0.99	48	19.99	PG-13	2013-05-26 14:50:58.951	{Trailers,Commentaries}	'butler':8 'canadian':19