Query Editor | Query History

```
1  SELECT num
2  FROM sort_demo
3  ORDER BY num;
4
```

Data Output | Explain | Messages

	num integer
1	1
2	2
3	3
4	[null]

Query Editor | Query History

```
1  SELECT num
2  FROM sort_demo
3  ORDER BY num NULLS FIRST;
4
```

Data Output | Explain | Messages | Notificati

	num integer
1	[null]
2	1
3	2
4	3

Query Editor | Query History

```
1  SELECT num
2  FROM sort_demo
3  ORDER BY num NULLS LAST;
4
```

Data Output | Explain | Messages | Notifications

	num integer
1	1
2	2
3	3
4	[null]

Query Editor | Query History

```
1  SELECT num
2  FROM sort_demo
3  ORDER BY num DESC;
4
```

Data Output | Explain | Messages | N

	num integer
1	[null]
2	3
3	2
4	1

Query Editor | Query History

```
1  SELECT id, bcolor,fcolor
2  FROM distict_demo;
```

Data Output | Explain | Messages | Notifications

	id [PK] integer	bcolor character varying	fcolor character varying
1	1	red	red
2	2	red	red
3	3	red	[null]
4	4	[null]	red
5	5	red	green
6	6	red	blue

Query Editor | Query History

```
1  SELECT bcolor
2  FROM distict_demo
3  ORDER BY bcolor;
```

Data Output | Explain | Messages | Notificati

	bcolor character varying
1	blue
2	blue
3	blue
4	green
5	green
6	green
7	red
8	red
9	red
10	red
11	red
12	[null]

Query Editor | Query History

```
1  SELECT DISTINCT bcolor
2  FROM distict_demo
3  ORDER BY bcolor;
```

Data Output | Explain | Messages | Notifications

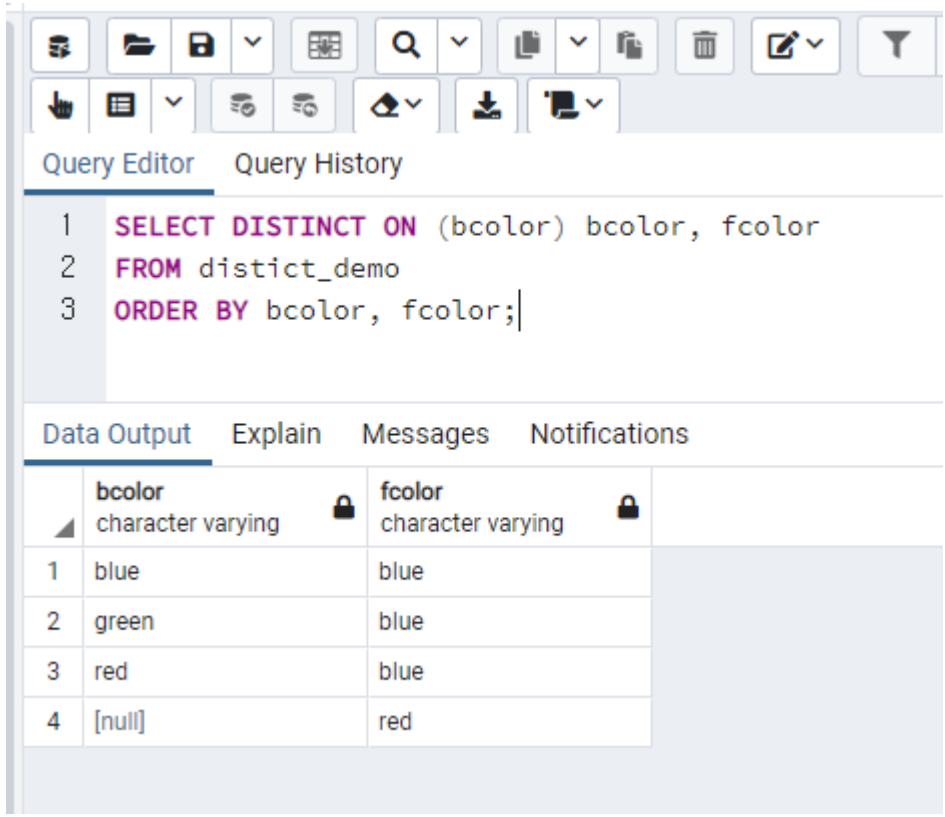
	bcolor character varying
1	blue
2	green
3	red
4	[null]

Query Editor | Query History

```
1  SELECT DISTINCT bcolor, fcolor
2  FROM distict_demo
3  ORDER BY bcolor, fcolor;
```

Data Output | Explain | Messages | Notifications

	bcolor character varying	fcolor character varying
1	blue	blue
2	blue	green
3	blue	red
4	green	blue
5	green	green
6	green	red
7	red	blue
8	red	green
9	red	red
10	red	[null]
11	[null]	red



bcolor한 다음 중복 그룹에 대한 반환 결과 집합의 첫번째 행을 나타냄

Query Editor | Query History

```
1  SELECT last_name, first_name
2  FROM customer
3  WHERE first_name In ('Ann','Anne','Annie');
4
```

Data Output | Explain | Messages | Notifications

	last_name character varying (45)	first_name character varying (45)
1	Evans	Ann
2	Powell	Anne
3	Russell	Annie

Operator	Description
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<> or !=	Not equal
AND	Logical operator AND
OR	Logical operator OR
IN	Return true if a value matches any value in a list
BETWEEN	Return true if a value is between a range of values
LIKE	Return true if a value matches a pattern
IS NULL	Return true if a value is NULL
NOT	Negate the result of other operators

Dashboard Properties SQL Statistics Dependencies

Query Editor Query History

```
1  SELECT last_name, first_name
2  FROM customer
3  WHERE first_name LIKE 'Ann%'
4
```

Data Output Explain Messages Notifications

	last_name character varying (45)	first_name character varying (45)
1	Hill	Anna
2	Evans	Ann
3	Powell	Anne
4	Russell	Annie
5	Olson	Annette

Query Editor Query History

```
1  SELECT first_name, last_name
2  FROM customer
3  WHERE first_name LIKE 'Bra%' AND
4  last_name <> 'Motley';
5
```

Data Output Explain Messages Notifications

	first_name character varying (45)	last_name character varying (45)
1	Brandy	Graves
2	Brandon	Huey
3	Brad	Mccurdy

Query Editor Query History

```
1  SELECT film_id, title, release_year
2  FROM film
3  ORDER BY film_id
4  LIMIT 4;
5
```

Data Output Explain Messages Notifications

	film_id [PK] integer	title character varying (255)	release_year integer
1	1	Academy Dinosaur	2006
2	2	Ace Goldfinger	2006
3	3	Adaptation Holes	2006
4	4	Affair Prejudice	2006

Film_id 를 4까지 본다

No limit

Query Editor Query History

```
1  SELECT film_id, title, release_year
2  FROM film
3  ORDER BY film_id
4  LIMIT 4 OFFSET 3;
5
```

Data Output Explain Messages Notifications

	film_id [PK] integer	title character varying (255)	release_year integer
1	4	Affair Prejudice	2006
2	5	African Egg	2006
3	6	Agent Truman	2006
4	7	Airplane Sierra	2006

Film_id 를 4 ~ 7(4+3)까지 본다

Query Editor | Query History

```sql
SELECT film_id, title, rental_rate
FROM film
ORDER BY rental_rate DESC
LIMIT 10;
```

Data Output | Explain | Messages | Notifications

	film_id [PK] integer	title character varying (255)	rental_rate numeric (4,2)
1	13	Ali Forever	4.99
2	20	Amelie Hellfighters	4.99
3	7	Airplane Sierra	4.99
4	10	Aladdin Calendar	4.99
5	2	Ace Goldfinger	4.99
6	8	Airport Pollock	4.99
7	98	Bright Encounters	4.99
8	133	Chamber Italian	4.99
9	384	Grosse Wonderful	4.99
10	21	American Circus	4.99

Query Editor | Query History

```sql
SELECT customer_id, rental_id, return_date
FROM rental
WHERE customer_id IN (1,2)
ORDER BY return_date DESC;
```

Data Output | Explain | Messages | Notifications

	customer_id smallint	rental_id [PK] integer	return_date timestamp without time zone
1	2	15145	2005-08-31 15:51:04
2	1	15315	2005-08-30 01:51:46
3	2	14743	2005-08-29 00:18:56
4	1	15298	2005-08-28 22:49:37
5	2	14475	2005-08-27 08:59:32

Query Editor | Query History

```sql
SELECT customer_id, rental_id, return_date
FROM rental
WHERE customer_id NOT IN (1,2);
```

Data Output | Explain | Messages | Notifications

# 데이터 타입 변환 함수
CAST(expression AS data_type(length))

Query Editor | Query History

```
1  SELECT customer_id
2  FROM rental
3  WHERE CAST (return_date AS DATE) = '2005-05-27'
4  ORDER BY customer_id;
```

Data Output | Explain | Messages | Notifications

	customer_id smallint
1	37
2	47
3	48
4	65
5	73
6	75

Dashboard | Properties | SQL | Statistics | Dependencies | Dependents | dvdrental/postgres

No limit

Query Editor | Query History

```
1  SELECT customer_id,first_name, last_name
2  FROM customer
3  WHERE customer_id IN(SELECT customer_id
4                       FROM rental
5                       WHERE CAST (return_date AS DATE) = '2005-05-27')
6  ORDER BY customer_id;
```

Data Output | Explain | Messages | Notifications

	customer_id [PK] integer	first_name character varying (45)	last_name character varying (45)
1	37	Pamela	Baker
2	47	Frances	Parker
3	48	Ann	Evans
4	65	Rose	Howard
5	73	Beverly	Brooks
6	75	Tammy	Sanders
7	93	Phyllis	Foster
8	114	Grace	Ellis

Return_date를 DATA 형식으로 변환

Dashboard Properties SQL Statistics Dependencies D

Query Editor Query History

```
1  SELECT customer_id,payment_id,amount
2  FROM payment
3  WHERE amount NOT BETWEEN 8 AND 9
4
```

Data Output Explain Messages Notifications

	customer_id smallint	payment_id [PK] integer	amount numeric (5,2)
1	343	17517	8.99
2	347	17529	8.99
3	347	17532	8.99
4	348	17535	8.99
5	349	17540	8.99
6	379	17648	8.99

Query Editor Query History

```
1  SELECT first_name, last_name
2  FROM customer
3  WHERE first_name LIKE '%er'
4  ORDER BY first_name;
```

Data Output Explain Messages Notifications

	first_name character varying (45)	last_name character varying (45)
1	Alexander	Fennell
2	Amber	Dixon
3	Chester	Benner
4	Christopher	Greco
5	Elmer	Noe
6	Esther	Crawford
7	Heather	Morris
8	Javier	Elrod

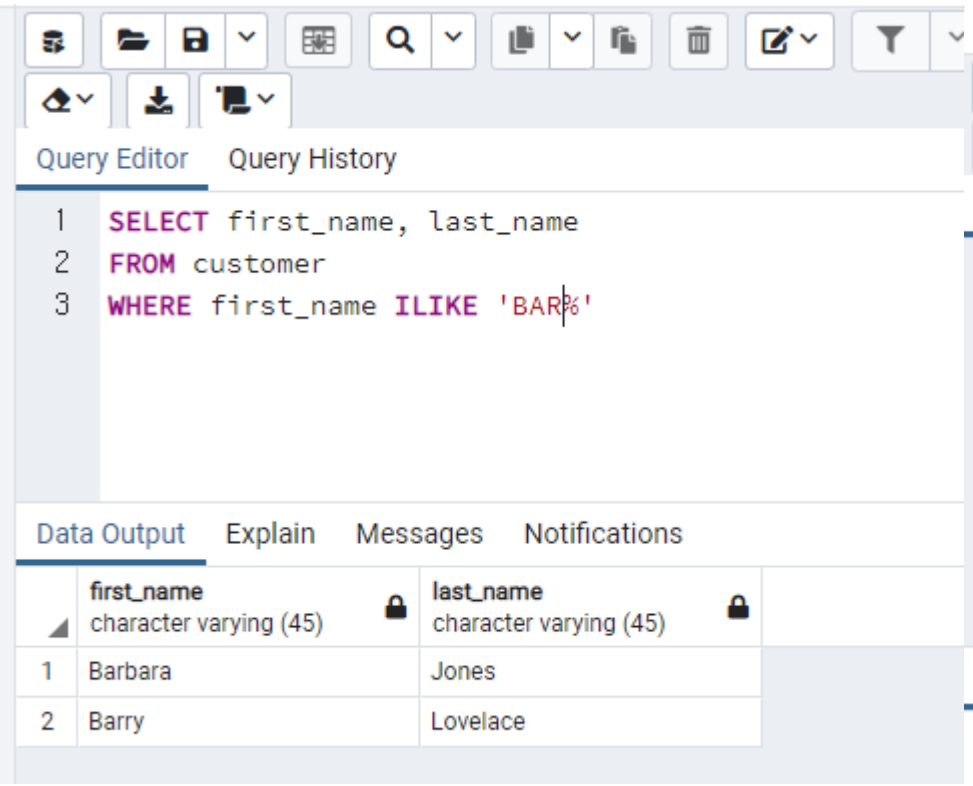
Dashboard Properties SQL Statistics Dependencies D

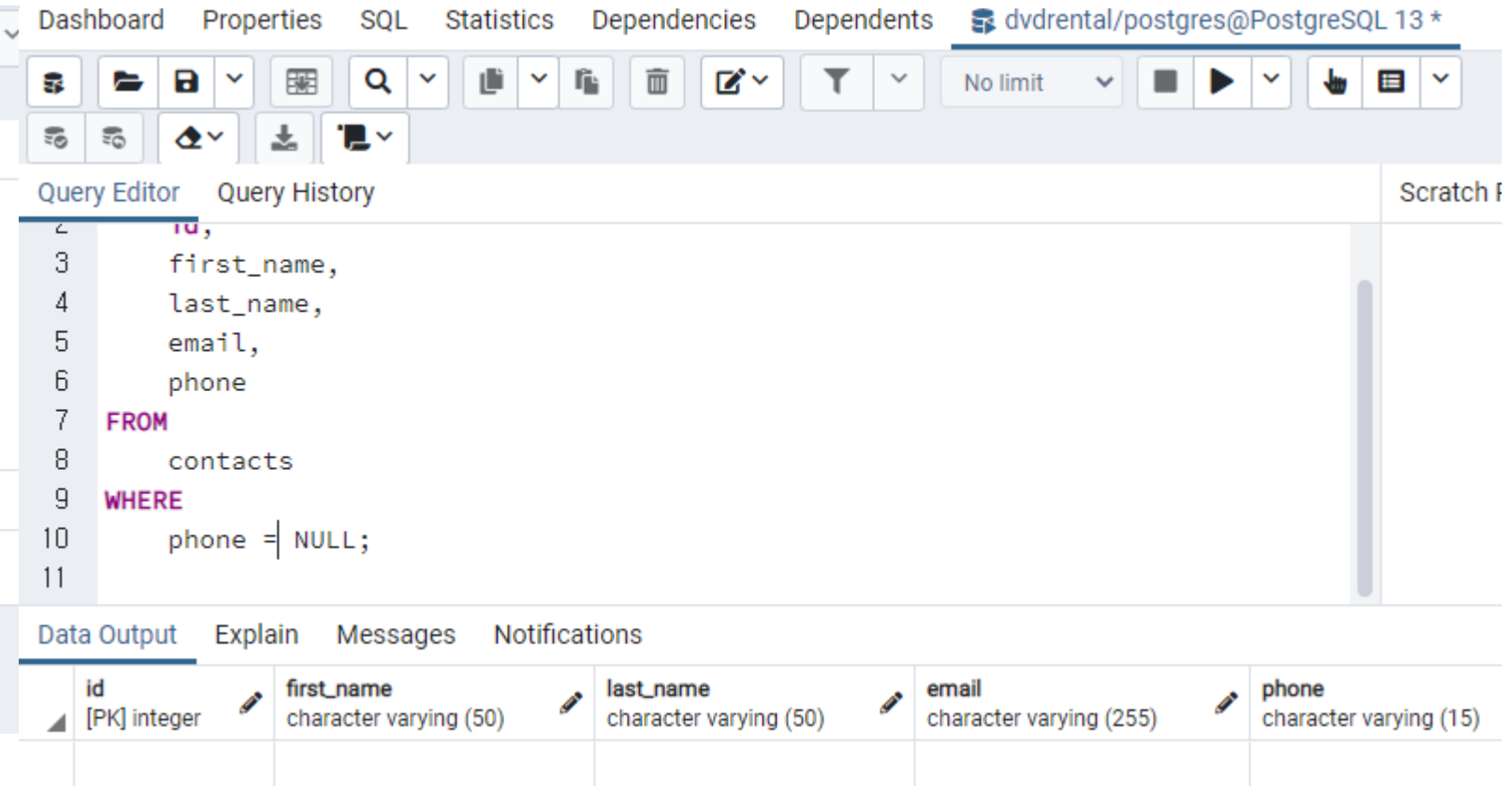
Query Editor Query History

```
1  SELECT first_name, last_name
2  FROM customer
3  WHERE first_name NOT LIKE 'Jen%'
4  ORDER BY first_name;
```

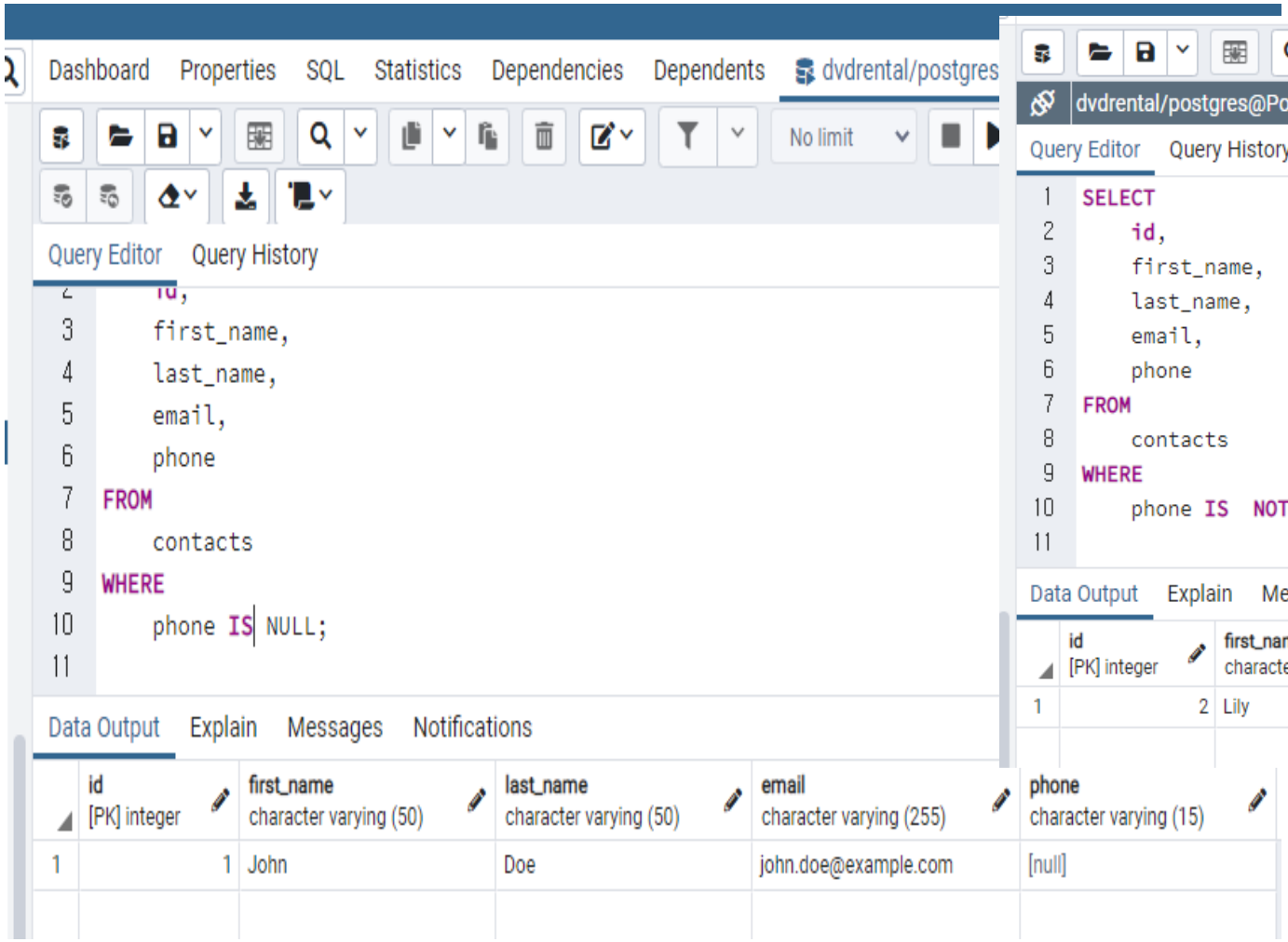
Data Output Explain Messages Notifications

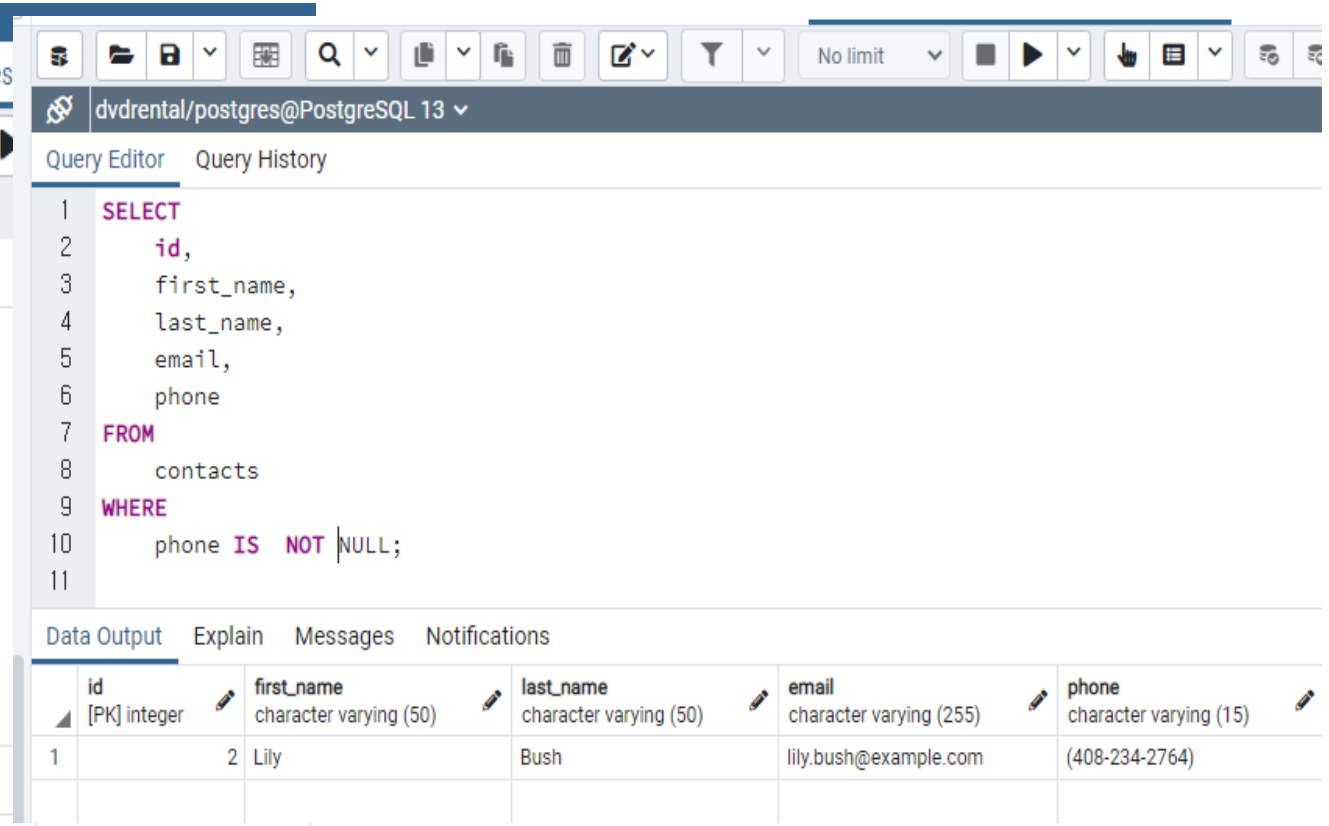
	first_name character varying (45)	last_name character varying (45)
1	Aaron	Selby





ILIKE = LIKE 기능 + 대소문자 혼용


Dashboard
Properties
SQL
Statistics
Dependencies
Dependents
dvdrental/postgres
No limit
Query Editor
Query History
id,
first_name,
last_name,
email,
phone
FROM
contacts
WHERE
phone IS NULL;
Data Output
Explain
Messages
Notifications
id
[PK] integer
first_name
character varying (50)
last_name
character varying (50)
email
character varying (255)
phone
character varying (15)
1
1
John
Doe
john.doe@example.com
[null]


No limit
dvdrental/postgres@PostgreSQL 13
Query Editor
Query History
SELECT
id,
first_name,
last_name,
email,
phone
FROM
contacts
WHERE
phone IS NOT NULL;
Data Output
Explain
Messages
Notifications
id
[PK] integer
first_name
character varying (50)
last_name
character varying (50)
email
character varying (255)
phone
character varying (15)
1
2
Lily
Bush
lily.bush@example.com
(408-234-2764)

Query Editor | Query History

```
1  SELECT film_id, title
2  FROM film
3  ORDER BY title
4  OFFSET 5 ROWS
5  FETCH FIRST 7 ROWS ONLY;
6
7
```

Data Output | Explain | Messages | Notifications

	film_id [PK] integer	title character varying (255)
1	6	Agent Truman
2	7	Airplane Sierra
3	8	Airport Pollock
4	9	Alabama Devil
5	10	Aladdin Calendar
6	11	Alamo Videotape
7	12	Alaska Phantom

```
OFFSET start { ROW | ROWS }
FETCH { FIRST | NEXT } [ row_count ] { ROW | ROWS } ONLY
```

Start = 0 or 양수
Row_count 1 이상