DBMS

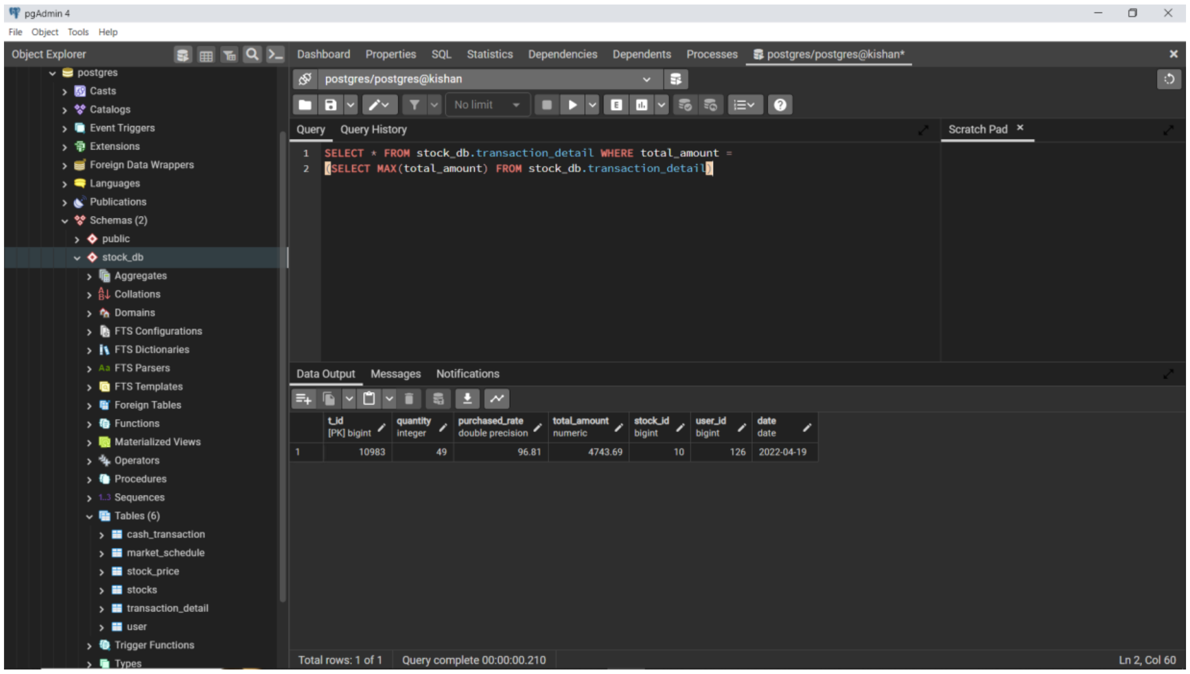
ASSIGNMENT 2

Name : Kishan R Vaghamashi Student

ID: 202312014

Display record for the maximum total amount in transaction

SELECT * FROM stock_db.transaction_detail WHERE total_amount = (SELECT MAX(total_amount) FROM stock_db.transaction_detail)



```
SELECT * FROM stock_db.transaction_detail WHERE total_amount =
(SELECT MAX(total_amount) FROM stock_db.transaction_detail)
```

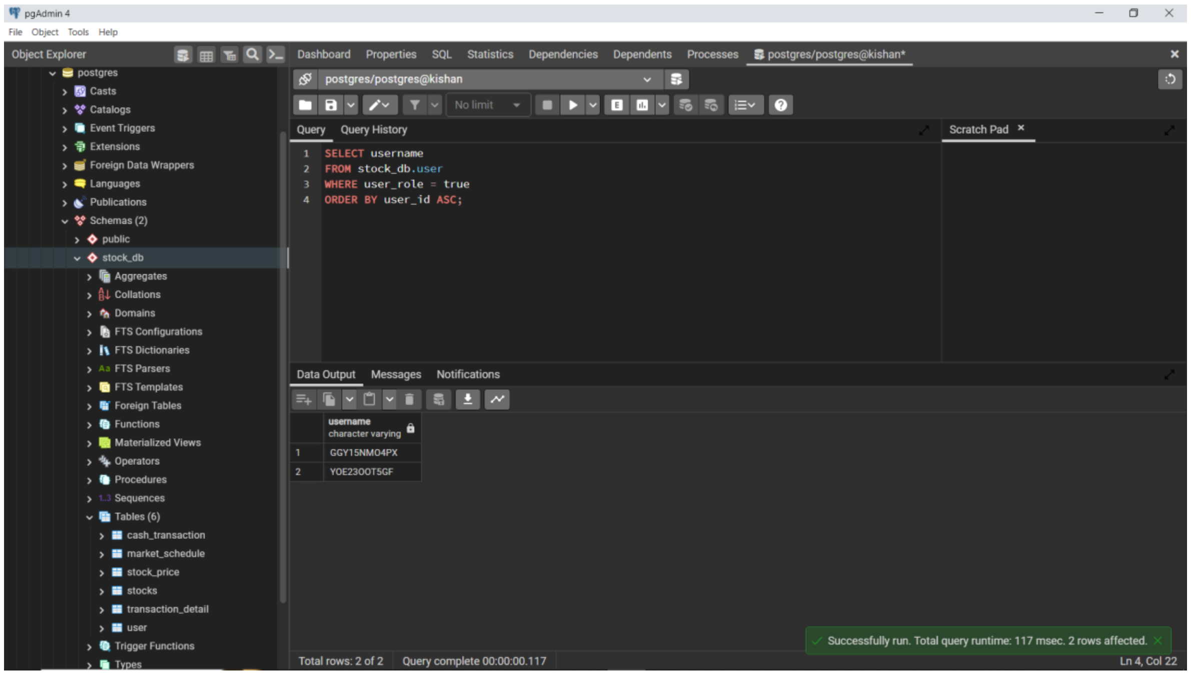
	t_id [PK] bigint	quantity integer	purchased_rate double precision	total_amount numeric	stock_id bigint	user_id bigint	date date
1	10983	49	96.81	4743.69	10	126	2022-04-19

Total rows: 1 of 1   Query complete 00:00:00.210   Ln 2, Col 60

1.

2. Find the username of the user in ascending order of user_id for admin

SELECT username FROM stock_db.user WHERE user_role = true ORDER BY user_id ASC;



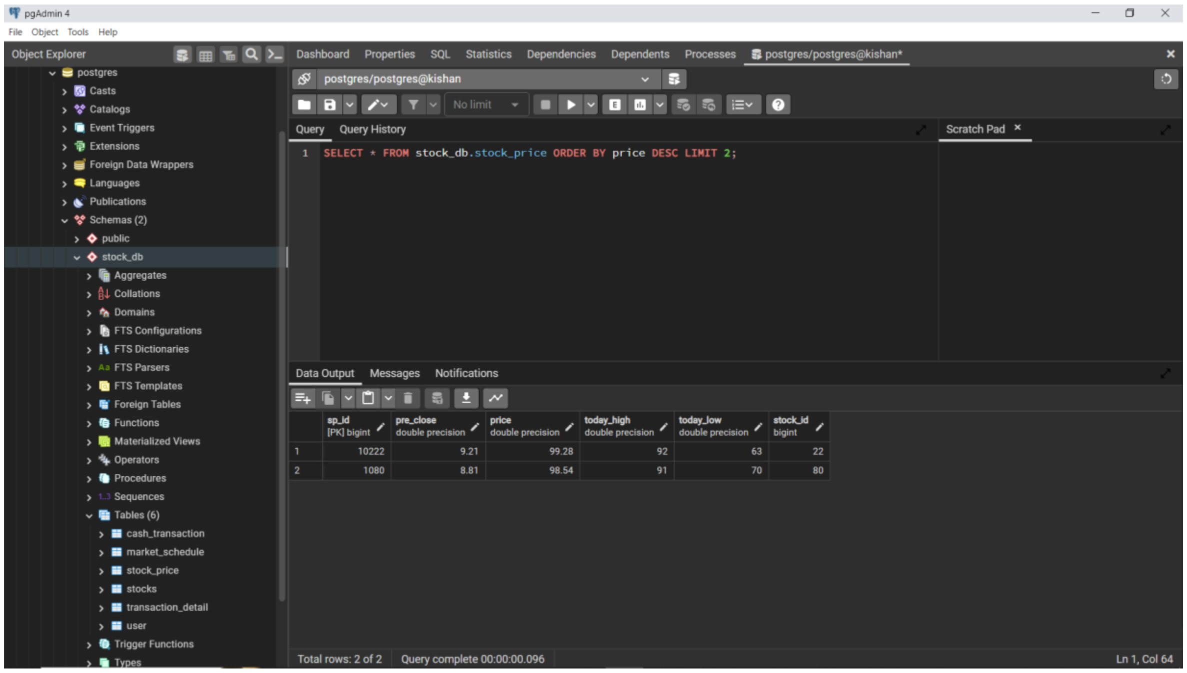
```
SELECT username
FROM stock_db.user
WHERE user_role = true
ORDER BY user_id ASC;
```

	username character varying
1	GGY15NMO4PX
2	YOE23OOT5GF

```
Successfully run. Total query runtime: 117 msec. 2 rows affected.
Total rows: 2 of 2    Query complete 00:00:00.117    Ln 4, Col 22
```

Select details of the top 2 highest stock prices from stocks

SELECT * FROM stock_db.stock_price ORDER BY price DESC LIMIT 2;



```
SELECT * FROM stock_db.stock_price ORDER BY price DESC LIMIT 2;
```

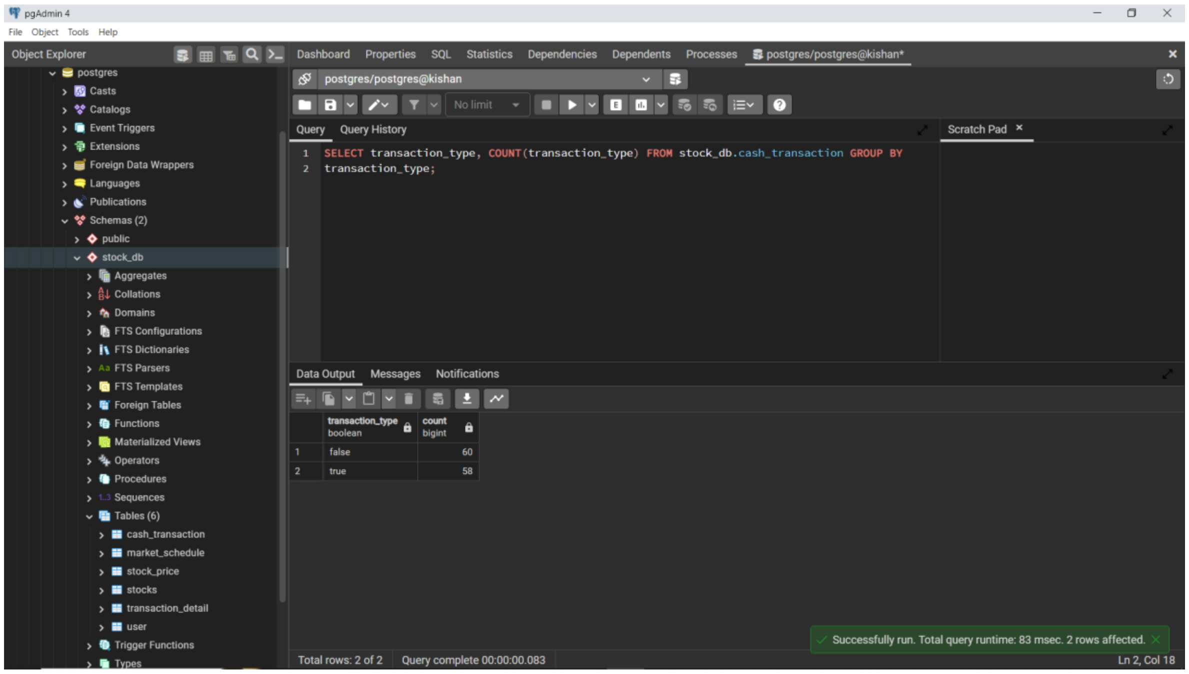
	sp_id [PK] bigint	pre_close double precision	price double precision	today_high double precision	today_low double precision	stock_id bigint
1	10222	9.21	99.28	92	63	22
2	1080	8.81	98.54	91	70	80

```
Total rows: 2 of 2    Query complete 00:00:00.096    Ln 1, Col 64
```

3.

4. Print count of different types of cash transactions

```
SELECT transaction_type, COUNT(transaction_type) FROM stock_db.cash_transaction GROUP BY
transaction_type;
```



```
SELECT transaction_type, COUNT(transaction_type) FROM stock_db.cash_transaction GROUP BY
transaction_type;
```

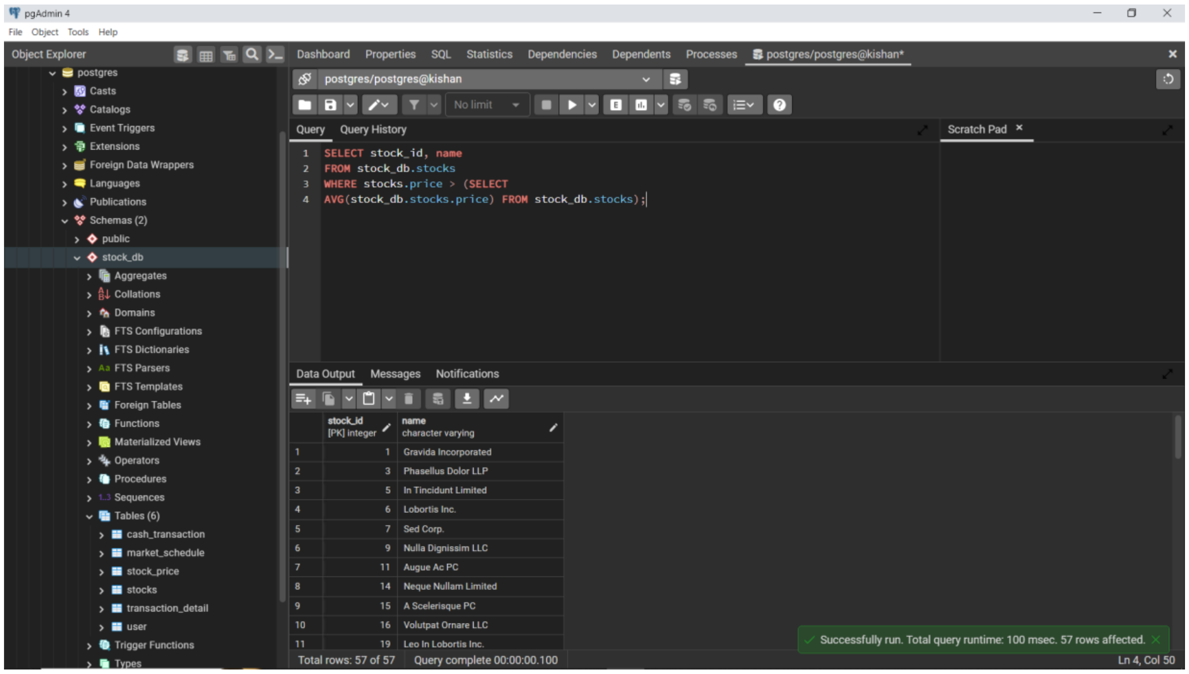
	transaction_type boolean	count bigint
1	false	60
2	true	58

Successfully run. Total query runtime: 83 msec. 2 rows affected.

Total rows: 2 of 2   Query complete 00:00:00.083   Ln 2, Col 18

Print stock id and name whose price exceeds the average price value

SELECT stock_id, name FROM stock_db.stocks WHERE stocks.price > (SELECT AVG(stock_db.stocks.price) FROM stock_db.stocks);



```
SELECT stock_id, name
FROM stock_db.stocks
WHERE stocks.price > (SELECT
AVG(stock_db.stocks.price) FROM stock_db.stocks);
```

	stock_id [PK] integer	name character varying
1	1	Gravida Incorporated
2	3	Phasellus Dolor LLP
3	5	In Tincidunt Limited
4	6	Lobortis Inc.
5	7	Sed Corp.
6	9	Nulla Dignissim LLC
7	11	Augue Ac PC
8	14	Neque Nullam Limited
9	15	A Scelerisque PC
10	16	Volutpat Ornare LLC
11	19	Leo In Lobortis Inc.

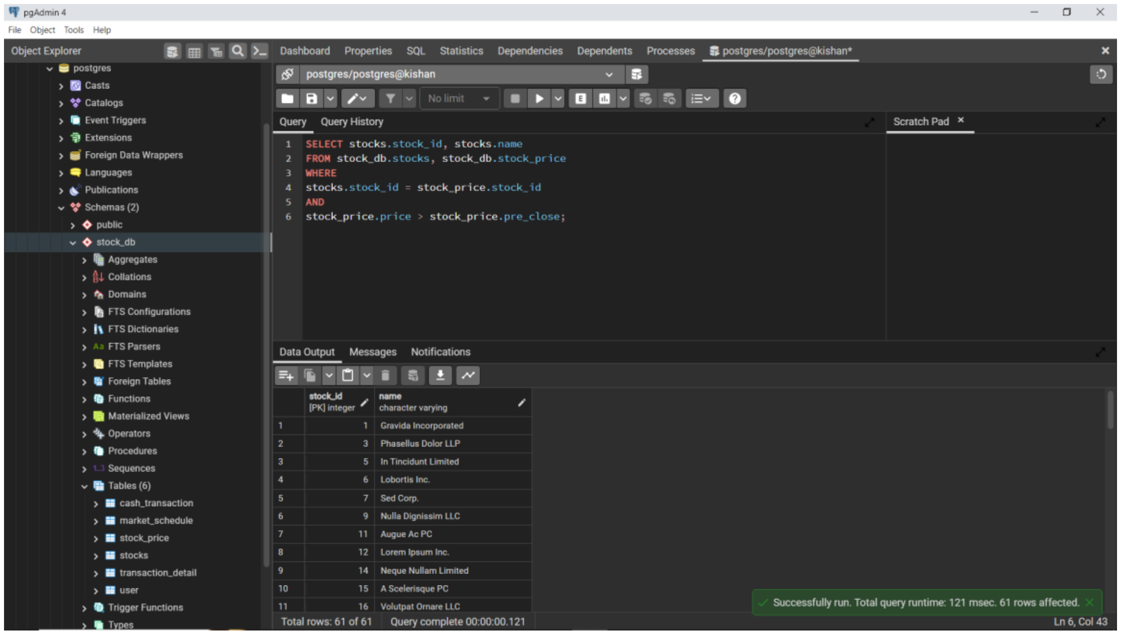
Successfully run. Total query runtime: 100 msec. 57 rows affected.

Total rows: 57 of 57 Query complete 00:00:00.100

5.

6. Find stock names and ids whose price exceeds the previous close

```
SELECT stocks.stock_id, stocks.name FROM stock_db.stocks, stock_db.stock_price WHERE
stocks.stock_id = stock_price.stock_id AND stock_price.price > stock_price.pre_close;
```



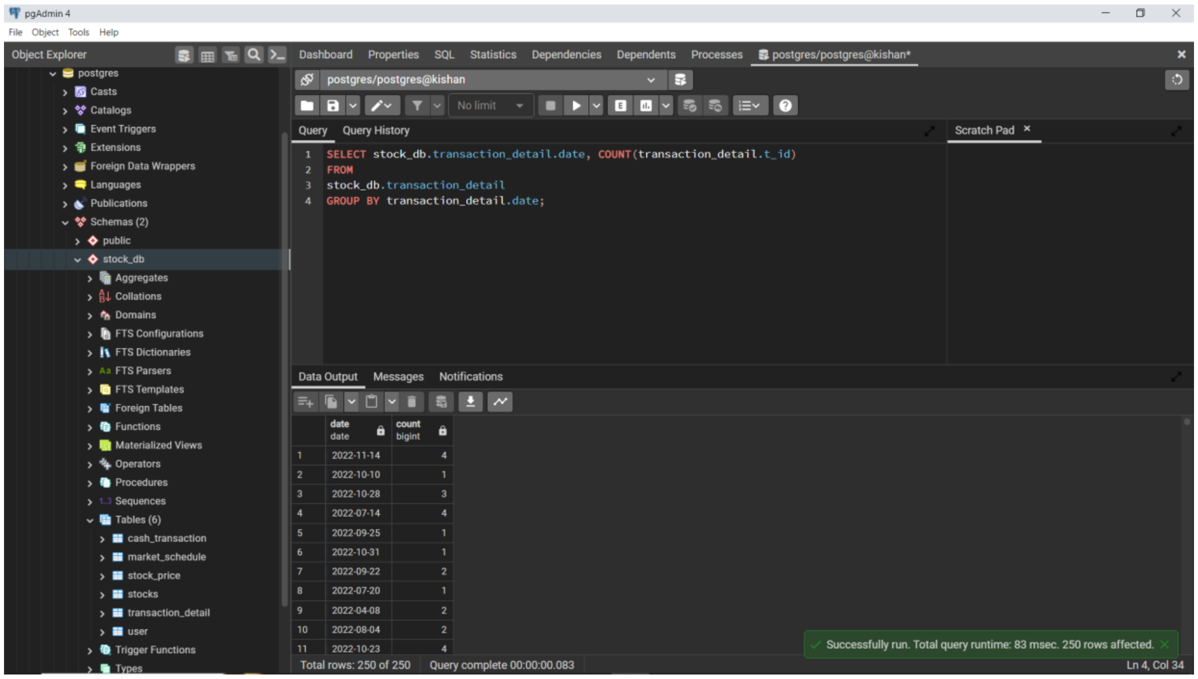
```
SELECT stocks.stock_id, stocks.name
FROM stock_db.stocks, stock_db.stock_price
WHERE
stocks.stock_id = stock_price.stock_id
AND
stock_price.price > stock_price.pre_close;
```

	stock_id [PK] integer	name character varying
1	1	Gravida Incorporated
2	3	Phasellus Dolor LLP
3	5	In Tincidunt Limited
4	6	Lobortis Inc.
5	7	Sed Corp.
6	9	Nulla Dignissim LLC
7	11	Augue Ac PC
8	12	Lorem Ipsum Inc.
9	14	Neque Nullam Limited
10	15	A Scelerisque PC
11	16	Volutpat Ornare LLC

```
Successfully run. Total query runtime: 121 msec. 61 rows affected.
Total rows: 61 of 61    Query complete 00:00:00.121    Ln 6, Col 43
```

Find the total quantity of stocks date-wise whose transaction is done on a particular date

```
SELECT stock_db.transaction_detail.date, COUNT(transaction_detail.t_id) FROM
stock_db.transaction_detail GROUP BY transaction_detail.date;
```



```
SELECT stock_db.transaction_detail.date, COUNT(transaction_detail.t_id)
FROM
stock_db.transaction_detail
GROUP BY transaction_detail.date;
```

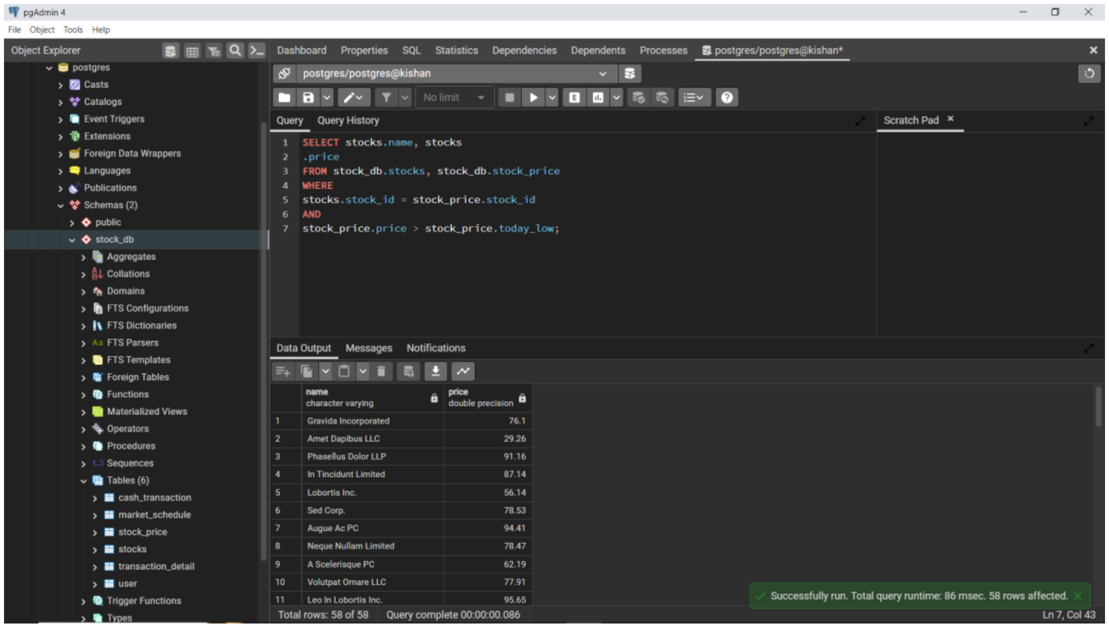
	date date	count bigint
1	2022-11-14	4
2	2022-10-10	1
3	2022-10-28	3
4	2022-07-14	4
5	2022-09-25	1
6	2022-10-31	1
7	2022-09-22	2
8	2022-07-20	1
9	2022-04-08	2
10	2022-08-04	2
11	2022-10-23	4

```
Successfully run. Total query runtime: 83 msec. 250 rows affected.
Total rows: 250 of 250    Query complete 00:00:00.083    Ln 4, Col 34
```

7.

8. Find stock name and price whose price is less than the average price of today_low

SELECT stocks.name, stocks.price FROM stock_db.stocks, stock_db.stock_price WHERE stocks.stock_id = stock_price.stock_id AND stock_price.price > stock_price.today_low;



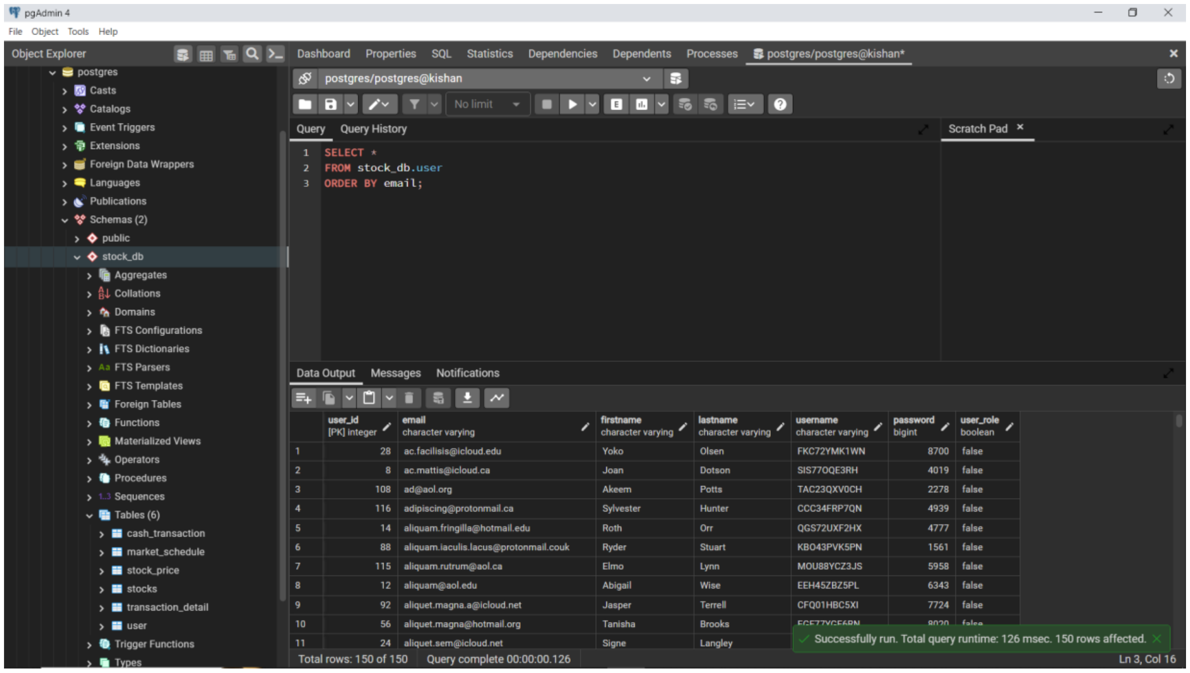
```
SELECT stocks.name, stocks
.price
FROM stock_db.stocks, stock_db.stock_price
WHERE
stocks.stock_id = stock_price.stock_id
AND
stock_price.price > stock_price.today_low;
```

	name character varying	price double precision
1	Gravida Incorporated	76.1
2	Amet Dapibus LLC	29.26
3	Phasellus Dolor LLP	91.16
4	In Tincidunt Limited	87.14
5	Lobortis Inc.	56.14
6	Sed Corp.	78.53
7	Augue Ac PC	94.41
8	Neque Nullam Limited	78.47
9	A Scelerisque PC	62.19
10	Volutpat Ornare LLC	77.91
11	Leo In Lobortis Inc.	95.65

```
Successfully run. Total query runtime: 86 msec. 58 rows affected.
Total rows: 58 of 58    Query complete 00:00:00.086    Ln 7, Col 43
```

Show table of users sorted email-wise

SELECT * FROM stock_db.user ORDER BY email;



```
SELECT *
FROM stock_db.user
ORDER BY email;
```

	user_id [PK] integer	email character varying	firstname character varying	lastname character varying	username character varying	password bigint	user_role boolean
1	28	ac.facilisis@icloud.edu	Yoko	Olsen	FKC72YMK1WN	8700	false
2	8	ac.mattis@icloud.ca	Joan	Dotson	SIS77OQE3RH	4019	false
3	108	ad@aol.org	Akeem	Potts	TAC23QXV0CH	2278	false
4	116	adipiscing@protonmail.ca	Sylvester	Hunter	CCC34FRP7QN	4939	false
5	14	aliquam.fringilla@hotmail.edu	Roth	Orr	QGS72UXF2HX	4777	false
6	88	aliquam.iaculis.lacus@protonmail.couk	Ryder	Stuart	KBO43PVK5PN	1561	false
7	115	aliquam.rutrum@aol.ca	Elmo	Lynn	MOU88YCZ3JS	5958	false
8	12	aliquam@aol.edu	Abigail	Wise	EEH45ZBZ5PL	6343	false
9	92	aliquet.magna.a@icloud.net	Jasper	Terrell	CFQ01HBC5XI	7724	false
10	56	aliquet.magna@hotmail.org	Tanisha	Brooks	[illegible]	8020	false
11	24	aliquet.sem@icloud.net	Signe	Langley	[illegible]	[illegible]	[illegible]

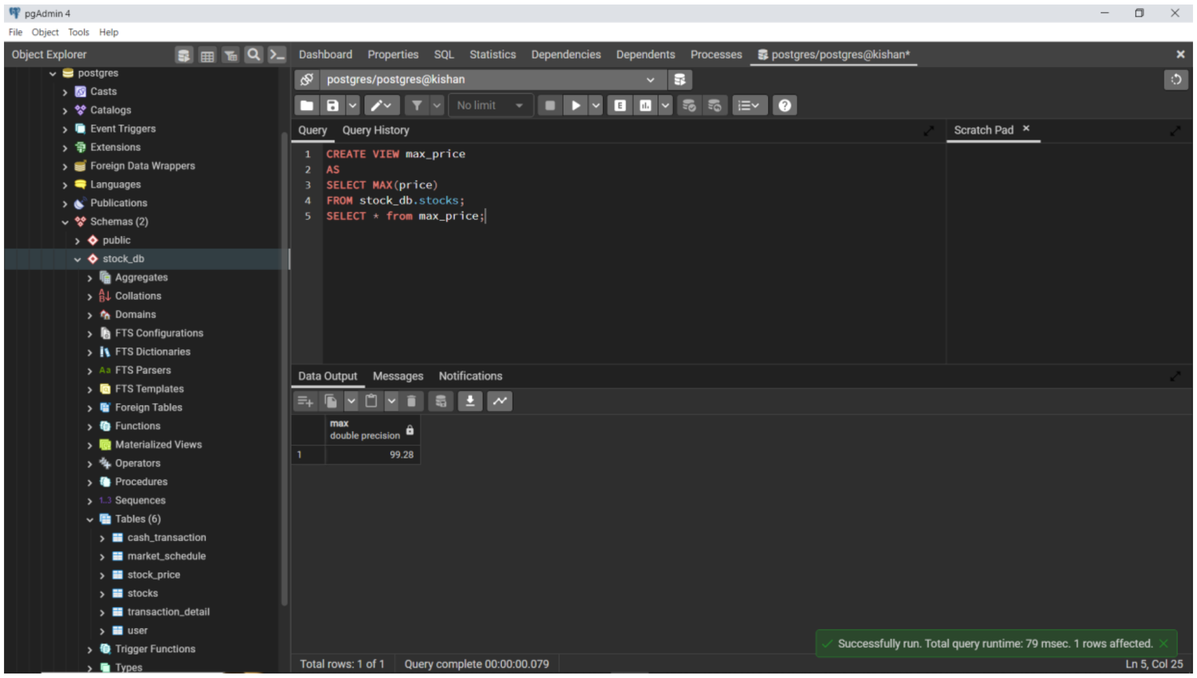
```
Successfully run. Total query runtime: 126 msec. 150 rows affected.
Total rows: 150 of 150    Query complete 00:00:00.126    Ln 3, Col 16
```

9.

10. create a view of maximum price

```
CREATE VIEW max_price AS SELECT MAX(price) FROM stock_db.stocks;

SELECT * from max_price;
```



```
CREATE VIEW max_price
AS
SELECT MAX(price)
FROM stock_db.stocks;
SELECT * from max_price;
```

	max double precision
1	99.28

Successfully run. Total query runtime: 79 msec. 1 rows affected.

Total rows: 1 of 1 Query complete 00:00:00.079 Ln 5, Col 25

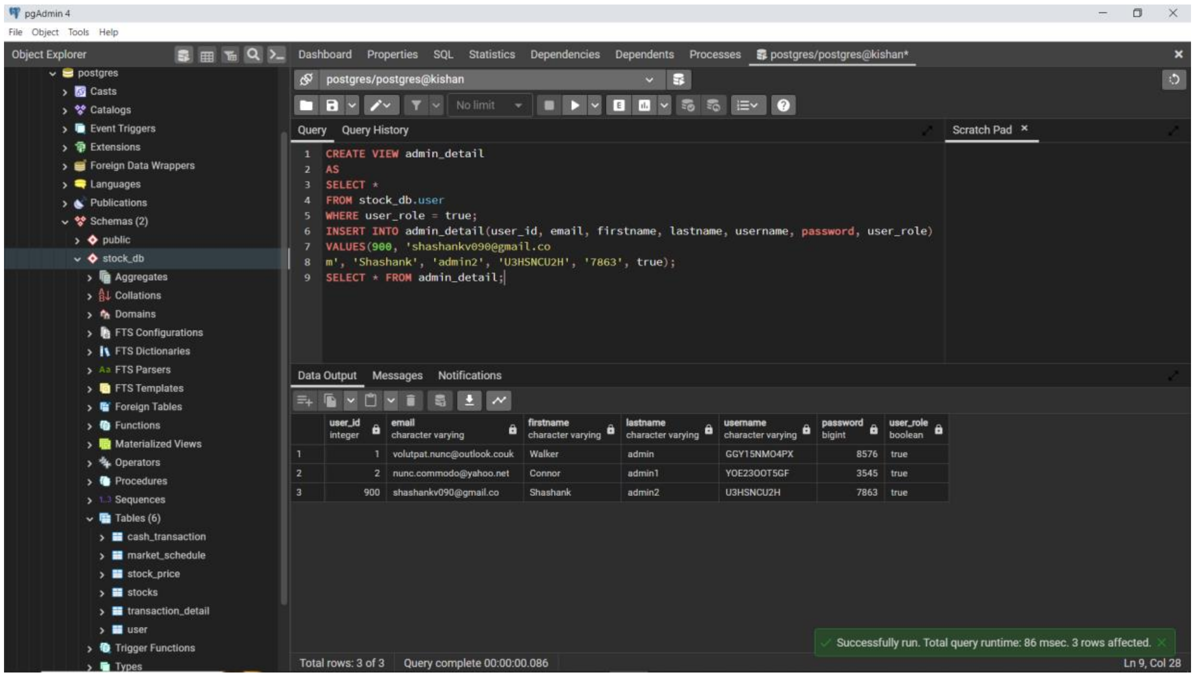
11. create a view for admin details, then insert and display some new admin to the recently created view.

CREATE VIEW admin_detail AS SELECT * FROM stock_db.user WHERE user_role = true;

INSERT INTO admin_detail(user_id, email, firstname, lastname, username, password, user_role)

VALUES(900, 'shashankv090@gmail.com', 'Shashank', 'admin2', 'U3HSNCU2H', '7863', true);

SELECT * FROM admin_detail;



```
CREATE VIEW admin_detail
AS
SELECT *
FROM stock_db.user
WHERE user_role = true;
INSERT INTO admin_detail(user_id, email, firstname, lastname, username, password, user_role)
VALUES(900, 'shashankv090@gmail.co
m', 'Shashank', 'admin2', 'U3HSNCU2H', '7863', true);
SELECT * FROM admin_detail;
```

	user_id integer	email character varying	firstname character varying	lastname character varying	username character varying	password bigint	user_role boolean
1	1	volutpat.nunc@outlook.couk	Walker	admin	GGY15NMO4PX	8576	true
2	2	nunc.commodo@yahoo.net	Connor	admin1	YOE23OOT5GF	3545	true
3	900	shashankv090@gmail.co	Shashank	admin2	U3HSNCU2H	7863	true

```
Successfully run. Total query runtime: 86 msec. 3 rows affected.
Total rows: 3 of 3    Query complete 00:00:00.086    Ln 9, Col 28
```

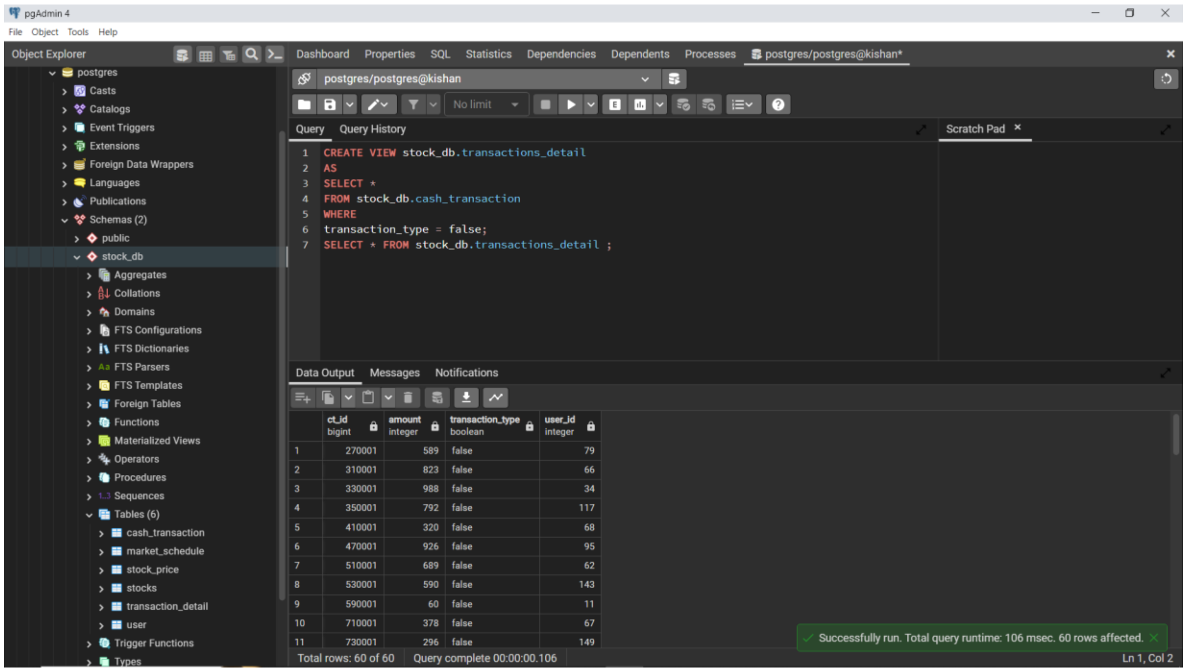
12. Create and display a view of all the details of transactions whose type is a debit (debit = false)

CREATE VIEW stock_db.transactions_detail AS SELECT * FROM stock_db.cash_transaction WHERE transaction_type = false;

SELECT * FROM stock_db.transactions_detail ;

C

REATE VIEW stock_db.transactions_detail AS SELECT * FROM stock_db.cash_transaction



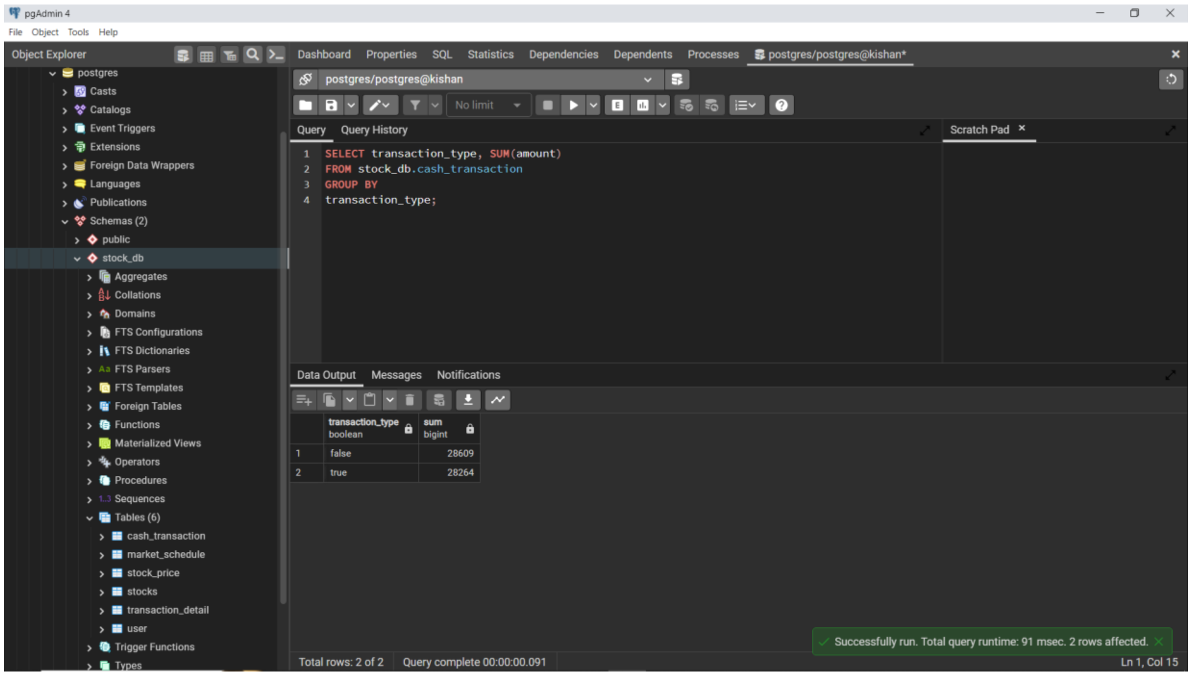
```
CREATE VIEW stock_db.transactions_detail
AS
SELECT *
FROM stock_db.cash_transaction
WHERE
transaction_type = false;
SELECT * FROM stock_db.transactions_detail ;
```

	ct_id bigint	amount integer	transaction_type boolean	user_id integer
1	270001	589	false	79
2	310001	823	false	66
3	330001	988	false	34
4	350001	792	false	117
5	410001	320	false	68
6	470001	926	false	95
7	510001	689	false	62
8	530001	590	false	143
9	590001	60	false	11
10	710001	378	false	67
11	730001	296	false	149

```
Successfully run. Total query runtime: 106 msec. 60 rows affected.
Total rows: 60 of 60    Query complete 00:00:00.106
```

13. Display the sum of the total amount transaction-wise

SELECT transaction_type, SUM(amount) FROM stock_db.cash_transaction GROUP BY transaction_type;



```
SELECT transaction_type, SUM(amount)
FROM stock_db.cash_transaction
GROUP BY
transaction_type;
```

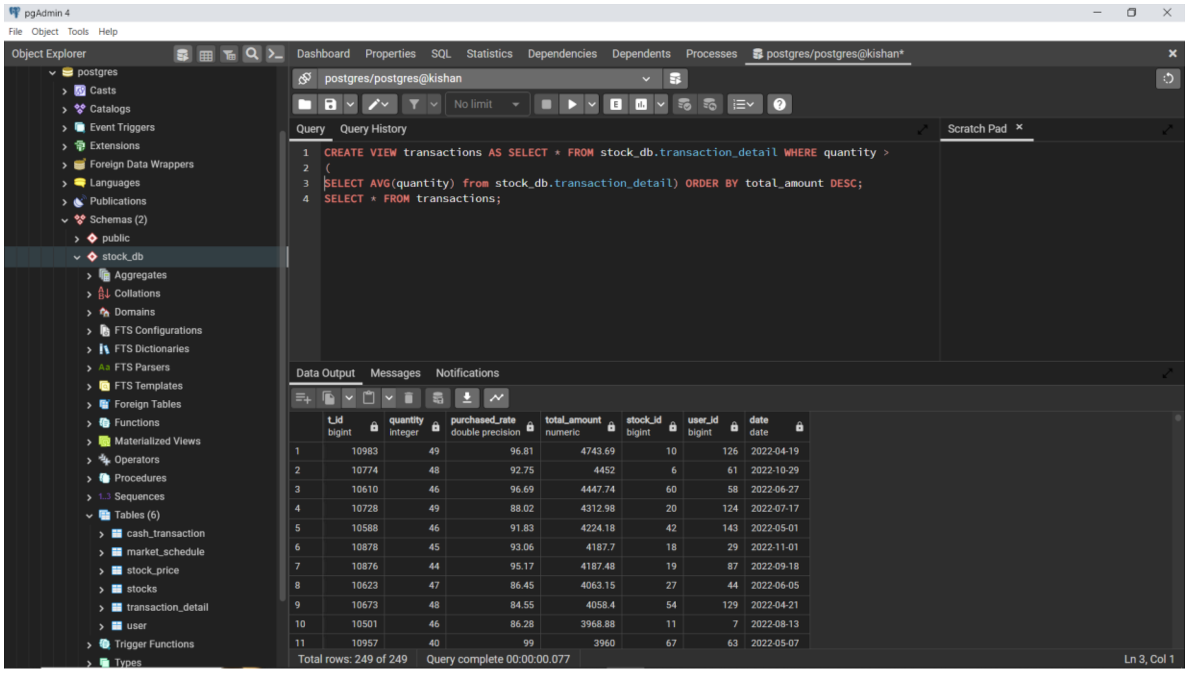
	transaction_type boolean	sum bigint
1	false	28609
2	true	28264

```
Successfully run. Total query runtime: 91 msec. 2 rows affected.
Total rows: 2 of 2    Query complete 00:00:00.091
```

14. Create a view with transactions made whose quantity exceeds the average of all quantities in descending order of amount

CREATE VIEW transactions AS SELECT * FROM stock_db.transaction_detail WHERE quantity > (SELECT AVG(quantity) from stock_db.transaction_detail) ORDER BY total_amount DESC;

SELECT * FROM transactions;



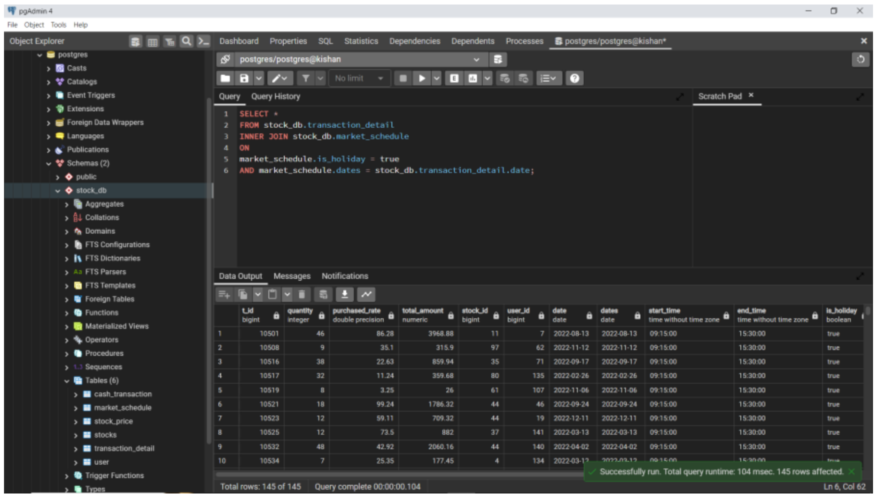
```
CREATE VIEW transactions AS SELECT * FROM stock_db.transaction_detail WHERE quantity >
(
SELECT AVG(quantity) from stock_db.transaction_detail) ORDER BY total_amount DESC;
SELECT * FROM transactions;
```

	t_id bigint	quantity integer	purchased_rate double precision	total_amount numeric	stock_id bigint	user_id bigint	date date
1	10983	49	96.81	4743.69	10	126	2022-04-19
2	10774	48	92.75	4452	6	61	2022-10-29
3	10610	46	96.69	4447.74	60	58	2022-06-27
4	10728	49	88.02	4312.98	20	124	2022-07-17
5	10588	46	91.83	4224.18	42	143	2022-05-01
6	10878	45	93.06	4187.7	18	29	2022-11-01
7	10876	44	95.17	4187.48	19	87	2022-09-18
8	10623	47	86.45	4063.15	27	44	2022-06-05
9	10673	48	84.55	4058.4	54	129	2022-04-21
10	10501	46	86.28	3968.88	11	7	2022-08-13
11	10957	40	99	3960	67	63	2022-05-07

Total rows: 249 of 249 Query complete 00:00:00.077 Ln 3, Col 1

15. Display all the transaction details done on the holiday

SELECT * FROM stock_db.transaction_detail INNER JOIN stock_db.market_schedule ON market_schedule.is_holiday = true AND market_schedule.dates = stock_db.transaction_detail.date;



```
SELECT *
FROM stock_db.transaction_detail
INNER JOIN stock_db.market_schedule
ON
market_schedule.is_holiday = true
AND market_schedule.dates = stock_db.transaction_detail.date;
```

	t_id bigint	quantity integer	purchased_rate double precision	total_amount numeric	stock_id bigint	user_id bigint	date date	dates date	start_time time without time zone	end_time time without time zone	is_holiday boolean
1	10501	46	86.28	3968.88	11	7	2022-08-13	2022-08-13	09:15:00	15:30:00	true
2	10508	9	35.1	315.9	97	62	2022-11-12	2022-11-12	09:15:00	15:30:00	true
3	10516	38	22.63	859.94	35	71	2022-09-17	2022-09-17	09:15:00	15:30:00	true
4	10517	32	11.24	359.68	80	135	2022-02-26	2022-02-26	09:15:00	15:30:00	true
5	10519	8	3.25	26	61	107	2022-11-06	2022-11-06	09:15:00	15:30:00	true
6	10521	18	99.24	1786.32	44	46	2022-09-24	2022-09-24	09:15:00	15:30:00	true
7	10523	12	59.11	709.32	44	19	2022-12-11	2022-12-11	09:15:00	15:30:00	true
8	10525	12	73.5	882	37	141	2022-03-13	2022-03-13	09:15:00	15:30:00	true
9	10532	48	42.92	2060.16	44	140	2022-04-02	2022-04-02	09:15:00	15:30:00	true
10	10534	7	25.35	177.45	4	134	2022-03-12	2022-03-12	09:15:00	15:30:00	true

Successfully run. Total query runtime: 104 msec. 145 rows affected.

Total rows: 145 of 145 Query complete 00:00:00.104 Ln 6, Col 62