**OPTIMIZATION OF MACHINE DOWNTIME**

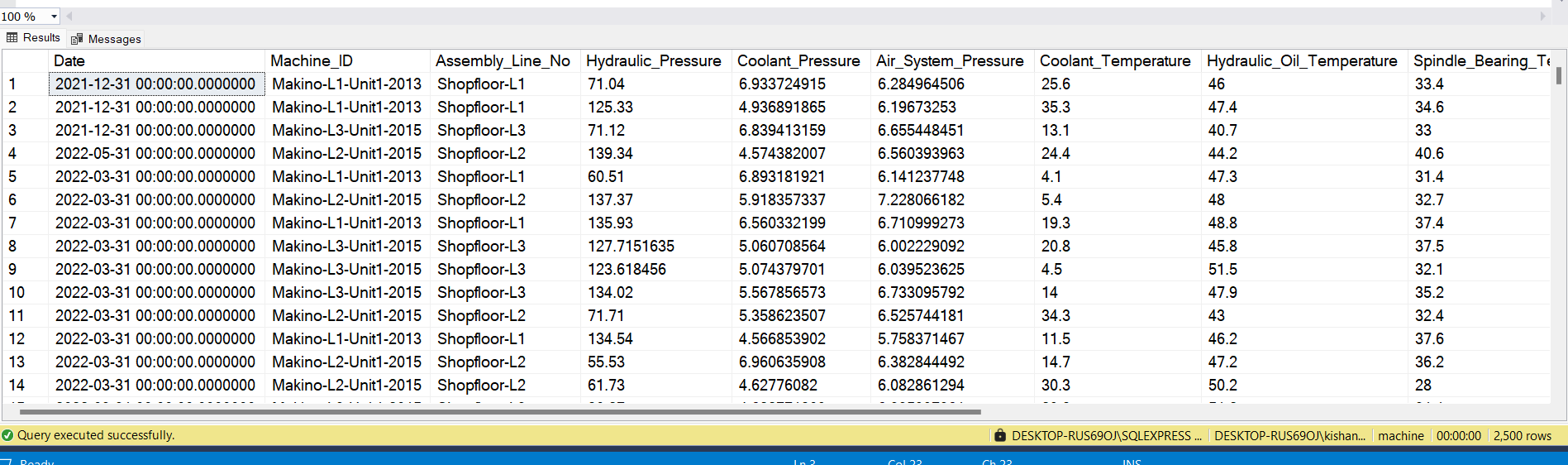
Pre-processing Code (SQL) By “ BALA HARSHITHA “

Software: MS-SQL Server

1. create database Project; -- creating the database
2. use Project; -- Set the current database to “Project”
3. Importing the table in CSV format into MS SQL Server

→ Right click on database(project) -> tasks -> Import flat file ->browse to select the CSV file “Data”

1. select \* from Machine; ----Display the “Machine” table

****

1. EXEC sp\_rename 'Machine', 'MachineDowntime'; --Changing the name of the table
2. To find mean

SELECT

AVG(Hydraulic\_Pressure) AS Mean\_Hydraulic\_Pressure,

AVG(Coolant\_Pressure) AS Mean\_Coolant\_Pressure,

AVG(Air\_System\_Pressure) AS Mean\_Air\_System\_Pressure,

AVG(Coolant\_Temperature) AS Mean\_Coolant\_Temperature,

AVG(Hydraulic\_Oil\_Temperature) AS Mean\_Hydraulic\_Oil\_Temperature,

AVG(Spindle\_Bearing\_Temperature) AS Mean\_Spindle\_Bearing\_Temperature,

AVG(Spindle\_Vibration) AS Mean\_Spindle\_Vibration,

AVG(Tool\_Vibration) AS Mean\_Tool\_Vibration,

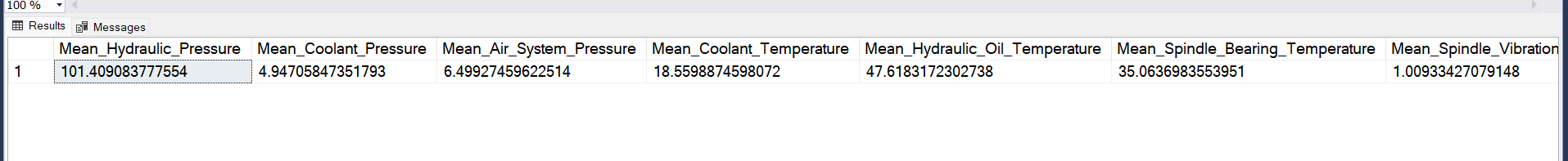
AVG(Spindle\_Speed) AS Mean\_Spindle\_Speed,

AVG(Voltage) AS Mean\_Voltage,

AVG(Torque) AS Mean\_Torque,

AVG(Cutting) AS Mean\_Cutting

FROM MachineDowntime;

****

1. To find Median

WITH CTE AS (

SELECT

Hydraulic\_Pressure,

Coolant\_Pressure,

Air\_System\_Pressure,

Coolant\_Temperature,

Hydraulic\_Oil\_Temperature,

Spindle\_Bearing\_Temperature,

Spindle\_Vibration,

Tool\_Vibration,

Spindle\_Speed,

Voltage,

Torque,

Cutting,

ROW\_NUMBER() OVER (ORDER BY Hydraulic\_Pressure) AS RowNum,

COUNT(\*) OVER () AS TotalCount

FROM MachineDowntime

)

SELECT

AVG(Hydraulic\_Pressure) AS Median\_Hydraulic\_Pressure,

AVG(Coolant\_Pressure) AS Median\_Coolant\_Pressure,

AVG(Air\_System\_Pressure) AS Median\_Air\_System\_Pressure,

AVG(Coolant\_Temperature) AS Median\_Coolant\_Temperature,

AVG(Hydraulic\_Oil\_Temperature) AS Median\_Hydraulic\_Oil\_Temperature,

AVG(Spindle\_Bearing\_Temperature) AS Median\_Spindle\_Bearing\_Temperature,

AVG(Spindle\_Vibration) AS Median\_Spindle\_Vibration,

AVG(Tool\_Vibration) AS Median\_Tool\_Vibration,

AVG(Spindle\_Speed) AS Median\_Spindle\_Speed,

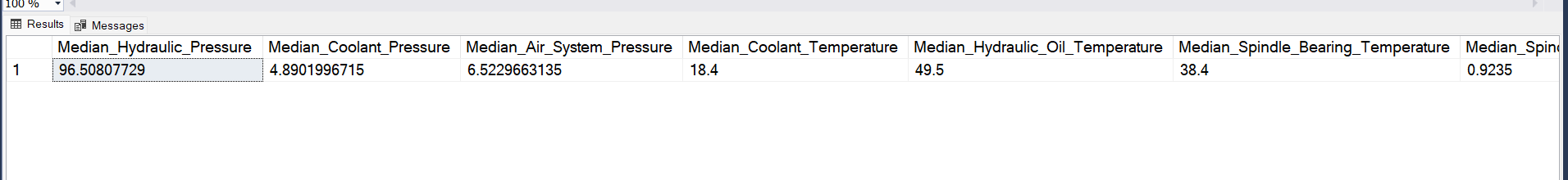
AVG(Voltage) AS Median\_Voltage,

AVG(Torque) AS Median\_Torque,

AVG(Cutting) AS Median\_Cutting

FROM CTE

WHERE RowNum IN ((TotalCount + 1) / 2, (TotalCount + 2) / 2);



1. To Find Mode

SELECT TOP 1

(SELECT TOP 1 Hydraulic\_Pressure

FROM MachineDowntime

GROUP BY Hydraulic\_Pressure

ORDER BY COUNT(\*) DESC) AS Mode\_Hydraulic\_Pressure,

(SELECT TOP 1 Coolant\_Pressure

FROM MachineDowntime

GROUP BY Coolant\_Pressure

ORDER BY COUNT(\*) DESC) AS Mode\_Coolant\_Pressure,

(SELECT TOP 1 Air\_System\_Pressure

FROM MachineDowntime

GROUP BY Air\_System\_Pressure

ORDER BY COUNT(\*) DESC) AS Mode\_Air\_System\_Pressure,

(SELECT TOP 1 Coolant\_Temperature

FROM MachineDowntime

GROUP BY Coolant\_Temperature

ORDER BY COUNT(\*) DESC) AS Mode\_Coolant\_Temperature,

(SELECT TOP 1 Hydraulic\_Oil\_Temperature

FROM MachineDowntime

GROUP BY Hydraulic\_Oil\_Temperature

ORDER BY COUNT(\*) DESC) AS Mode\_Hydraulic\_Oil\_Temperature,

(SELECT TOP 1 Spindle\_Bearing\_Temperature

FROM MachineDowntime

GROUP BY Spindle\_Bearing\_Temperature

ORDER BY COUNT(\*) DESC) AS Mode\_Spindle\_Bearing\_Temperature,

(SELECT TOP 1 Spindle\_Vibration

FROM MachineDowntime

GROUP BY Spindle\_Vibration

ORDER BY COUNT(\*) DESC) AS Mode\_Spindle\_Vibration,

(SELECT TOP 1 Tool\_Vibration

FROM MachineDowntime

GROUP BY Tool\_Vibration

ORDER BY COUNT(\*) DESC) AS Mode\_Tool\_Vibration,

(SELECT TOP 1 Spindle\_Speed

FROM MachineDowntime

GROUP BY Spindle\_Speed

ORDER BY COUNT(\*) DESC) AS Mode\_Spindle\_Speed,

(SELECT TOP 1 Voltage

FROM MachineDowntime

GROUP BY Voltage

ORDER BY COUNT(\*) DESC) AS Mode\_Voltage,

(SELECT TOP 1 Torque

FROM MachineDowntime

GROUP BY Torque

ORDER BY COUNT(\*) DESC) AS Mode\_Torque,

(SELECT TOP 1 Cutting

FROM MachineDowntime

GROUP BY Cutting

ORDER BY COUNT(\*) DESC) AS Mode\_Cutting,

(SELECT TOP 1 Downtime

FROM MachineDowntime

GROUP BY Downtime

ORDER BY COUNT(\*) DESC) AS Mode\_Downtime

FROM MachineDowntime;



1. To Find Variance

SELECT

ROUND(VAR(Hydraulic\_Pressure),2) AS Hydraulic\_Pressure\_var,

ROUND(VAR(Coolant\_Pressure),2) AS Coolant\_Pressure\_var,

ROUND(VAR(Air\_System\_Pressure),2) AS Air\_System\_Pressure\_var,

ROUND(VAR(Coolant\_Temperature),2) AS Coolant\_Temperature\_var,

ROUND(VAR(Hydraulic\_Oil\_Temperature),2) AS Hydraulic\_Oil\_Temperature\_var,

ROUND(VAR(Spindle\_Bearing\_Temperature),2) AS Spindle\_Bearing\_Temperature\_var,

ROUND(VAR(Spindle\_Vibration),2) AS Spindle\_Vibration\_var,

ROUND(VAR(Tool\_Vibration),2) AS Tool\_Vibration\_var,

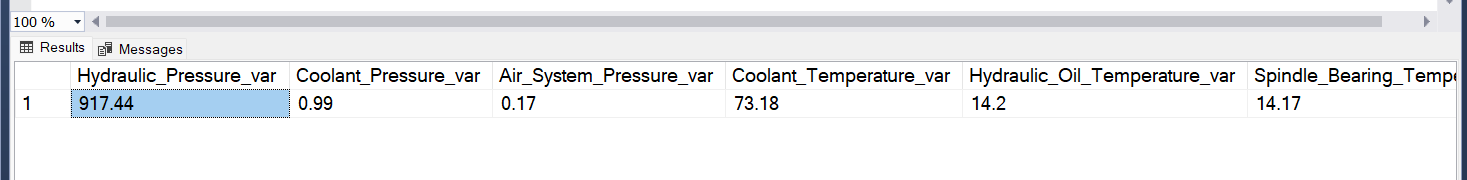
ROUND(VAR(Spindle\_Speed),2) AS Spindle\_Speed\_var,

ROUND(VAR(Voltage),2) AS Voltage\_var,

ROUND(VAR(Torque),2) AS Torque\_var,

ROUND(VAR(Cutting),2) AS Cutting\_var

from MachineDowntime;



1. To find STANDARD DEVIATION

SELECT

ROUND(STDEV(Hydraulic\_Pressure),2) AS Hydraulic\_Pressure\_STDEV,

ROUND(STDEV(Coolant\_Pressure),2) AS Coolant\_Pressure\_STDEV,

ROUND(STDEV(Air\_System\_Pressure),2) AS Air\_System\_Pressure\_STDEV,

ROUND(STDEV(Coolant\_Temperature),2) AS Coolant\_Temperature\_STDEV,

ROUND(STDEV(Hydraulic\_Oil\_Temperature),2) AS Hydraulic\_Oil\_Temperature\_STDEV,

ROUND(STDEV(Spindle\_Bearing\_Temperature),2) AS Spindle\_Bearing\_Temperature\_STDEV,

ROUND(STDEV(Spindle\_Vibration),2) AS Spindle\_Vibration\_STDEV,

ROUND(STDEV(Tool\_Vibration),2) AS Tool\_Vibration\_STDEV,

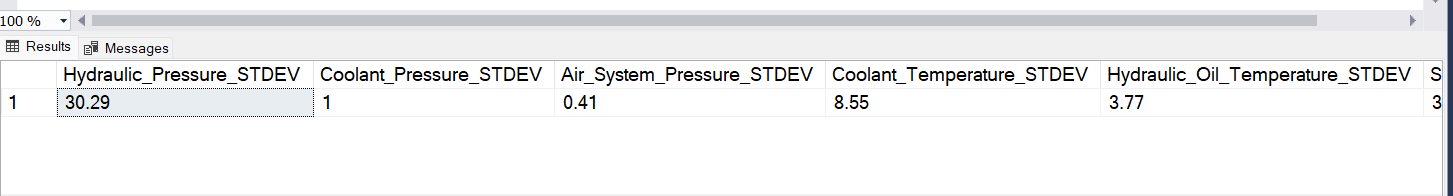
ROUND(STDEV(Spindle\_Speed),2) AS Spindle\_Speed\_STDEV,

ROUND(STDEV(Voltage),2) AS Voltage\_STDEV,

ROUND(STDEV(Torque),2) AS Torque\_STDEV,

ROUND(STDEV(Cutting),2) AS Cutting\_STDEV

from MachineDowntime;



1. To find Skewness of Hydraulic\_Pressure, Cutting and Torque

-- Calculate the mean

DECLARE @Mean FLOAT;

SELECT @Mean = AVG(Hydraulic\_Pressure) FROM MachineDowntime;

-- Calculate the standard deviation

DECLARE @StdDev FLOAT;

SELECT @StdDev = STDEV(Hydraulic\_Pressure) FROM MachineDowntime;

-- Calculate the count

DECLARE @Count INT;

SELECT @Count = COUNT(Hydraulic\_Pressure) FROM MachineDowntime;

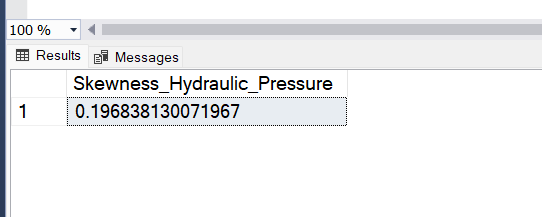
-- Calculate skewness

DECLARE @Skewness FLOAT;

SELECT @Skewness = SUM(POWER(Hydraulic\_Pressure - @Mean, 3)) / @Count / POWER(@StdDev, 3) FROM MachineDowntime;

-- Output the skewness

SELECT @Skewness AS Skewness\_Hydraulic\_Pressure;



-- Calculate the mean

DECLARE @Mean FLOAT;

SELECT @Mean = AVG(Cutting) FROM MachineDowntime;

-- Calculate the standard deviation

DECLARE @StdDev FLOAT;

SELECT @StdDev = STDEV(Cutting) FROM MachineDowntime;

-- Calculate the count

DECLARE @Count INT;

SELECT @Count = COUNT(Cutting) FROM MachineDowntime;

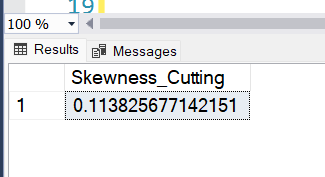
-- Calculate skewness

DECLARE @Skewness FLOAT;

SELECT @Skewness = SUM(POWER(Cutting - @Mean, 3)) / @Count / POWER(@StdDev, 3) FROM MachineDowntime;

-- Output the skewness

SELECT @Skewness AS Skewness\_Cutting;



-- Calculate the mean

DECLARE @Mean FLOAT;

SELECT @Mean = AVG(Torque) FROM MachineDowntime;

-- Calculate the standard deviation

DECLARE @StdDev FLOAT;

SELECT @StdDev = STDEV(Torque) FROM MachineDowntime;

-- Calculate the count

DECLARE @Count INT;

SELECT @Count = COUNT(Torque) FROM MachineDowntime;

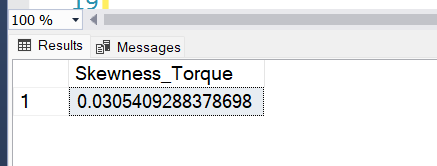
-- Calculate skewness

DECLARE @Skewness FLOAT;

SELECT @Skewness = SUM(POWER(Torque - @Mean, 3)) / @Count / POWER(@StdDev, 3) FROM MachineDowntime;

-- Output the skewness

SELECT @Skewness AS Skewness\_Torque;



1. To find Kurtosis of Hydraulic\_Pressure, Cutting and Torque

WITH Stats AS (

SELECT

AVG(Hydraulic\_Pressure) AS mean,

STDEV(Hydraulic\_Pressure) AS stddev,

COUNT(Hydraulic\_Pressure) AS n

FROM

MachineDowntime

),

Moments AS (

SELECT

POWER(Hydraulic\_Pressure - Stats.mean, 2) AS squared\_diff,

POWER(Hydraulic\_Pressure - Stats.mean, 4) AS fourth\_diff

FROM

MachineDowntime, Stats

),

AggregatedMoments AS (

SELECT

SUM(squared\_diff) AS sum\_squared\_diff,

SUM(fourth\_diff) AS sum\_fourth\_diff,

COUNT(Hydraulic\_Pressure) AS n

FROM

Moments, Stats,MachineDowntime

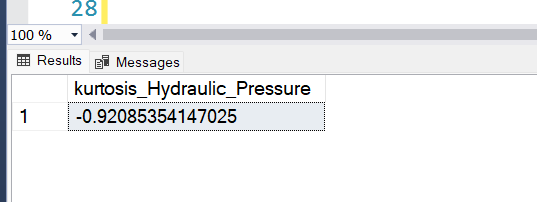
)

SELECT

(n \* sum\_fourth\_diff) / (sum\_squared\_diff \* sum\_squared\_diff) - 3 AS kurtosis\_Hydraulic\_Pressure

FROM

AggregatedMoments;



WITH Stats AS (

SELECT

AVG(Cutting) AS mean,

STDEV(Cutting) AS stddev,

COUNT(Cutting) AS n

FROM

MachineDowntime

),

Moments AS (

SELECT

POWER(Cutting - Stats.mean, 2) AS squared\_diff,

POWER(Cutting - Stats.mean, 4) AS fourth\_diff

FROM

MachineDowntime, Stats

),

AggregatedMoments AS (

SELECT

SUM(squared\_diff) AS sum\_squared\_diff,

SUM(fourth\_diff) AS sum\_fourth\_diff,

COUNT(Cutting) AS n

FROM

Moments, Stats,MachineDowntime

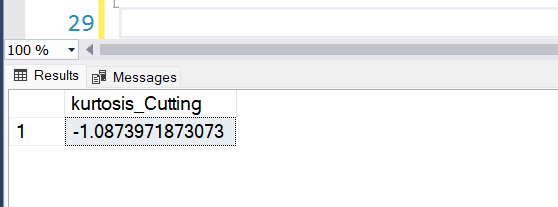
)

SELECT

(n \* sum\_fourth\_diff) / (sum\_squared\_diff \* sum\_squared\_diff) - 3 AS kurtosis\_Cutting

FROM

AggregatedMoments;



WITH Stats AS (

SELECT

AVG(Torque) AS mean,

STDEV(Torque) AS stddev,

COUNT(Torque) AS n

FROM

MachineDowntime

),

Moments AS (

SELECT

POWER(Torque - Stats.mean, 2) AS squared\_diff,

POWER(Torque - Stats.mean, 4) AS fourth\_diff

FROM

MachineDowntime, Stats

),

AggregatedMoments AS (

SELECT

SUM(squared\_diff) AS sum\_squared\_diff,

SUM(fourth\_diff) AS sum\_fourth\_diff,

COUNT(Torque) AS n

FROM

Moments, Stats,MachineDowntime

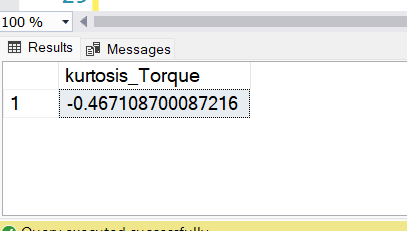
)

SELECT

(n \* sum\_fourth\_diff) / (sum\_squared\_diff \* sum\_squared\_diff) - 3 AS kurtosis\_Torque

FROM

AggregatedMoments;



1. Find Null Values

SELECT

SUM(CASE WHEN Date IS NULL THEN 1 ELSE 0 END) AS Null\_Date,

SUM(CASE WHEN Machine\_ID IS NULL THEN 1 ELSE 0 END) AS Null\_Machine\_ID,

SUM(CASE WHEN Assembly\_Line\_No IS NULL THEN 1 ELSE 0 END) AS Null\_Assembly\_Line\_No,

SUM(CASE WHEN Hydraulic\_Pressure IS NULL THEN 1 ELSE 0 END) AS Null\_Hydraulic\_Pressure,

SUM(CASE WHEN Coolant\_Pressure IS NULL THEN 1 ELSE 0 END) AS Null\_Coolant\_Pressure,

SUM(CASE WHEN Air\_System\_Pressure IS NULL THEN 1 ELSE 0 END) AS Null\_Air\_System\_Pressure,

SUM(CASE WHEN Coolant\_Temperature IS NULL THEN 1 ELSE 0 END) AS Null\_Coolant\_Temperature,

SUM(CASE WHEN Hydraulic\_Oil\_Temperature IS NULL THEN 1 ELSE 0 END) AS Null\_Hydraulic\_Oil\_Temperature,

SUM(CASE WHEN Spindle\_Bearing\_Temperature IS NULL THEN 1 ELSE 0 END) AS Null\_Spindle\_Bearing\_Temperature,

SUM(CASE WHEN Spindle\_Vibration IS NULL THEN 1 ELSE 0 END) AS Null\_Spindle\_Vibration,

SUM(CASE WHEN Tool\_Vibration IS NULL THEN 1 ELSE 0 END) AS Null\_Tool\_Vibration,

SUM(CASE WHEN Spindle\_Speed IS NULL THEN 1 ELSE 0 END) AS Null\_Spindle\_Speed,

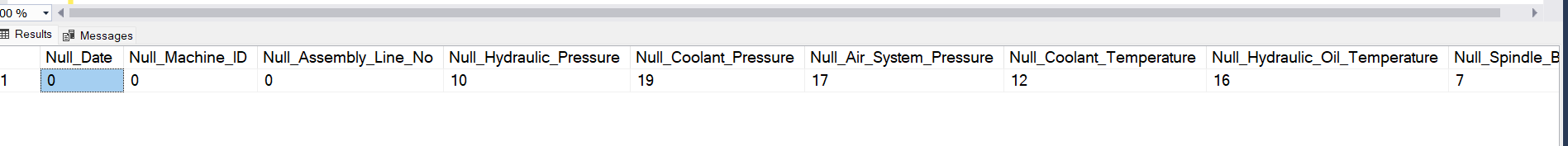
SUM(CASE WHEN Voltage IS NULL THEN 1 ELSE 0 END) AS Null\_Voltage,

SUM(CASE WHEN Torque IS NULL THEN 1 ELSE 0 END) AS Null\_Torque,

SUM(CASE WHEN Cutting IS NULL THEN 1 ELSE 0 END) AS Null\_Cutting,

SUM(CASE WHEN Downtime IS NULL THEN 1 ELSE 0 END) AS Null\_Downtime

FROM MachineDowntime;



1. Remove Null Values

DELETE FROM MachineDowntime

WHERE

Date IS NULL OR

Machine\_ID IS NULL OR

Assembly\_Line\_No IS NULL OR

Hydraulic\_Pressure IS NULL OR

Coolant\_Pressure IS NULL OR

Air\_System\_Pressure IS NULL OR

Coolant\_Temperature IS NULL OR

Hydraulic\_Oil\_Temperature IS NULL OR

Spindle\_Bearing\_Temperature IS NULL OR

Spindle\_Vibration IS NULL OR

Tool\_Vibration IS NULL OR

Spindle\_Speed IS NULL OR

Voltage IS NULL OR

Torque IS NULL OR

Cutting IS NULL OR

Downtime IS NULL;

1. Remove Outliers

WITH Q\_Hydraulic\_Pressure AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Hydraulic\_Pressure) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Hydraulic\_Pressure) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Hydraulic\_Pressure AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Hydraulic\_Pressure

),

Q\_Coolant\_Pressure AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Coolant\_Pressure) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Coolant\_Pressure) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Coolant\_Pressure AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Coolant\_Pressure

),

Q\_Air\_System\_Pressure AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Air\_System\_Pressure) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Air\_System\_Pressure) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Air\_System\_Pressure AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Air\_System\_Pressure

),

Q\_Coolant\_Temperature AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Coolant\_Temperature) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Coolant\_Temperature) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Coolant\_Temperature AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Coolant\_Temperature

),

Q\_Hydraulic\_Oil\_Temperature AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Hydraulic\_Oil\_Temperature) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Hydraulic\_Oil\_Temperature) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Hydraulic\_Oil\_Temperature AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Hydraulic\_Oil\_Temperature

),

Q\_Spindle\_Bearing\_Temperature AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Spindle\_Bearing\_Temperature) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Spindle\_Bearing\_Temperature) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Spindle\_Bearing\_Temperature AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Spindle\_Bearing\_Temperature

),

Q\_Spindle\_Vibration AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Spindle\_Vibration) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Spindle\_Vibration) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Spindle\_Vibration AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Spindle\_Vibration

),

Q\_Tool\_Vibration AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Tool\_Vibration) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Tool\_Vibration) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Tool\_Vibration AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Tool\_Vibration

),

Q\_Spindle\_Speed AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Spindle\_Speed) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Spindle\_Speed) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Spindle\_Speed AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Spindle\_Speed

),

Q\_Voltage AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Voltage) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Voltage) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Voltage AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Voltage

),

Q\_Torque AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Torque) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Torque) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Torque AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Torque

),

Q\_Cutting AS (

SELECT

PERCENTILE\_CONT(0.25) WITHIN GROUP (ORDER BY Cutting) OVER () AS Q1,

PERCENTILE\_CONT(0.75) WITHIN GROUP (ORDER BY Cutting) OVER () AS Q3

FROM MachineDowntime

),

IQR\_Cutting AS (

SELECT

MAX(Q1) AS Q1,

MAX(Q3) AS Q3,

(MAX(Q3) - MAX(Q1)) \* 1.5 AS IQR

FROM Q\_Cutting

)

SELECT \* INTO data\_cleaned

FROM MachineDowntime

WHERE

Hydraulic\_Pressure BETWEEN

(SELECT Q1 - IQR FROM IQR\_Hydraulic\_Pressure) AND

(SELECT Q3 + IQR FROM IQR\_Hydraulic\_Pressure) AND

Coolant\_Pressure BETWEEN

(SELECT Q1 - IQR FROM IQR\_Coolant\_Pressure) AND

(SELECT Q3 + IQR FROM IQR\_Coolant\_Pressure) AND

Air\_System\_Pressure BETWEEN

(SELECT Q1 - IQR FROM IQR\_Air\_System\_Pressure) AND

(SELECT Q3 + IQR FROM IQR\_Air\_System\_Pressure) AND

Coolant\_Temperature BETWEEN

(SELECT Q1 - IQR FROM IQR\_Coolant\_Temperature) AND

(SELECT Q3 + IQR FROM IQR\_Coolant\_Temperature) AND

Hydraulic\_Oil\_Temperature BETWEEN

(SELECT Q1 - IQR FROM IQR\_Hydraulic\_Oil\_Temperature) AND

(SELECT Q3 + IQR FROM IQR\_Hydraulic\_Oil\_Temperature) AND

Spindle\_Bearing\_Temperature BETWEEN

(SELECT Q1 - IQR FROM IQR\_Spindle\_Bearing\_Temperature) AND

(SELECT Q3 + IQR FROM IQR\_Spindle\_Bearing\_Temperature) AND

Spindle\_Vibration BETWEEN

(SELECT Q1 - IQR FROM IQR\_Spindle\_Vibration) AND

(SELECT Q3 + IQR FROM IQR\_Spindle\_Vibration) AND

Tool\_Vibration BETWEEN

(SELECT Q1 - IQR FROM IQR\_Tool\_Vibration) AND

(SELECT Q3 + IQR FROM IQR\_Tool\_Vibration) AND

Spindle\_Speed BETWEEN

(SELECT Q1 - IQR FROM IQR\_Spindle\_Speed) AND

(SELECT Q3 + IQR FROM IQR\_Spindle\_Speed) AND

Voltage BETWEEN

(SELECT Q1 - IQR FROM IQR\_Voltage) AND

(SELECT Q3 + IQR FROM IQR\_Voltage) AND

Torque BETWEEN

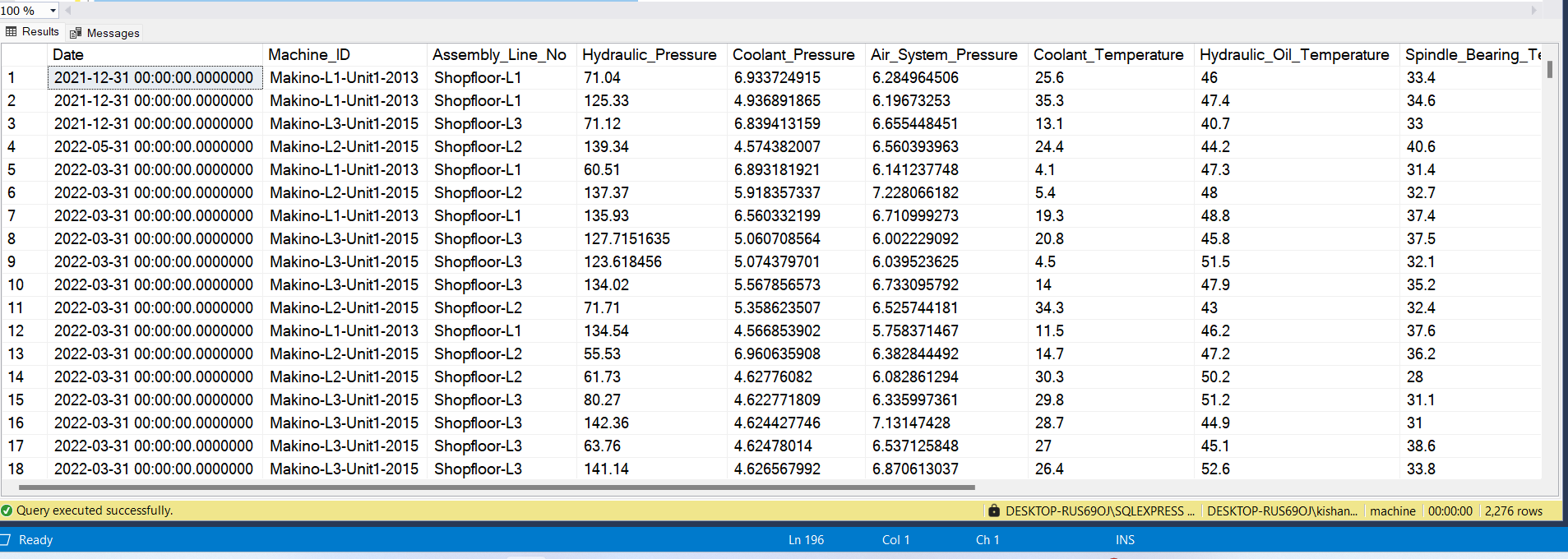
(SELECT Q1 - IQR FROM IQR\_Torque) AND

(SELECT Q3 + IQR FROM IQR\_Torque) AND

Cutting BETWEEN

(SELECT Q1 - IQR FROM IQR\_Cutting) AND

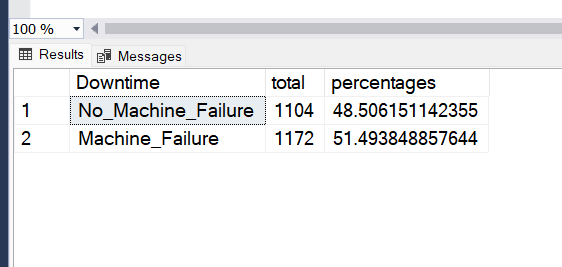
(SELECT Q3 + IQR FROM IQR\_Cutting);



**EDA FROM CLEANED DATASET**

1. Proportion of machine downtime occurrences

select Downtime, count(\*) AS total, (COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM data\_cleaned)) AS percentages from data\_cleaned group by Downtime;



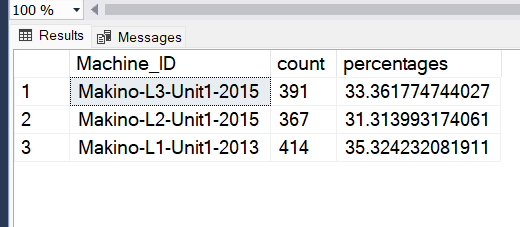
1. Downtime Occurrences by Machine ID

SELECT Machine\_ID, COUNT(\*) AS count,(COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM data\_cleaned where Downtime='Machine\_Failure')) AS percentages

FROM data\_cleaned

WHERE Downtime='Machine\_Failure'

GROUP BY Machine\_ID;

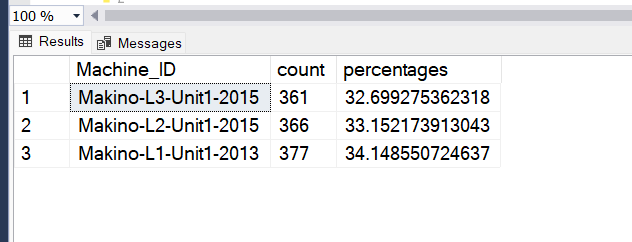


SELECT Machine\_ID, COUNT(\*) AS count,(COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM data\_cleaned where Downtime='No\_Machine\_Failure')) AS percentages

FROM data\_cleaned

WHERE Downtime='No\_Machine\_Failure'

GROUP BY Machine\_ID;



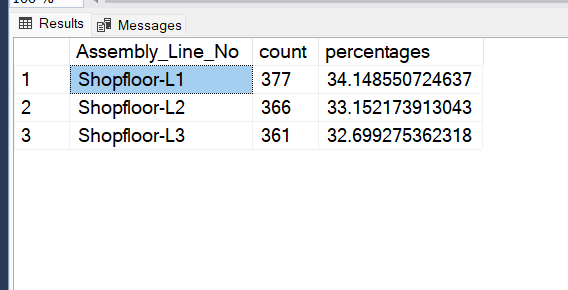
1. Downtime Occurrences by Assembly\_Line\_No

SELECT Assembly\_Line\_No, COUNT(\*) AS count,(COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM data\_cleaned where Downtime='No\_Machine\_Failure')) AS percentages

FROM data\_cleaned

WHERE Downtime='No\_Machine\_Failure'

GROUP BY Assembly\_Line\_No;

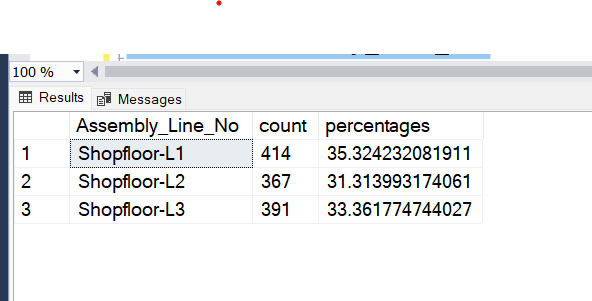


SELECT Assembly\_Line\_No, COUNT(\*) AS count,(COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM data\_cleaned where Downtime='Machine\_Failure')) AS percentages

FROM data\_cleaned

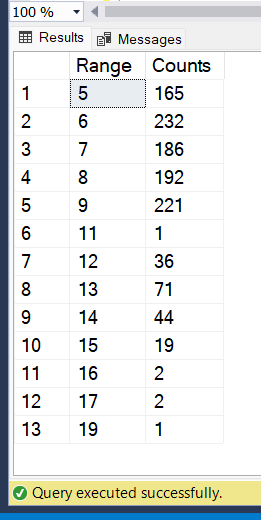
WHERE Downtime='Machine\_Failure'

GROUP BY Assembly\_Line\_No;



1. Hydraulic\_Pressure histogram Plot

select cast (Hydraulic\_Pressure/10 as int) AS Range,count(\*) AS Counts FROM data\_cleaned where Downtime='Machine\_Failure' GROUP BY cast (Hydraulic\_Pressure/10 as int) order by cast (Hydraulic\_Pressure/10 as int) ASC;



Insights for Machine Downtime :

Data cleaning slightly reduced the mean in columns like Hydraulic Pressure (bar) and Coolant Temperature (°C), indicating the removal of high outliers and suggesting stable, accurate measurements.

Columns such as Coolant Pressure, Air System Pressure, and Torque show stable outputs, indicating reliable data without significant outliers and consistent readings, reflecting good data quality.

Hydraulic Pressure (-0.553) and Torque (Nm) (-0.411) are strongly negatively correlated with Downtime, while Cutting Force (kN) (0.454) is strongly positively correlated with Downtime.

Spindle Speed (0.2730) is moderately correlated with Downtime

Downtime occurrence is 51.49% for machine failures and 48.51% for no machine downtime failure.

The data is predominantly from 2022 (99.03%), with only 0.97% from 2021.

Hydraulic Pressure below 100 bar increases the likelihood of machine failure; maintaining optimal hydraulic pressure reduces downtime.

Cutting Force between 1kN to 2.2kN or 2.6kN to 3.3kN decreases the chance of machine failure; monitoring and controlling cutting force prevents increased downtime.

Torque between 30Nm to 35Nm minimizes the chance of machine failure; maintaining adequate torque levels helps minimize downtime.

Hydraulic Pressure between 75 and 175 bar and Cutting Force between 1kN to 2.2kN or 2.9kN to 3.4kN reduce the chance of machine failure.