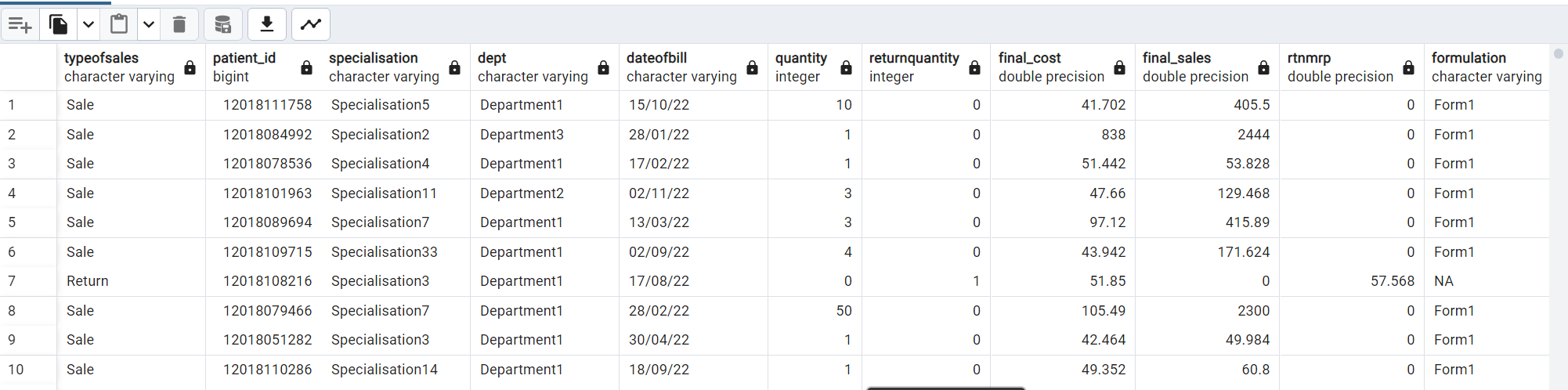
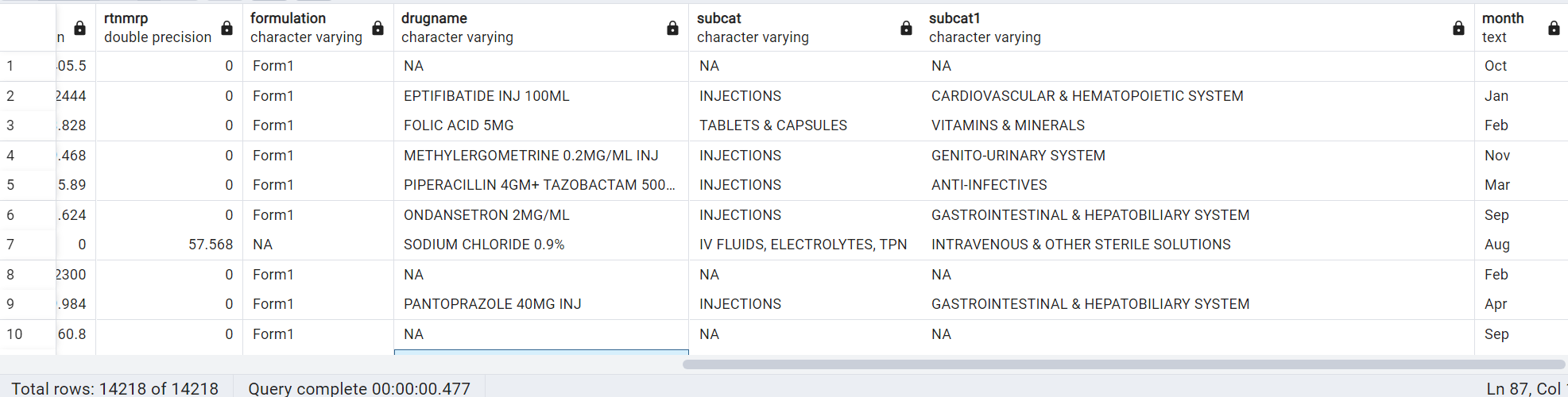
EDA Analysis:

Clean data view:

/\* EDA analysis \*/

select \* from MioView1





Finding Maximum and Minimum, average and count of quantity of sale and Maximum and minimum, average and count of returnquantity of return items out of sale

-- Descriptive statistics:

-- Maximum quantity, minimum quantity, maximum return quantity, minimum return quantity, avg of quantity and return quantity and total quantity sales and return quantity

select Max(quantity) as "max quantity",

MIN(quantity) as "min quantity",

AVG(quantity) as "avg quantity",

count(quantity) as "no of countity",

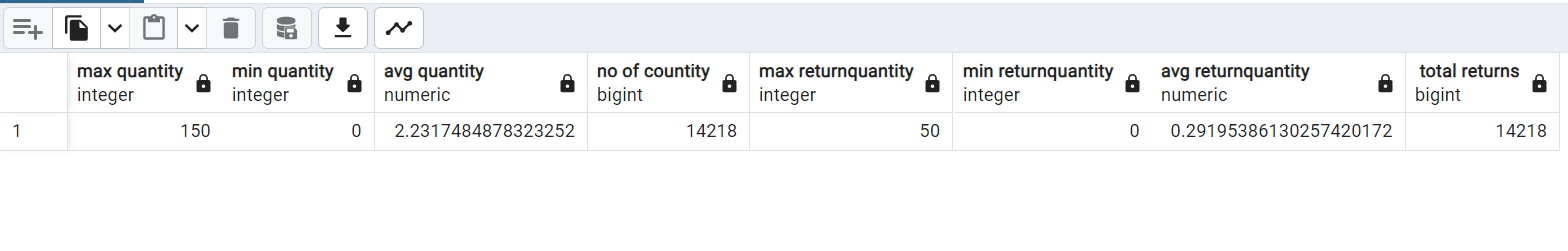
MAX(returnquantity) as "max returnquantity",

MIN(returnquantity) as "min returnquantity",

AVG(returnquantity) as "avg returnquantity",

count(returnquantity) as " total returns"

from MioView1



Frequency distribution of type of sales items

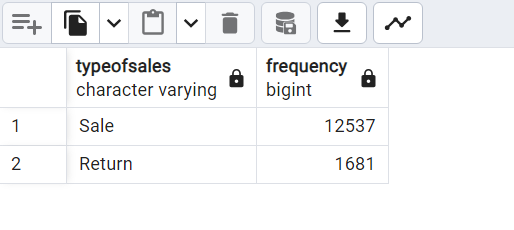
--frequency of sales

SELECT typeofsales, COUNT(\*) as frequency

FROM MioView1

GROUP BY typeofsales

ORDER BY frequency DESC;



Out of 14218 sales data 12537 are total sales and 1681 is total returns

First Moment Bussiness Decision:

First Moment Business Decision contains mean, median and mode values

Mean:

select round(avg(quantity)) as "average quantity",

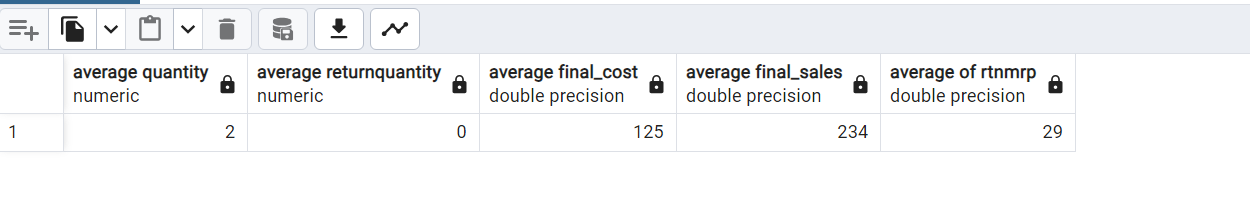
round(avg(returnquantity)) as "average returnquantity",

round(avg(final\_cost)) as "average final\_cost",

round(avg(final\_sales)) as "average final\_sales",

round(avg(rtnmrp)) as "average of rtnmrp"

from MioView1



Median:

SELECT QUANTITY\_MEDIAN AS QUANTITY\_MEDIAN\_VALUE,

RETURNQUANTITY\_MEDIAN AS RETURNQUANTITY\_MEDIAN\_VALUE,

FINAL\_COST\_MEDIAN AS FINAL\_COST\_MEDIAN\_VALUE,

FINAL\_SALES\_MEDIAN AS FINAL\_SALES\_MEDIAN\_VALUE,

RTNMRP\_MEDIAN AS RTNMRP\_MEDIAN\_VALUE

FROM

(

SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY quantity) AS median

FROM

MioView1) AS QUANTITY\_MEDIAN,

(SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY RETURNQUANTITY) AS MEDIAN

FROM

MioView1) AS RETURNQUANTITY\_MEDIAN,

(SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY final\_cost) AS MEDIAN

FROM

MioView1) AS FINAL\_COST\_MEDIAN,

(SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY final\_sales) AS MEDIAN

FROM

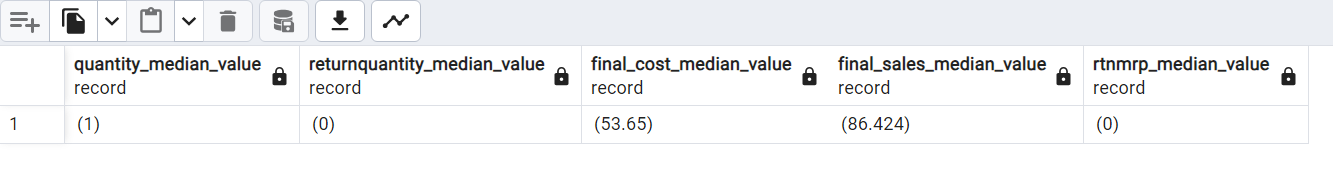
MioView1) AS FINAL\_SALES\_MEDIAN,

(SELECT

PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY rtnmrp) AS MEDIAN

FROM

MioView1) AS RTNMRP\_MEDIAN



Mode:

select quantity\_mode as quantity\_mode\_value,

return\_mode as returnquantity\_mode\_value,

finalcost\_mode as final\_cost\_mode\_value,

finalsales\_mode as final\_sales\_mode\_value,

rtnmrp\_mode as rtnmrp\_mode\_valuw

from

(SELECT quantity AS mode\_value, COUNT(\*) AS frequency

FROM MioView1

GROUP BY quantity

ORDER BY COUNT(\*) DESC

LIMIT 1) as quantity\_mode,

(SELECT returnquantity AS mode\_value, COUNT(\*) AS frequency

FROM MioView1

GROUP BY returnquantity

ORDER BY COUNT(\*) DESC

LIMIT 1) as return\_mode,

(SELECT final\_cost AS mode\_value, COUNT(\*) AS frequency

FROM MioView1

GROUP BY final\_cost

ORDER BY COUNT(\*) DESC

LIMIT 1) as finalcost\_mode,

(SELECT final\_sales AS mode\_value, COUNT(\*) AS frequency

FROM MioView1

GROUP BY final\_sales

ORDER BY COUNT(\*) DESC

LIMIT 1) as finalsales\_mode,

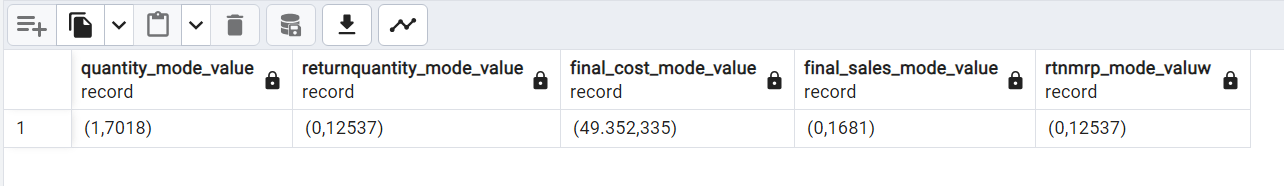
(SELECT rtnmrp AS mode\_value, COUNT(\*) AS frequency

FROM MioView1

GROUP BY rtnmrp

ORDER BY COUNT(\*) DESC

LIMIT 1) as rtnmrp\_mode



Second Moment Business Decision contains: Variance, standard deviation and range

Variance:

select variance(quantity::numeric) as variance\_quantity,

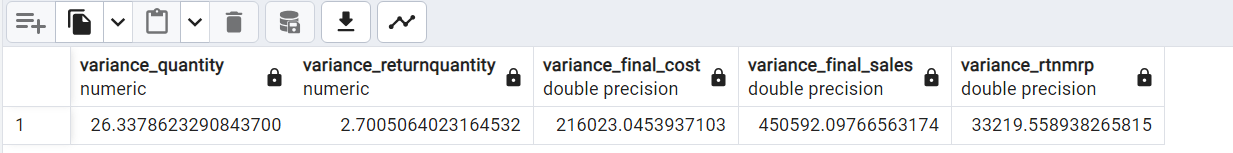
variance(returnquantity::numeric) as variance\_returnquantity,

variance(final\_cost) as variance\_final\_cost,

variance(final\_sales) as variance\_final\_sales,

variance(rtnmrp) as variance\_rtnmrp

from MioView1



Standard Deviation:

-- standard Deviation:

select stddev(quantity::numeric) as stdev\_quantity,

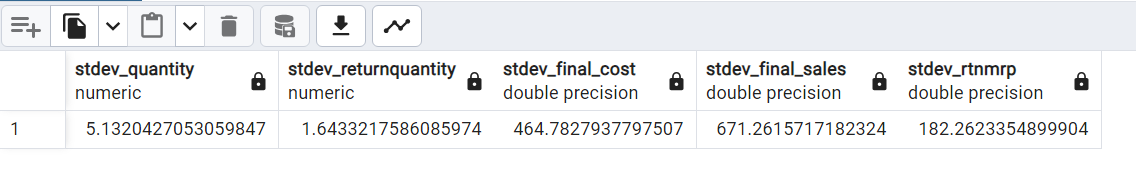
stddev(returnquantity::numeric) as stdev\_returnquantity,

stddev(final\_cost) as stdev\_final\_cost,

stddev(final\_sales) as stdev\_final\_sales,

stddev(rtnmrp) as stdev\_rtnmrp

from MioView1



Range:

select

max(quantity) - min(quantity) as range\_of\_quantity,

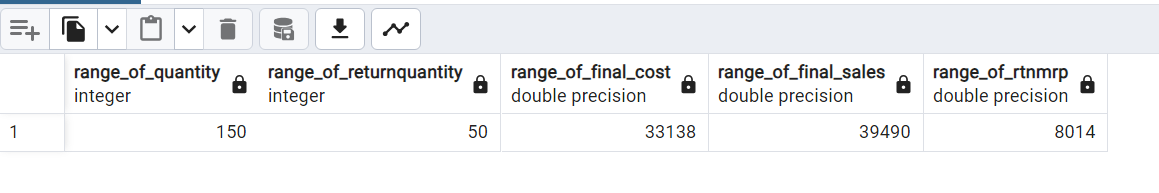
max(returnquantity) - min(returnquantity) as range\_of\_returnquantity,

max(final\_cost) - min(final\_cost) as range\_of\_final\_cost,

max(final\_sales) - min(final\_sales) as range\_of\_final\_sales,

max(rtnmrp) - min(rtnmrp) as range\_of\_rtnmrp

from MioView1;



Third Movement Business Decision:

Skewness:

Skewness is a measure of the asymmetry of a distribution

-- Calculate skewness for a numerical column

WITH moments1 AS (

SELECT

AVG(quantity) AS mean\_quantity,

STDDEV(quantity) AS stddev\_quantity,

AVG(returnquantity) as mean\_returnquantity,

STDDEV(returnquantity) as stddev\_returnquantity,

AVG(final\_cost) as mean\_final\_cost,

STDDEV(final\_cost) as stddev\_final\_cost,

AVG(final\_sales) as mean\_final\_sales,

STDDEV(final\_cost) as stddev\_final\_sales,

AVG(rtnmrp) as mean\_rtnmrp,

STDDEV(rtnmrp) as stddev\_rtnmrp

FROM MioView1

)

SELECT

--quantity as quantity,

SUM(POWER(quantity - mean\_quantity, 3)) / (COUNT(\*) \* POWER(stddev\_quantity, 3)) AS skewness\_for\_quantity,

SUM(POWER(returnquantity - mean\_returnquantity, 3)) / (COUNT(\*) \* POWER(stddev\_returnquantity, 3)) AS skewness\_for\_returnquantity,

SUM(POWER(final\_cost - mean\_final\_cost, 3)) / (COUNT(\*) \* POWER(stddev\_final\_cost, 3)) AS skewness\_for\_final\_cost,

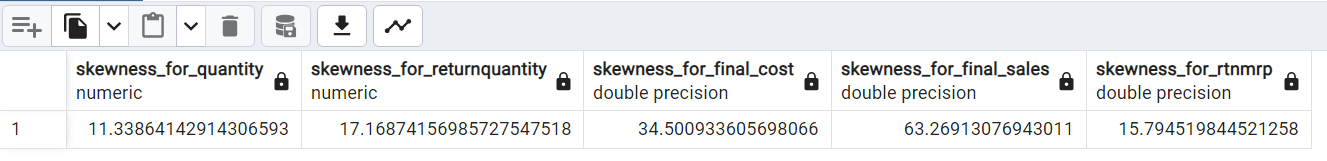
SUM(POWER(final\_sales - mean\_final\_sales, 3)) / (COUNT(\*) \* POWER(stddev\_final\_sales, 3)) AS skewness\_for\_final\_sales,

SUM(POWER(rtnmrp - mean\_rtnmrp, 3)) / (COUNT(\*) \* POWER(stddev\_rtnmrp, 3)) AS skewness\_for\_rtnmrp

FROM

MioView1, moments1

GROUP BY stddev\_quantity,stddev\_returnquantity,stddev\_final\_cost,stddev\_final\_sales,stddev\_rtnmrp;



Kurtosis: Kurtosis is a statistical measure that describes the distribution of a dataset.

-- Kurtoris

WITH moments AS (

SELECT

AVG((quantity - (select avg(quantity)from MioView1))::NUMERIC ^ 4) AS fourth\_moment1,

AVG((quantity - (select avg(quantity)from MioView1))::NUMERIC ^ 2) AS second\_moment1,

AVG((returnquantity - (select avg(returnquantity)from MioView1))::NUMERIC ^ 4) AS fourth\_moment2,

AVG((returnquantity - (select avg(returnquantity)from MioView1))::NUMERIC ^ 2) AS second\_moment2,

AVG((final\_cost - (select avg(final\_cost)from MioView1))::NUMERIC ^ 4) AS fourth\_moment3,

AVG((final\_cost - (select avg(final\_cost)from MioView1))::NUMERIC ^ 2) AS second\_moment3,

AVG((final\_sales - (select avg(final\_sales)from MioView1))::NUMERIC ^ 4) AS fourth\_moment4,

AVG((final\_sales - (select avg(final\_sales)from MioView1))::NUMERIC ^ 2) AS second\_moment4,

AVG((rtnmrp - (select avg(rtnmrp)from MioView1))::NUMERIC ^ 4) AS fourth\_moment5,

AVG((rtnmrp - (select avg(rtnmrp)from MioView1))::NUMERIC ^ 2) AS second\_moment5

FROM

MioView1

)

SELECT

fourth\_moment1 / (second\_moment1 ^ 2) AS kurtosis\_for\_quantity,

fourth\_moment2 / (second\_moment2 ^ 2) AS kurtosis\_for\_returnquantity,

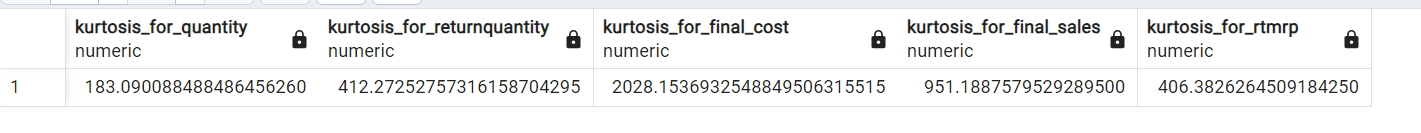
fourth\_moment3 / (second\_moment3 ^ 2) AS kurtosis\_for\_final\_cost,

fourth\_moment4 / (second\_moment4 ^ 2) AS kurtosis\_for\_final\_sales,

fourth\_moment5 / (second\_moment5 ^ 2) AS kurtosis\_for\_rtmrp

FROM

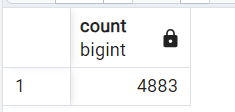
moments;



Bounce Rate Analysis:

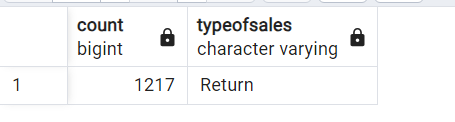
Q1) Total Distinct patients

select count(Distinct patient\_id) from MioView1



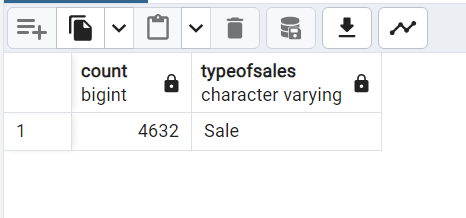
Q2) patient\_id count where type of sale is return

select count(Distinct patient\_id), typeofsales from MioView1 where typeofsales = 'Return' group by typeofsales



Q3) patient\_id count where type of sale is sale

select count(Distinct patient\_id), typeofsales from MioView where typeofsales = 'Sale' group by typeofsales



Q4) Overall Bounce Rate

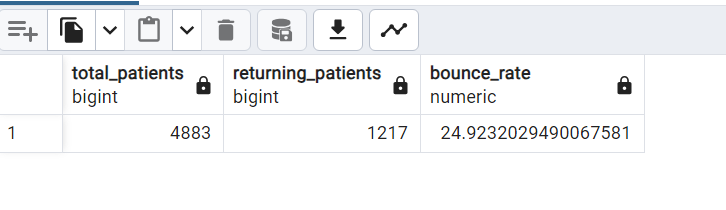
SELECT

COUNT(DISTINCT patient\_id) AS total\_patients,

COUNT(DISTINCT CASE WHEN returnquantity > 0 THEN patient\_id END) AS returning\_patients,

(COUNT(DISTINCT CASE WHEN returnquantity > 0 THEN patient\_id END) \* 100.0) / COUNT(DISTINCT patient\_id) AS bounce\_rate

FROM MioView1;



Q4) Bounce rate by specilization

select

specialisation,

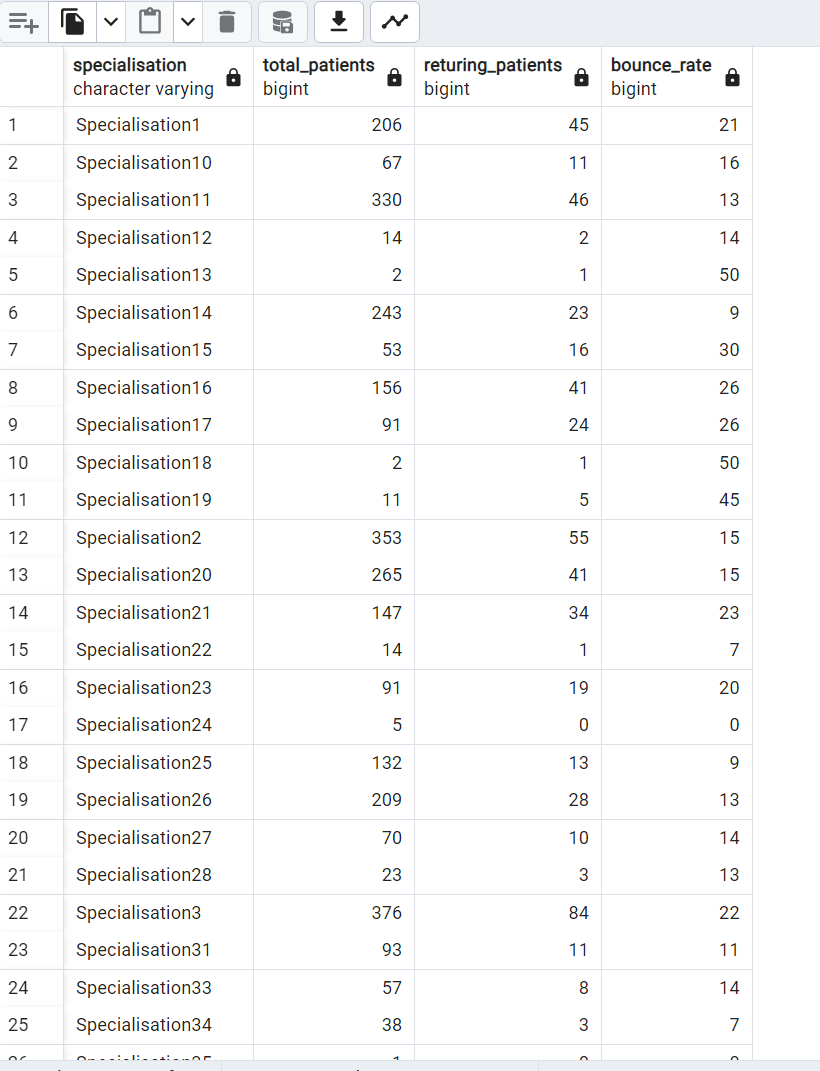
count(distinct patient\_id) as total\_patients,

count(Distinct case when returnquantity > 0 then patient\_id end) as returing\_patients,

(count(Distinct case when returnquantity > 0 then patient\_id end)\*100)/count(distinct patient\_id) as Bounce\_rate

from MioView1

group by specialisation



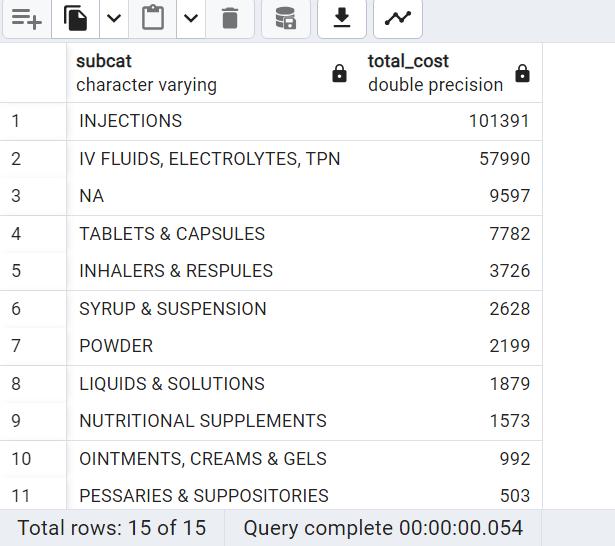
Q5) total cost of purchase that return from subcat

select subcat,round(sum(final\_cost)) as total\_cost

from MioView where typeofsales = 'Return'

group by subcat

order by total\_cost Desc



Insight: We can see Injections which hast highest final cost with 101391 followed by iv fluids, electronics, TPN with 57990. Returning this subcategories items will lead to cost burden to the medical inventory

Q6) count of drug returned without sales

select subcat,

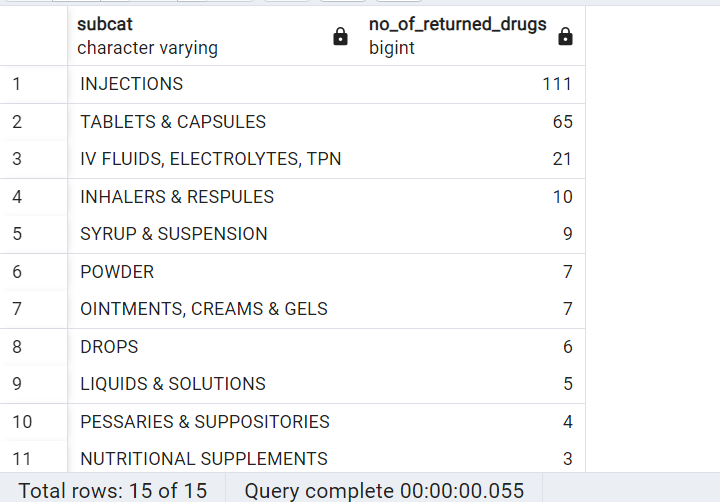
count(distinct drugname) as no\_of\_returned\_drugs

from MioView1

where typeofsales = 'Return' and final\_sales = 0

group by subcat

order by no\_of\_returned\_drugs desc



Insight: We can observe from above that subcategory Injection has the highest count of returns with 111 following by Tablets and capsules with 65, which can indicate that the customer disatification with this categories or other factors leading to return

Q7) Return items based on month

select month,count(\*) as return\_count

from MioView1

where typeofsales = 'Return'

group by month

order by return\_count Desc



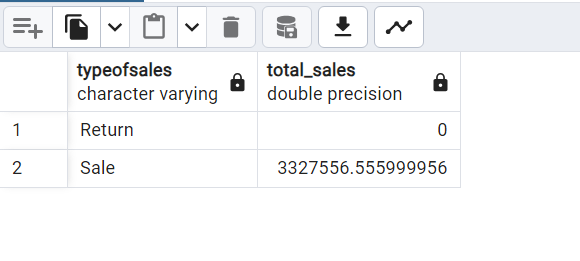
Q8) total sales when sales is return

select typeofsales,

sum(final\_sales) as total\_final\_sales

from MioView1

group by typeofsales;



Even though return was done it is not effecting the total sales

Q9) durgs which are mostly return

select drugname, count(\*) as return\_count

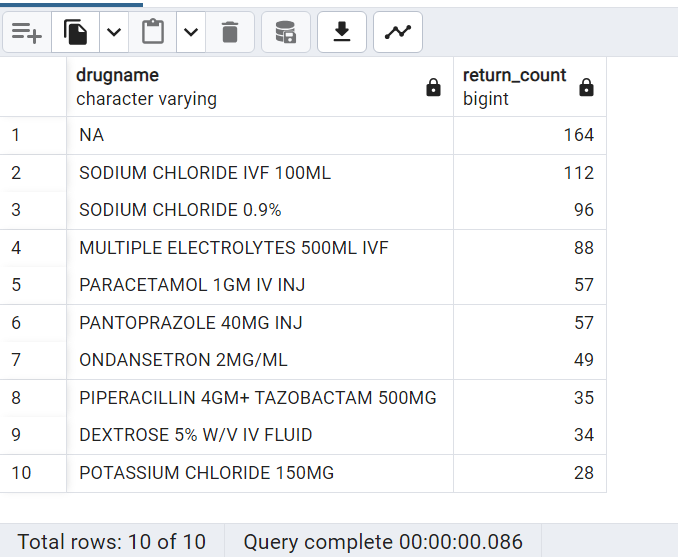
from MioView1

where returnquantity > 0

group by drugname

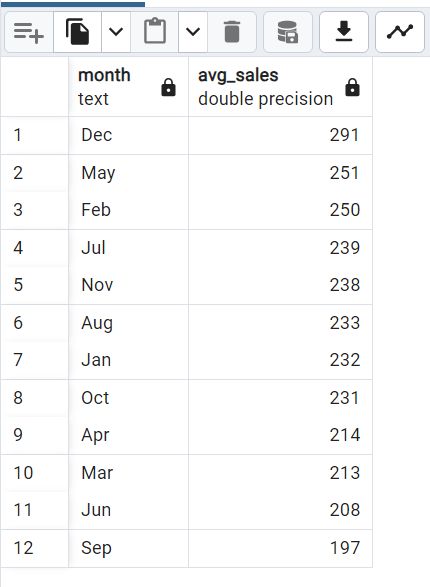
order by return\_count Desc

limit 10;



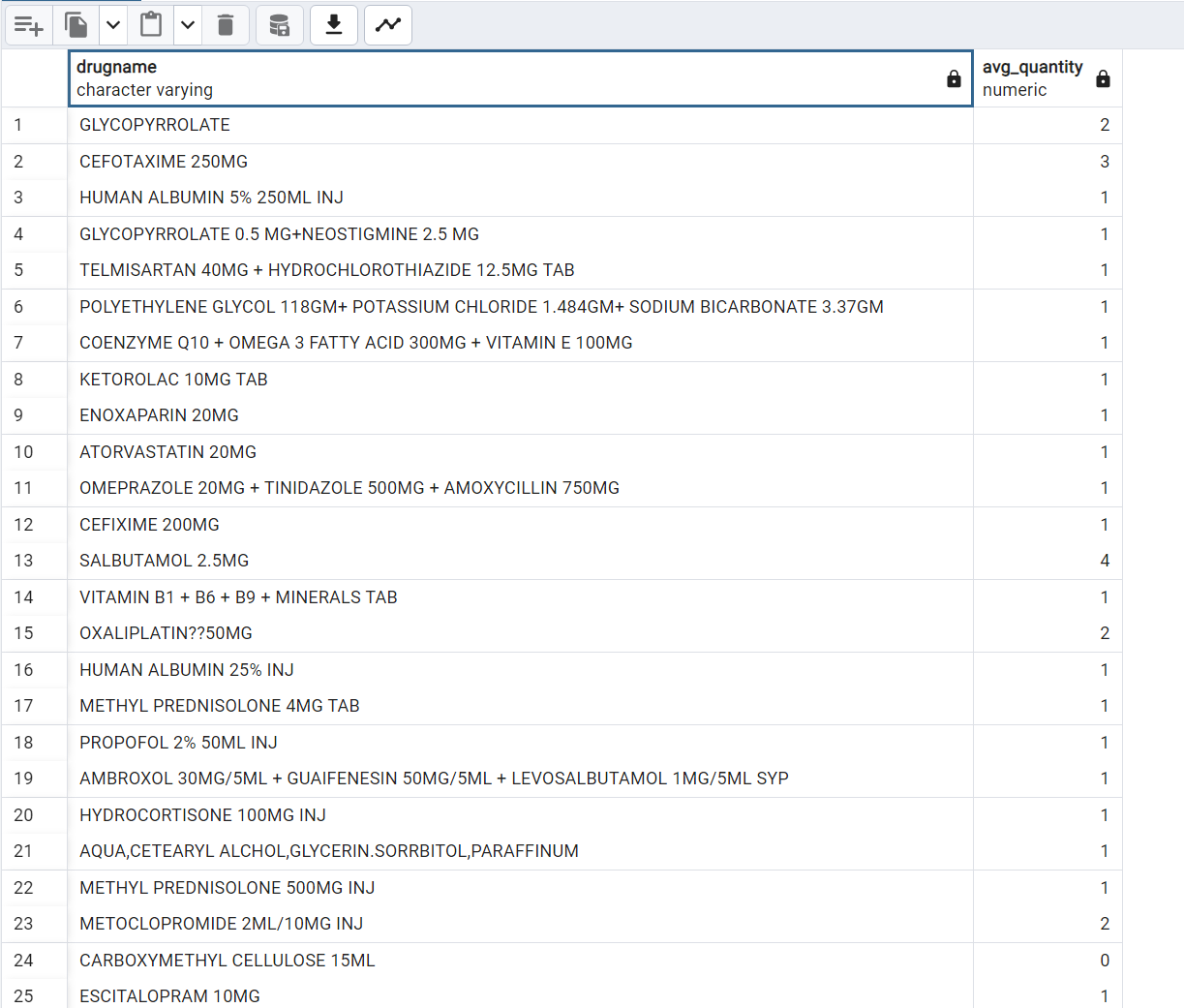
Q10) Total sales based on month

select month,round(avg(final\_sales)) as avg\_sales from MioView1 group by month order by avg\_sales Desc



Q11) Average quantity of durg purchases

select drugname,round(Avg(quantity)) as AVG\_quantity from MioView1 group by drugname

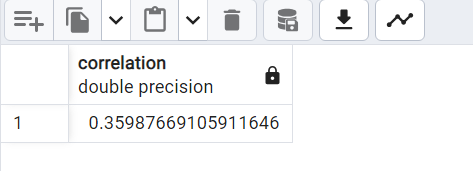


Q12) Relation between quantity and total sales

SELECT

CORR(quantity, final\_sales) AS correlation

FROM MioView1;



Q13) Average Sales based on Sepcialisation

SELECT

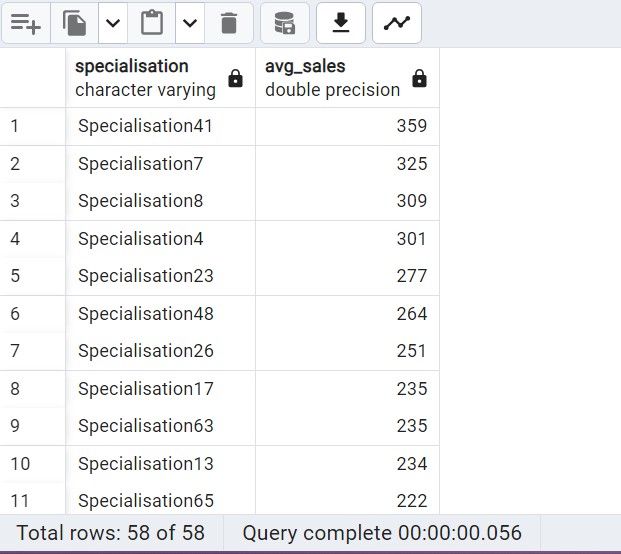
specialisation,

ROUND(AVG(final\_sales)) AS avg\_sales

FROM MioView

GROUP BY specialisation

ORDER BY avg\_sales DESC;



Q14) Frequency of return quantity

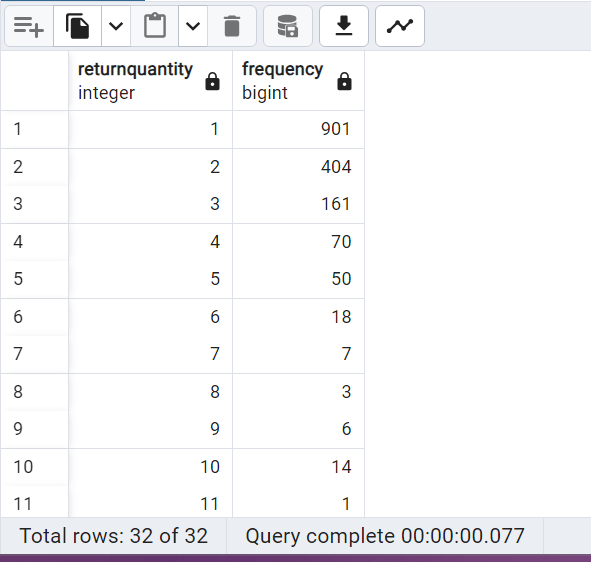
select returnquantity,count(\*) as frequency

from MioView1

where returnquantity > 0

group by returnquantity

order by returnquantity



901 times 1 item is returned after purchase followed by 401 times for 2 items. 50 items also returned at 3 times

Conclusion and insights:

From the above analysis we can see the sub categories Injections and tablets were returned frequently , so there may be some dissatisfaction with this products we need to check for those reasons for the items returned, the data we analyse does not consists that field so there is no correct finding for the reasons

We can these two categories injections, tablets, IV fluids return frequently and it costs lots of money, finding the reason for these returns we can reduce the amount of money lose to inventory

We can find highest average sales in December month and the highest returns in may month. Finding the reason we can increase the sales in other months.