

1. Write a query to get all data of actor.

ANS. `SELECT * FROM actor;` [Displayed 200 rows]

2. Write a query to get email and last name of customer.

ANS. `SELECT last_name, email FROM customer;` [Displayed 599 rows]

3. Write a query to get title, description and release year of film.

ANS.

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays the database structure, with 'Tables (15)' expanded under the 'public' schema. The 'Query Editor' pane shows the following SQL query:

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

The 'Data Output' pane displays the results of the query in a table format:

	title	description	release_year
1	Chamber Italian	A Fateful Reflection of a Moose And a Husband who must Overcome a Monkey in Nigeria	2006
2	Grosse Wonderful	A Epic Drama of a Cat And a Explorer who must Redeem a Moose in Australia	2006
3	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India	2006
4	Bright Encounters	A Fateful Yarn of a Lumberjack And a Feminist who must Conquer a Student in A Jet Boat	2006
5	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	2006
6	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China	2006
7	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory	2006
8	Affair Prejudice	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank	2006
9	African Egg	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico	2006

4. Query city and country id in the city table.

ANS.SELECT city\_id, country\_id FROM city;

The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to show the 'public' schema, with 'Tables (15)' selected. The 'Query Editor' pane contains the following SQL query:

```
1 SELECT city_id, country_id FROM city;
```

The 'Data Output' pane displays the results of the query in a table format:

	city_id [PK] integer	country_id smallint
1	1	87
2	2	82
3	3	101
4	4	60
5	5	97
6	6	31
7	7	107
8	8	44
9	9	44
10	10	50

5. Write a query to get amount, payment date and customer id from customer table.

ANS.SELECT amount, payment\_date, customer\_id FROM payment;

The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to show the 'public' schema, with 'Tables (15)' selected. The 'Query Editor' pane contains the following SQL query:

```
1 SELECT amount, payment_date, customer_id FROM payment;
```

The 'Data Output' pane displays the results of the query in a table format:

	amount numeric (5,2)	payment_date timestamp without time zone	customer_id smallint
1	7.99	2007-02-15 22:25:46.996577	341
2	1.99	2007-02-16 17:23:14.996577	341
3	7.99	2007-02-16 22:41:45.996577	341
4	2.99	2007-02-19 19:39:56.996577	341
5	7.99	2007-02-20 17:31:48.996577	341
6	5.99	2007-02-21 12:33:49.996577	341
7	5.99	2007-02-17 23:58:17.996577	342
8	5.99	2007-02-20 02:11:44.996577	342
9	2.99	2007-02-20 13:57:39.996577	342
10	4.99	2007-02-16 00:10:50.996577	343

6. Write a query to get all data of language.

ANS.SELECT \* FROM language;

1	"English	"	"2006-02-15 10:02:19"
2	"Italian	"	"2006-02-15 10:02:19"
3	"Japanese	"	"2006-02-15 10:02:19"
4	"Mandarin	"	"2006-02-15 10:02:19"
5	"French	"	"2006-02-15 10:02:19"
6	"German	"	"2006-02-15 10:02:19"

7. Query all columns for a payment in payment table with customer id 10.

ANS.SELECT payment FROM payment WHERE customer\_id=10;

PAYMENT

"(18532,10,1,1801,4.99,""2007-02-16 18:50:19.996577"")"

"(18533,10,1,1995,4.99,""2007-02-17 09:39:40.996577"")"

"(18534,10,2,2222,3.99,""2007-02-18 01:54:49.996577"")"

"(18535,10,1,2814,0.99,""2007-02-19 18:30:25.996577"")"

"(18536,10,1,2865,0.99,""2007-02-19 22:29:21.996577"")"

"(22773,10,2,10671,8.99,""2007-03-01 15:38:25.996577"")"

"(22774,10,2,11289,2.99,""2007-03-02 13:23:26.996577"")"

"(22775,10,1,11405,0.99,""2007-03-02 17:42:05.996577"")"

"(22776,10,2,12031,2.99,""2007-03-17 18:40:01.996577"")"

"(22777,10,2,12400,2.99,""2007-03-18 07:47:38.996577"")"

"(22778,10,2,13316,4.99,""2007-03-19 17:51:56.996577"")"

"(22779,10,2,13917,2.99,""2007-03-20 15:11:54.996577"")"

"(22780,10,1,15370,5.99,""2007-03-22 20:27:55.996577"")"

"(29094,10,2,3790,3.99,""2007-04-06 12:42:11.996577"")"

"(29095,10,2,4042,4.99,""2007-04-07 01:35:06.996577"")"

"(29096,10,1,4255,1.99,""2007-04-07 12:42:39.996577"")"

"(29097,10,1,5038,7.99,""2007-04-09 01:41:18.996577"")"

"(29098,10,2,5068,2.99,""2007-04-09 03:21:44.996577"")"

"(29099,10,1,5444,0.99,""2007-04-09 20:27:23.996577"")"

"(29100,10,1,5905,2.99,""2007-04-10 19:09:35.996577"")"

"(29101,10,1,7738,2.99,""2007-04-28 03:50:08.996577"")"

"(29102,10,2,8001,6.99,""2007-04-28 13:39:21.996577"")"

"(29103,10,2,8188,4.99,""2007-04-28 21:02:38.996577"")"

"(29104,10,1,9935,4.99,""2007-04-30 13:55:33.996577"")"

8. Query last name and first name of customers in customer table whose first names are "Mary"

```
ANS.SELECT DISTINCT last_name, first_name FROM customer WHERE first_name='Mary';
```

last_name	first_name
Smith	Mary

9. Query last name and first name of customers in customer table whose first names are "Mary" and last names are "Smith".

```
ANS.SELECT DISTINCT last_name, first_name FROM customer WHERE first_name='Mary'
AND last_name='Smith';
```

last_name	first_name
Smith	Mary

10. Query last name and first name of customers in customer table whose first names are "Susan" or last names are "Jones".

```
ANS.Displayed 0 rows!
```

11. Query email of customers in customer table whose first name is "Mar", "Mary" or "Mari".

```
ANS.SELECT DISTINCT email FROM customer WHERE first_name = 'Mar' OR
first_name='Mary' OR First_name='Mari';
```

Email
mary.smith@sakilacustomer.org

(1 row)

12. Query last name and first name of customers in customer table whose first names start with "Ma".

ANS. `SELECT last_name, first_name FROM customer WHERE first_name like 'Ma%';`

The screenshot shows the pgAdmin 4 web interface. On the left, the 'Browser' pane displays a tree view of the database structure, with 'Tables (15)' under the 'public' schema selected. The main pane is divided into two sections: the 'Query Editor' and the 'Data Output' pane. The 'Query Editor' contains the following SQL query:

```
1 SELECT last_name, first_name FROM customer WHERE first_name
2 LIKE 'Ma%';
3
4
5
6
```

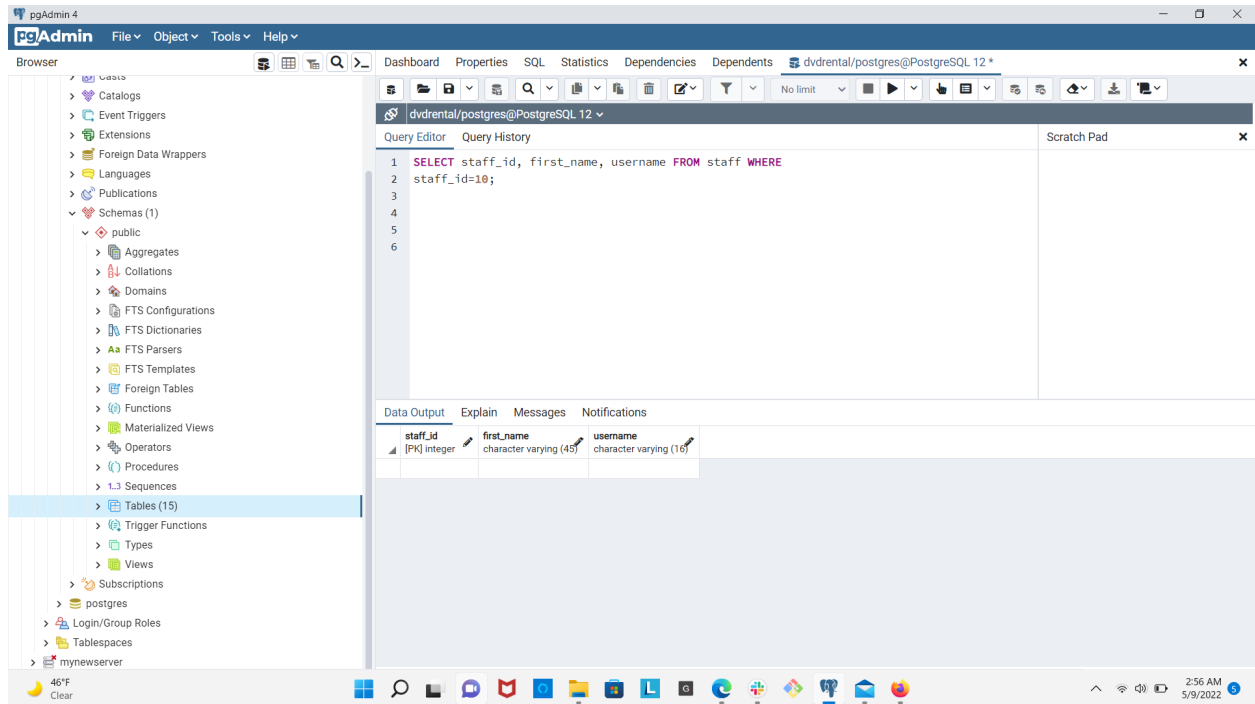
The 'Data Output' pane displays the results of the query in a table format. The table has two columns: 'last\_name' and 'first\_name'. The results are as follows:

last_name	first_name
Murrell	Manuel
Yee	Marvin
Cheatham	Mario
Hidalgo	Marcus
Crawley	Maurice
Outlaw	Marc
Bolin	Mathew
Pitt	Max
Thorn	Marshall
Ocampo	Marion

The bottom of the interface shows a Windows taskbar with various application icons and a system clock indicating 2:54 AM on 5/9/2022.

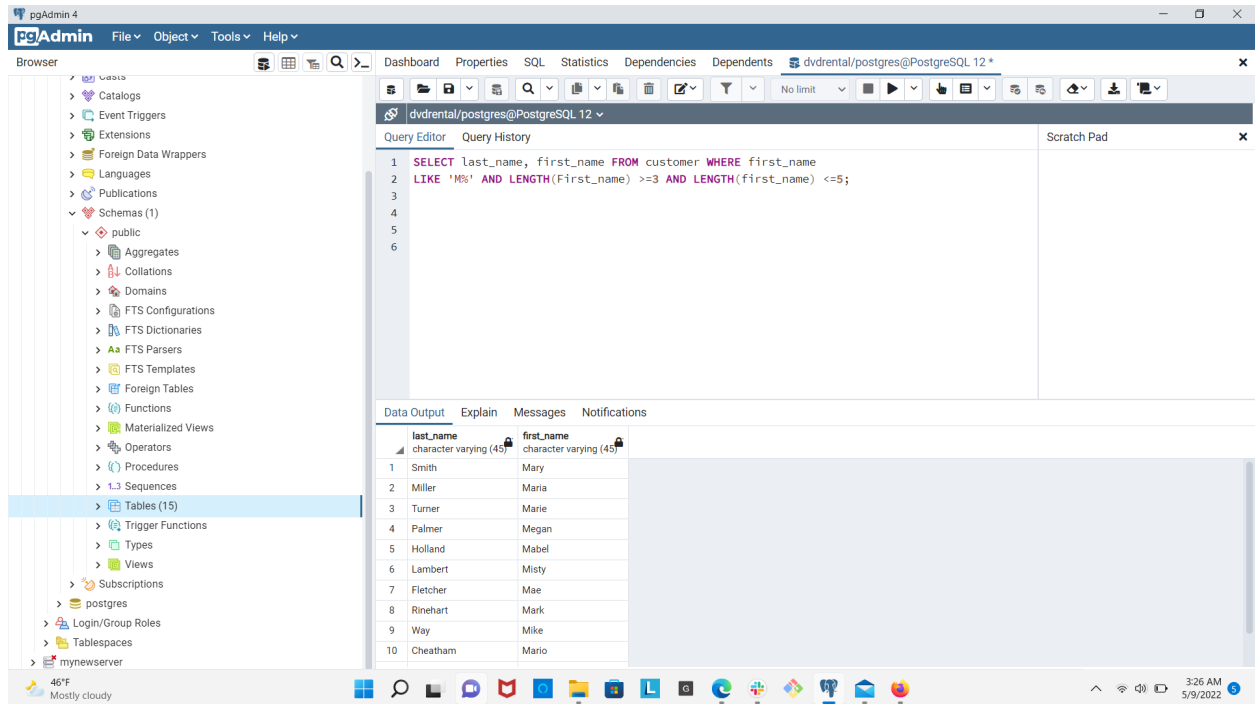
13. Write a query to get staff id, first name and username of staff in staff table whose staff id is 10.

ANS. `SELECT staff_id, first_name, username FROM staff WHERE staff_id=10;`



14. Query last name and first name of customers in customer table whose first name start with the letter "M" and contains 3 to 5 characters.

ANS. SELECT last\_name, first\_name FROM customer WHERE first\_name LIKE 'M%' AND LENGTH(First\_name) >=3 AND LENGTH(first\_name) <=5;



The screenshot shows the pgAdmin 4 web interface. On the left is the 'Browser' pane with a tree view of the database structure. The 'Tables (15)' folder is selected. The main area is divided into three panes: 'Query Editor', 'Query History', and 'Scratch Pad'. The 'Query Editor' contains the following SQL query:

```
1 SELECT last_name, first_name FROM customer WHERE first_name
2 LIKE 'M%' AND LENGTH(First_name) >=3 AND LENGTH(first_name) <=5;
3
4
5
6
```

Below the query editor is the 'Data Output' pane, which displays the results of the query in a table format:

	last_name	first_name
1	Smith	Mary
2	Miller	Maria
3	Turner	Marie
4	Palmer	Megan
5	Holland	Mabel
6	Lambert	Misty
7	Fletcher	Mae
8	Rinehart	Mark
9	Way	Mike
10	Cheatham	Mario

The bottom of the screen shows a Windows taskbar with various application icons and a system tray indicating the time as 3:26 AM on 5/9/2022.

15. Query last name and first name of customers in customer table whose first names start with "Bra" and last names are not "Motley".

ANS. SELECT last\_name, first\_name FROM customer WHERE first\_name  
LIKE '%Bra%' AND last\_name NOT LIKE 'Motley';

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays the database structure, including 'Servers (2)', 'PostgreSQL 12', 'Databases (3)', and 'dvdrental'. The 'public' schema is expanded, showing various database objects. The 'Query Editor' pane on the right contains the following SQL query:

```
1 SELECT last_name, first_name FROM customer WHERE first_name
2 LIKE '%Bra%' AND last_name NOT LIKE 'Motley';
```

Below the query editor, the 'Data Output' pane displays the results of the query in a table format:

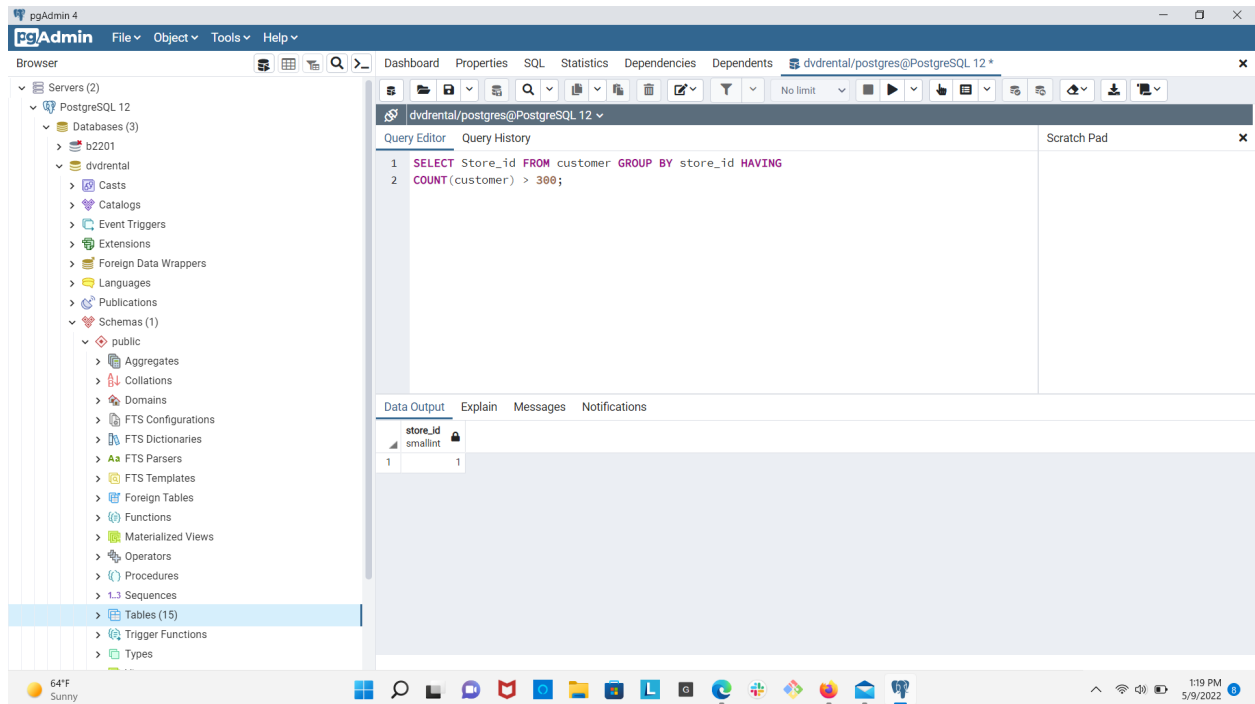
	last_name	first_name
1	Graves	Brandy
2	Huey	Brandon
3	Mccurdy	Brad

The bottom of the screen shows the Windows taskbar with the date and time as 1:08 PM on 5/9/2022.



16. Query store id of stores that have more than 300 customers in customer table.

ANS. SELECT Store\_id FROM customer GROUP BY store\_id HAVING  
COUNT(customer) > 300;



17. Write a query to select all details of the only customers who have been spending more than 200 in customer table.

ANS.SELECT customer\_id FROM payment GROUP BY customer\_id Having SUM(amount)  
>200

The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to show the 'public' schema, with 'Tables (15)' selected. The main pane displays a SQL query in the 'Query Editor':

```
1 SELECT customer_id FROM payment GROUP BY customer_id Having SUM(amount) >200
2
```

Below the query editor, the 'Data Output' tab shows the results of the query:

customer_id
526
148

The bottom status bar indicates the system is at 64°F and Sunny.

18. Query all columns in film table where the film\_id is less than 4.

ANS.SELECT \* FROM film WHERE film\_id < 4;

The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to show the 'public' schema, with 'Tables (15)' selected. The main pane displays a SQL query in the 'Query Editor':

```
1 SELECT * FROM film WHERE film_id < 4;
2
```

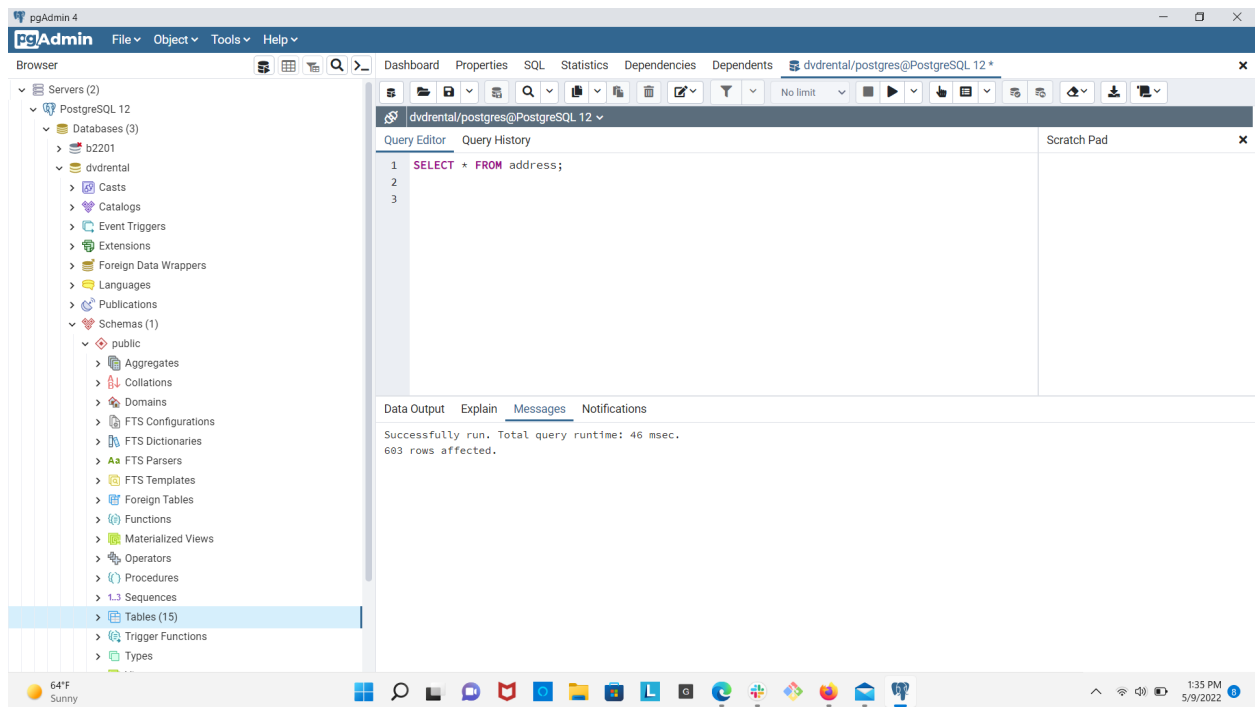
Below the query editor, the 'Data Output' tab shows the results of the query:

film_id	title	description	release_year	language_id	rental_id
1	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	2006	1	1
2	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China	2006	1	1
3	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory	2006	1	1

The bottom status bar indicates the system is at 64°F and Sunny.

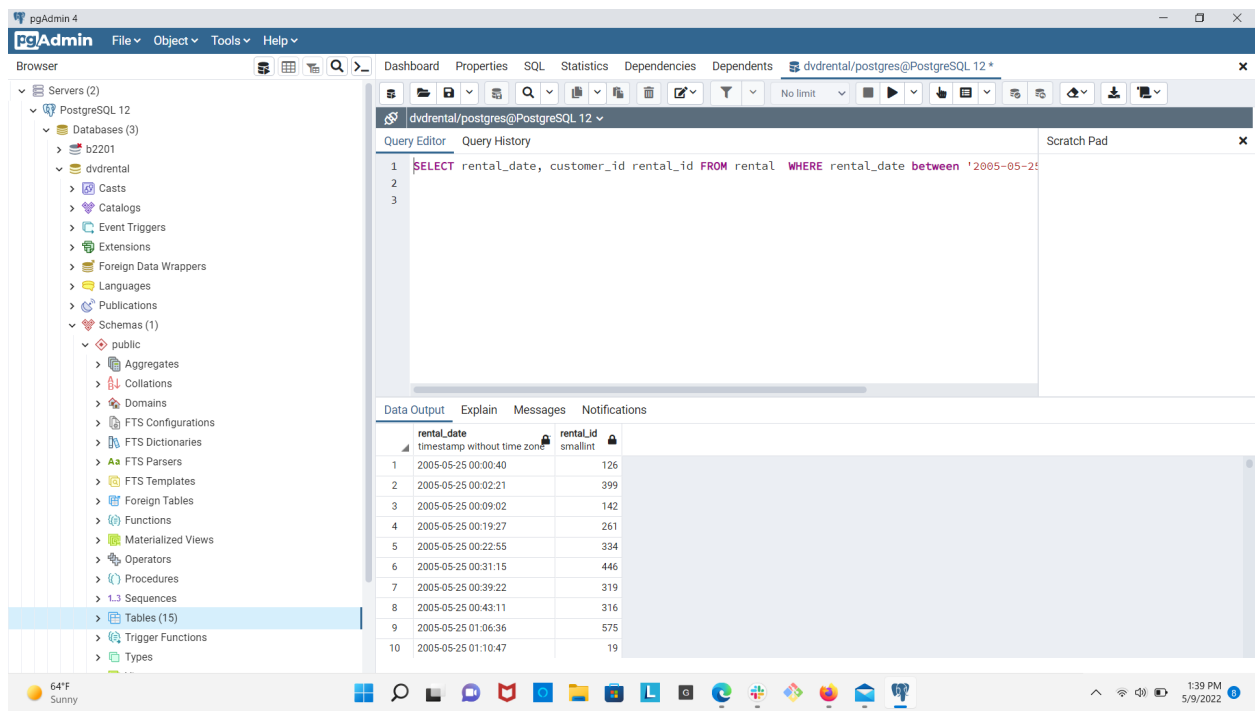
19. Write a query to get all data from address table.

ANS. `SELECT * FROM address;`



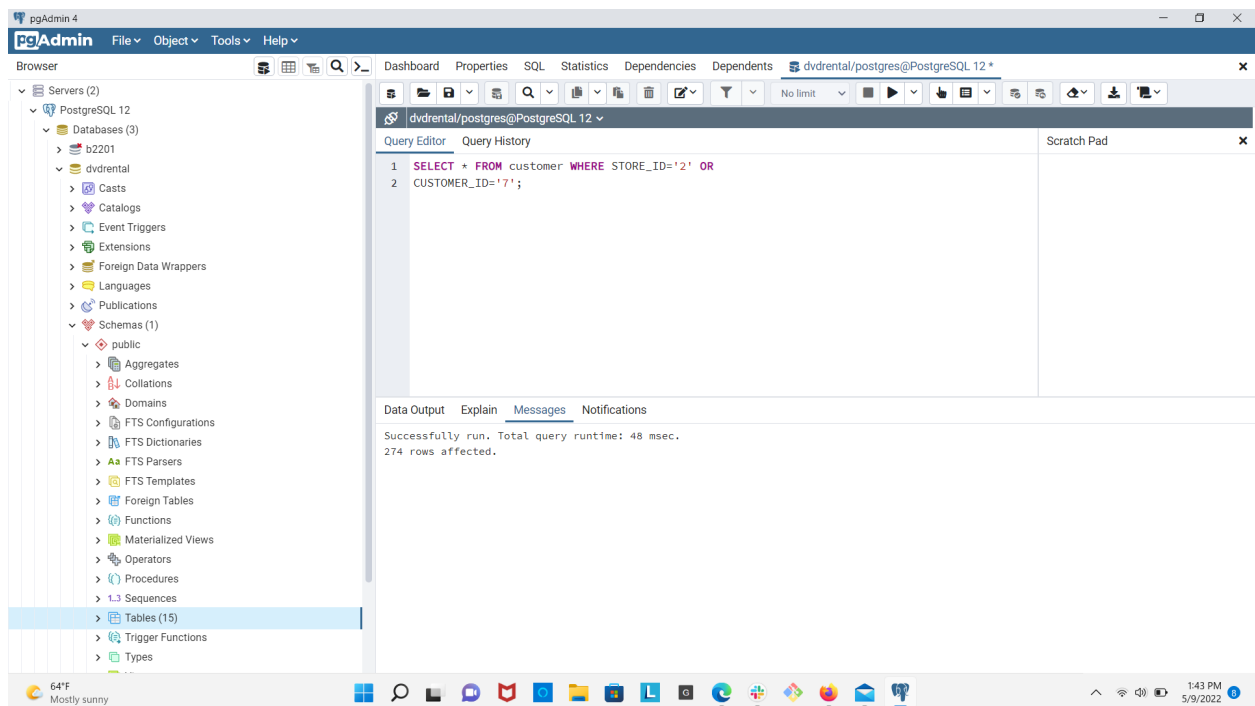
20. Query rental date, customer id and rental id in rental table when rental date is 2005-05-25.

ANS. `SELECT rental_date, customer_id rental_id FROM rental WHERE rental_date between '2005-05-25 00:00:00' and '2005-05-25 24:00:00'`



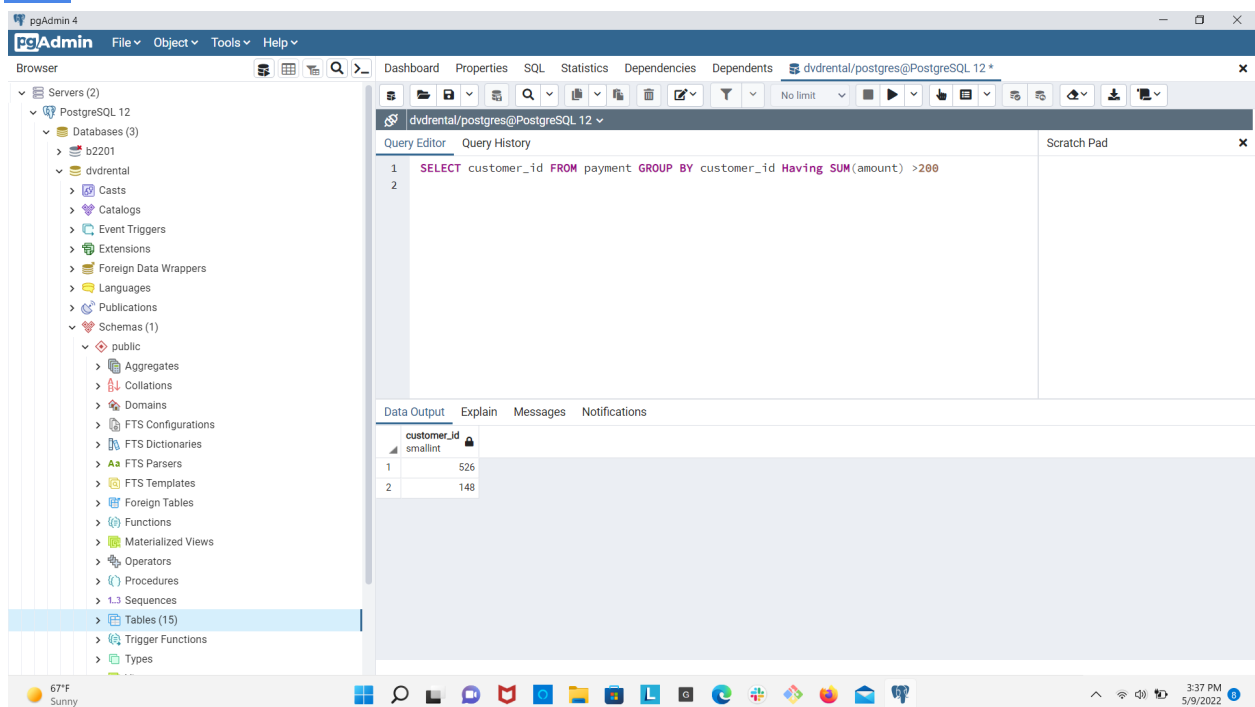
21. Query all columns for customers in customer table with store id 2 or customer id 7.

ANS. `SELECT * FROM customer WHERE STORE_ID='2' OR CUSTOMER_ID='7';`



22. Query all columns for customers in customer table who have spent amount more than \$200.

ANS. `SELECT customer_id FROM payment GROUP BY customer_id Having SUM(amount) >200`



23. Query amount and payment\_date from payment where the amount paid was less than \$2.

ANS. SELECT amount, payment\_date from payment WHERE amount < 2;

The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to 'PostgreSQL 12' > 'Databases' > 'b2201' > 'dvdrental' > 'Tables' > 'payment'. The 'Query Editor' tab is active, displaying the SQL query: `SELECT amount, payment_date from payment WHERE amount < 2;`. The 'Messages' tab shows the execution results: 'Successfully run. Total query runtime: 36 msec. 3325 rows affected.' The status bar at the bottom indicates '67°F Sunny' and the time '3:39 PM 5/9/2022'.

24. Write a query to get a list of actors with the first name Chris, Cameron, or Cuba.  
ANS. SELECT DISTINCT \* FROM actor WHERE first\_name = 'Chris' OR first\_name = 'Cameron' OR first\_name = 'Cuba';

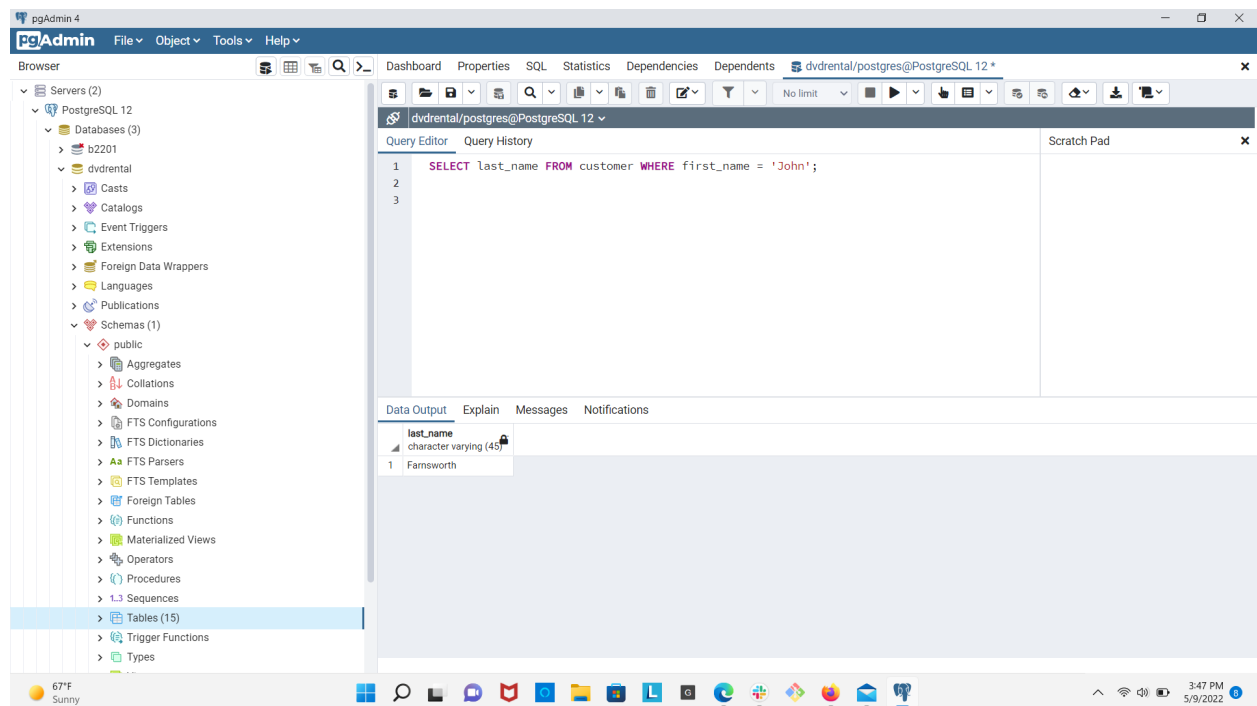
The screenshot shows the pgAdmin 4 interface. On the left, the 'Servers' tree is expanded to 'PostgreSQL 12' > 'Databases' > 'b2201' > 'dvdrental' > 'Tables' > 'actor'. The 'Query Editor' tab is active, displaying the SQL query: `SELECT DISTINCT * FROM actor WHERE first_name = 'Chris' OR first_name = 'Cameron' OR first_name = 'Cuba';`. The 'Messages' tab shows the execution results: 'Successfully run. Total query runtime: 36 msec. 8 rows affected.' The 'Data Output' tab displays the following table:

actor_id	first_name	last_name	last_update
24	Cameron	Streep	2013-05-26 14:47:57.62
118	Cuba	Allen	2013-05-26 14:47:57.62
98	Chris	Bridges	2013-05-26 14:47:57.62
15	Cuba	Olivier	2013-05-26 14:47:57.62
160	Chris	Depp	2013-05-26 14:47:57.62
111	Cameron	Zellweger	2013-05-26 14:47:57.62
63	Cameron	Wray	2013-05-26 14:47:57.62
189	Cuba	Birch	2013-05-26 14:47:57.62

The status bar at the bottom indicates '67°F Sunny' and the time '3:42 PM 5/9/2022'.

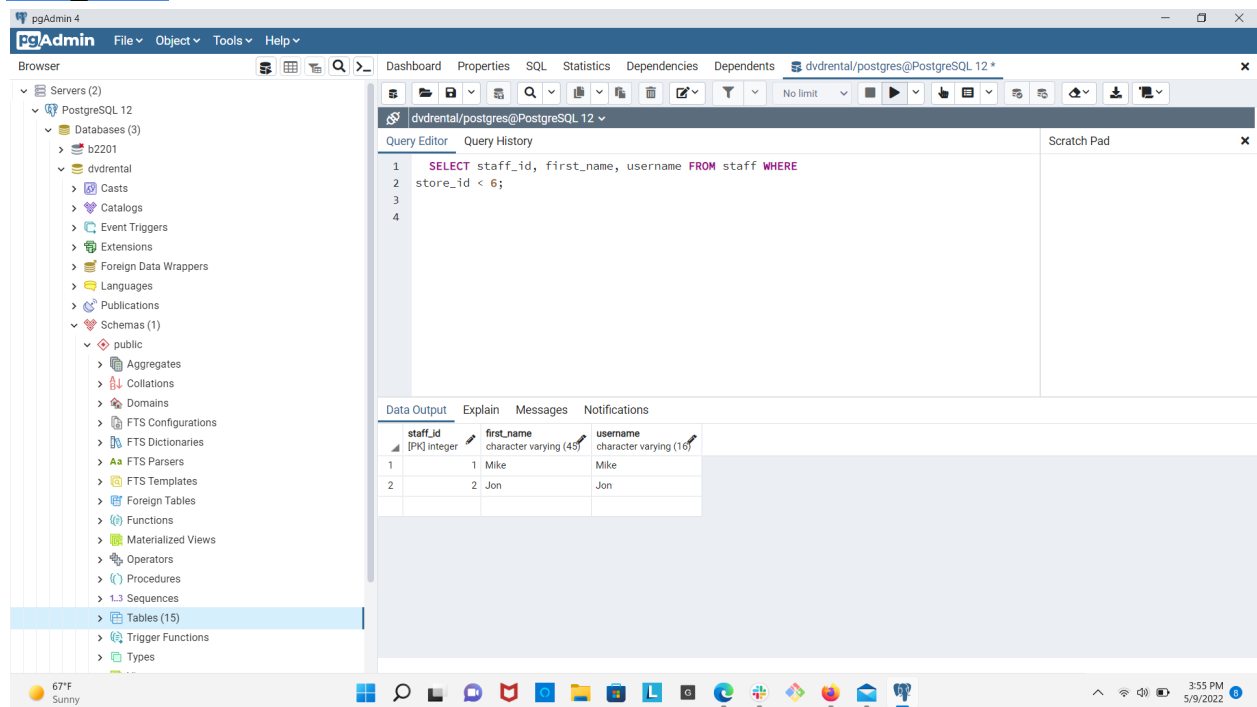
25. Query last name of customers in customer table whose first names are "John".

ANS. `SELECT last_name FROM customer WHERE first_name = 'John';`

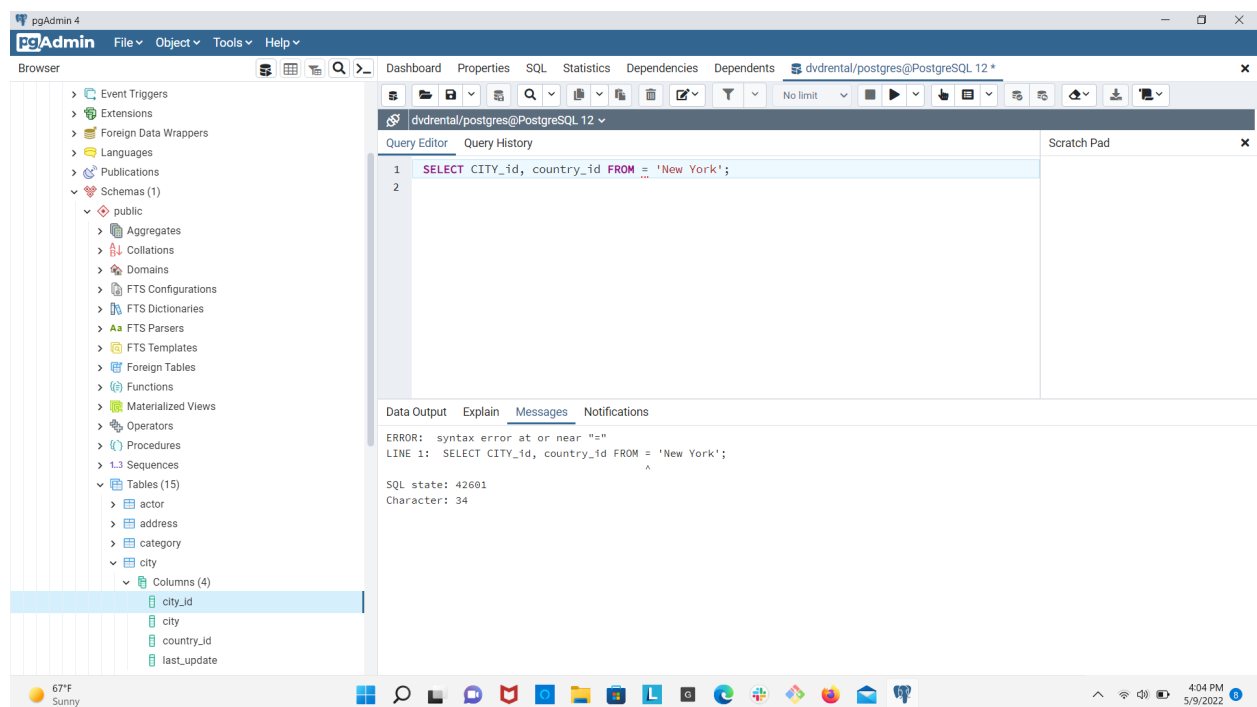
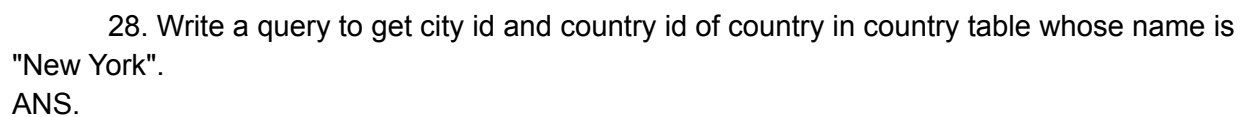


26. Write a query to get staff id, first name and username of staff in staff table whose store id is less than 6.

ANS. `SELECT staff_id, first_name, username FROM staff WHERE store_id < 6;`

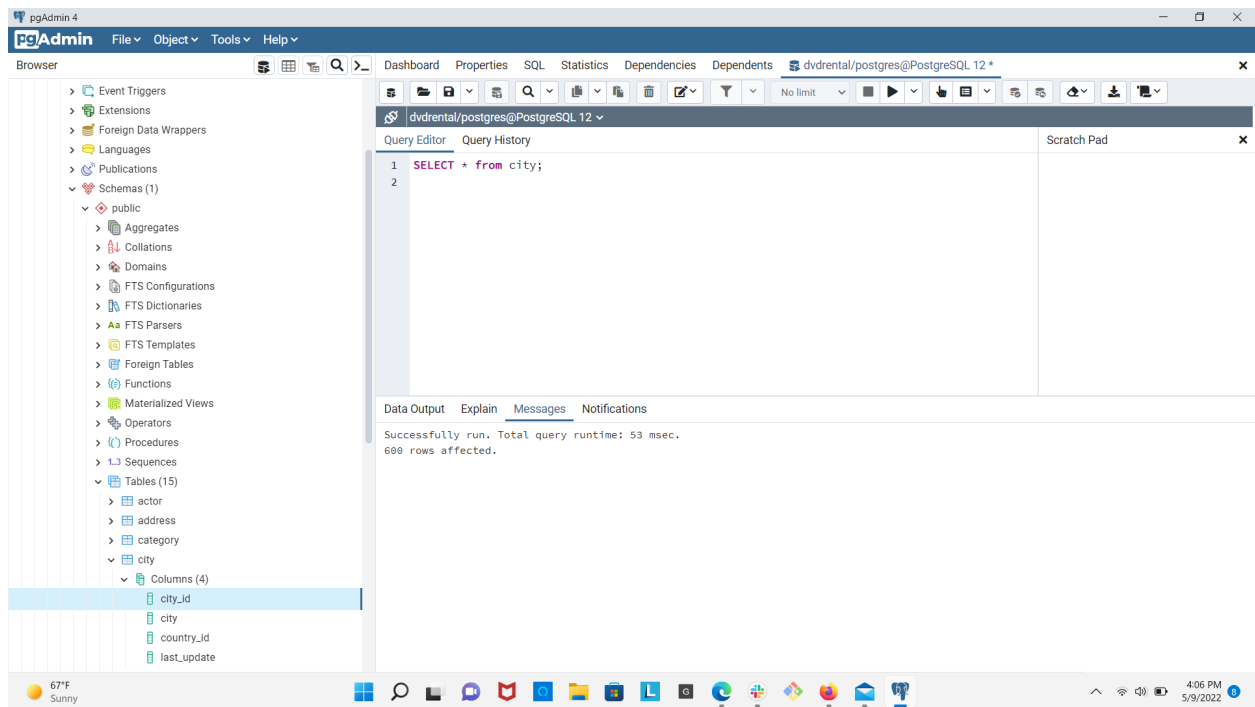


```
ANS. SELECT release_year, rental_duration, rental_rate FROM
film;
```



29. Write a query to get all data of city table.

ANS. `SELECT * from city;`



30. Write a query to get film id of film in film\_category table with category\_id 2.

ANS. `SELECT film_id from film_category WHERE category_id=2;`

