**SQL queries**

**-- Dropping the table if exist**

drop table machine\_downtime;

**-- creating the table as machine\_downtime**

CREATE TABLE machine\_downtime

(

Date date ,

Machine\_ID varchar(330) ,

Assembly\_Line\_No varchar(330) ,

Hydraulic\_Pressure\_bar float,

Coolant\_Pressure\_bar float,

Air\_System\_Pressure\_bar float,

Coolant\_Temperature float,

Hydraulic\_Oil\_Temperature\_Â°C float,

Spindle\_Bearing\_Temperature\_Â°C float,

Spindle\_Vibration\_Âµm float,

Tool\_Vibration\_Âµm float,

Spindle\_Speed\_RPM int,

Voltage\_volts int,

Torque\_Nm float,

Cutting\_kN float,

Downtime varchar(225)

);

**-- Importing the data of csv file in table as machine\_downtime**

SET datestyle = dmy;

copy machine\_downtime from 'C:\Users\DELL\Desktop\postgressql\360digitmg internship\Machine Downtime.csv'DELIMITER ',' CSV HEADER;

**-- To show the table**

select \* from machine\_downtime;

**-- To find the count of data in table**

select count(\*)

from machine\_downtime;

SELECT Downtime, COUNT(\*) AS num\_events

FROM machine\_downtime

GROUP BY Downtime

ORDER BY num\_events DESC;

**-- To find different Machine\_Id and their count**

select Machine\_ID,count(\*)

from machine\_downtime

group by Machine\_ID

having count(\*)>1;

**-- To find different Assembly\_line\_no and their count**

select Assembly\_line\_no ,count(\*)

from machine\_downtime

group by Assembly\_line\_no

having count(\*)>1;

**-- to identify count of machine in different assembly\_line**

select Machine\_ID, Assembly\_Line\_No, count(\*)

from machine\_downtime

group by Machine\_ID, Assembly\_Line\_No

order by Machine\_ID

**-- to identify count of machine\_id which has machine failure and which assembly\_line\_no it is assigned**

select Machine\_ID, Assembly\_Line\_No,Downtime, count(\*)

from machine\_downtime

group by Machine\_ID, Assembly\_Line\_No,Downtime

order by Machine\_ID

**-- converting categorical variable into numeric**

SELECT

CASE

WHEN Machine\_ID = 'Makino-L1-Unit1-2013' THEN 1

WHEN Machine\_ID = 'Makino-L2-Unit1-2015' THEN 2

ELSE 3

END AS Machine\_ID\_Num,

CASE

WHEN Assembly\_Line\_No = 'Shopfloor-L1' THEN 1

WHEN Assembly\_Line\_No = 'Shopfloor-L2' THEN 2

ELSE 3

END AS Assembly\_Line\_No\_Num

FROM machine\_downtime;

**-- checking the null values in the columns**

SELECT \*

FROM machine\_downtime

WHERE

Date IS NULL

OR Machine\_ID IS NULL

OR Assembly\_Line\_No IS NULL

OR Hydraulic\_Pressure\_bar IS NULL

OR Coolant\_Pressure\_bar IS NULL

OR Air\_System\_Pressure\_bar IS NULL

OR Coolant\_Temperature IS NULL

OR Hydraulic\_Oil\_Temperature\_Â°C IS NULL

OR Spindle\_Bearing\_Temperature\_Â°C IS NULL

OR Spindle\_Vibration\_Âµm IS NULL

OR Tool\_Vibration\_Âµm IS NULL

OR Spindle\_Speed\_RPM IS NULL

OR Voltage\_volts IS NULL

OR Torque\_Nm IS NULL

OR Cutting\_kN IS NULL

OR Downtime IS NULL

;

**-- count of null values in the columns**

SELECT

SUM(CASE WHEN date IS NULL THEN 1 ELSE 0 END) AS date\_null\_count,

SUM(CASE WHEN machine\_id IS NULL THEN 1 ELSE 0 END) AS machine\_id\_null\_count,

SUM(CASE WHEN assembly\_line\_no IS NULL THEN 1 ELSE 0 END) AS assembly\_line\_no\_null\_count,

SUM(CASE WHEN hydraulic\_pressure\_bar IS NULL THEN 1 ELSE 0 END) AS hydraulic\_pressure\_null\_count,

SUM(CASE WHEN coolant\_pressure\_bar IS NULL THEN 1 ELSE 0 END) AS coolant\_pressure\_null\_count,

SUM(CASE WHEN air\_system\_pressure\_bar IS NULL THEN 1 ELSE 0 END) AS air\_system\_pressure\_null\_count,

SUM(CASE WHEN coolant\_temperature IS NULL THEN 1 ELSE 0 END) AS coolant\_temperature\_null\_count,

SUM(CASE WHEN hydraulic\_oil\_temperature\_Â°C IS NULL THEN 1 ELSE 0 END) AS hydraulic\_oil\_temperature\_null\_count,

SUM(CASE WHEN spindle\_bearing\_temperature\_Â°C IS NULL THEN 1 ELSE 0 END) AS spindle\_bearing\_temperature\_null\_count,

SUM(CASE WHEN spindle\_vibration\_Âµm IS NULL THEN 1 ELSE 0 END) AS spindle\_vibration\_null\_count,

SUM(CASE WHEN Tool\_vibration\_Âµm IS NULL THEN 1 ELSE 0 END) AS Tool\_vibration\_null\_count,

SUM(CASE WHEN spindle\_speed\_RPM IS NULL THEN 1 ELSE 0 END) AS spindle\_speed\_null\_count,

SUM(CASE WHEN voltage\_volts IS NULL THEN 1 ELSE 0 END) AS voltage\_null\_count,

SUM(CASE WHEN torque\_Nm IS NULL THEN 1 ELSE 0 END) AS torque\_null\_count,

SUM(CASE WHEN cutting\_kN IS NULL THEN 1 ELSE 0 END) AS cutting\_null\_count,

SUM(CASE WHEN downtime IS NULL THEN 1 ELSE 0 END) AS downtime\_null\_count

FROM machine\_downtime;

**-- statistics summary**

select

'Hydraulic\_Pressure\_bar' AS variable\_name,

AVG(Hydraulic\_Pressure\_bar)as mean,

var\_samp(Hydraulic\_Pressure\_bar)as variance,

STDDEV\_samp(hydraulic\_pressure\_bar) AS stddev,

MIN(hydraulic\_pressure\_bar) AS min,

MAX(hydraulic\_pressure\_bar) AS max

FROM machine\_downtime

UNION ALL

select

'Coolant\_Pressure\_bar' as variable\_name,

AVG(Coolant\_Pressure\_bar)as mean,

var\_samp(Coolant\_Pressure\_bar)as variance,

STDDEV\_samp(Coolant\_pressure\_bar) AS stddev,

MIN(Coolant\_pressure\_bar) AS min,

MAX(Coolant\_pressure\_bar) AS max

FROM machine\_downtime

UNION ALL

select

'Air\_System\_Pressure\_bar' as variable\_name,

AVG(Air\_System\_Pressure\_bar)as mean,

var\_samp(Air\_System\_Pressure\_bar)as variance,

STDDEV\_samp(Air\_System\_Pressure\_bar) AS stddev,

MIN(Air\_System\_Pressure\_bar) AS min,

MAX(Air\_System\_Pressure\_bar) AS max

FROM machine\_downtime

UNION ALL

select

'Coolant\_Temperature' as variable\_name,

AVG(Coolant\_Temperature)as mean,

var\_samp(Coolant\_Temperature)as variance,

STDDEV\_samp(Coolant\_Temperature) AS stddev,

MIN(Coolant\_Temperature) AS min,

MAX(Coolant\_Temperature) AS max

FROM machine\_downtime

UNION ALL

select

'Hydraulic\_Oil\_Temperature\_Â°C' as variable\_name,

AVG(Hydraulic\_Oil\_Temperature\_Â°C)as mean,

var\_samp(Hydraulic\_Oil\_Temperature\_Â°C)as variance,

STDDEV\_samp(Hydraulic\_Oil\_Temperature\_Â°C) AS stddev,

MIN(Hydraulic\_Oil\_Temperature\_Â°C) AS min,

MAX(Hydraulic\_Oil\_Temperature\_Â°C) AS max

FROM machine\_downtime

UNION ALL

select

'Spindle\_Bearing\_Temperature\_Â°C' as variable\_name,

AVG(Spindle\_Bearing\_Temperature\_Â°C)as mean,

var\_samp(Spindle\_Bearing\_Temperature\_Â°C)as variance,

STDDEV\_samp(Spindle\_Bearing\_Temperature\_Â°C) AS stddev,

MIN(Spindle\_Bearing\_Temperature\_Â°C) AS min,

MAX(Spindle\_Bearing\_Temperature\_Â°C) AS max

FROM machine\_downtime

UNION ALL

select

'Spindle\_Vibration\_Âµm' as variable\_name,

AVG(Spindle\_Vibration\_Âµm)as mean,

var\_samp(Spindle\_Vibration\_Âµm)as variance,

STDDEV\_samp(Spindle\_Vibration\_Âµm) AS stddev,

MIN(Spindle\_Vibration\_Âµm) AS min,

MAX(Spindle\_Vibration\_Âµm) AS max

FROM machine\_downtime

UNION ALL

select

'Tool\_Vibration\_Âµm' as variable\_name,

AVG(Tool\_Vibration\_Âµm)as mean,

var\_samp(Tool\_Vibration\_Âµm)as variance,

STDDEV\_samp(Tool\_Vibration\_Âµm) AS stddev,

MIN(Tool\_Vibration\_Âµm) AS min,

MAX(Tool\_Vibration\_Âµm) AS max

FROM machine\_downtime

UNION ALL

select

'Spindle\_Speed\_RPM' as variable\_name,

AVG(Spindle\_Speed\_RPM)as mean,

var\_samp(Spindle\_Speed\_RPM)as variance,

STDDEV\_samp(Spindle\_Speed\_RPM) AS stddev,

MIN(Spindle\_Speed\_RPM) AS min,

MAX(Spindle\_Speed\_RPM) AS max

FROM machine\_downtime

UNION ALL

select

'Voltage\_volts' as variable\_name,

AVG(Voltage\_volts)as mean,

var\_samp(Voltage\_volts)as variance,

STDDEV\_samp(Voltage\_volts) AS stddev,

MIN(Voltage\_volts) AS min,

MAX(Voltage\_volts) AS max

FROM machine\_downtime

UNION ALL

select

'Torque\_Nm' as variable\_name,

AVG(Torque\_Nm)as mean,

var\_samp(Torque\_Nm)as variance,

STDDEV\_samp(Torque\_Nm) AS stddev,

MIN(Torque\_Nm) AS min,

MAX(Torque\_Nm) AS max

FROM machine\_downtime

UNION ALL

select

'Cutting\_kN' as variable\_name,

AVG(Cutting\_kN)as mean,

var\_samp(Cutting\_kN)as variance,

STDDEV\_samp(Cutting\_kN) AS stddev,

MIN(Cutting\_kN) AS min,

MAX(Cutting\_kN) AS max

FROM machine\_downtime;

-- replacing the null values

UPDATE machine\_downtime

SET

hydraulic\_pressure\_bar = COALESCE(hydraulic\_pressure\_bar, 101.40908377755412), -- Replace with your default value

coolant\_pressure\_bar = COALESCE(coolant\_pressure\_bar, 4.94705847351793), -- Replace with your default value

air\_system\_pressure\_bar = COALESCE(air\_system\_pressure\_bar, 6.4992745962251375), -- Replace with your default value

coolant\_temperature = COALESCE(coolant\_temperature, 18.559887459807154), -- Replace with your default value

hydraulic\_oil\_temperature\_Â°C = COALESCE(hydraulic\_oil\_temperature\_Â°C, 47.6183172302738), -- Replace with your default value

spindle\_bearing\_temperature\_Â°C = COALESCE(spindle\_bearing\_temperature\_Â°C, 35.06369835539507), -- Replace with your default value

spindle\_vibration\_Âµm = COALESCE(spindle\_vibration\_Âµm, 1.0093342707914827), -- Replace with your default value

tool\_vibration\_Âµm = COALESCE(tool\_vibration\_Âµm, 25.41197509039763), -- Replace with your default value

spindle\_speed\_RPM = COALESCE(spindle\_speed\_RPM, 20274.792301523656), -- Replace with your default value

voltage\_volts = COALESCE(voltage\_volts, 348.99679230152367), -- Replace with your default value

torque\_Nm = COALESCE(torque\_Nm, 25.23496773013697), -- Replace with your default value

cutting\_kN = COALESCE(cutting\_kN , 2.7825511432009598) -- Replace with your default value

WHERE

hydraulic\_pressure\_bar IS NULL

OR coolant\_pressure\_bar IS NULL

OR air\_system\_pressure\_bar IS NULL

OR coolant\_temperature IS NULL

OR hydraulic\_oil\_temperature\_Â°C IS NULL

OR spindle\_bearing\_temperature\_Â°C IS NULL

OR spindle\_vibration\_Âµm IS NULL

OR tool\_vibration\_Âµm IS NULL

OR spindle\_speed\_RPM IS NULL

OR voltage\_volts IS NULL

OR torque\_Nm IS NULL

OR cutting\_kN IS NULL;

**-- Descriptive statistics summary**

SELECT

'Hydraulic\_Pressure\_bar' AS variable\_name,

AVG(Hydraulic\_Pressure\_bar) AS mean,

var\_samp(Hydraulic\_Pressure\_bar) AS variance,

STDDEV\_samp(Hydraulic\_Pressure\_bar) AS stddev,

MIN(Hydraulic\_Pressure\_bar) AS min,

MAX(Hydraulic\_Pressure\_bar) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Hydraulic\_Pressure\_bar), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Hydraulic\_Pressure\_bar), 4)) AS kurtosis

FROM (

SELECT

Hydraulic\_Pressure\_bar,

POWER(Hydraulic\_Pressure\_bar - AVG(Hydraulic\_Pressure\_bar) OVER (), 3) AS cubed\_diff,

POWER(Hydraulic\_Pressure\_bar - AVG(Hydraulic\_Pressure\_bar) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Coolant\_Pressure\_bar' as variable\_name,

AVG(Coolant\_Pressure\_bar)as mean,

var\_samp(Coolant\_Pressure\_bar)as variance,

STDDEV\_samp(Coolant\_pressure\_bar) AS stddev,

MIN(Coolant\_pressure\_bar) AS min,

MAX(Coolant\_pressure\_bar) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Coolant\_Pressure\_bar), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Coolant\_Pressure\_bar), 4)) AS kurtosis

FROM (

SELECT

Coolant\_Pressure\_bar,

POWER(Coolant\_Pressure\_bar - AVG(Coolant\_Pressure\_bar) OVER (), 3) AS cubed\_diff,

POWER(Coolant\_Pressure\_bar - AVG(Coolant\_Pressure\_bar) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Air\_System\_Pressure\_bar' as variable\_name,

AVG(Air\_System\_Pressure\_bar)as mean,

var\_samp(Air\_System\_Pressure\_bar)as variance,

STDDEV\_samp(Air\_System\_Pressure\_bar) AS stddev,

MIN(Air\_System\_Pressure\_bar) AS min,

MAX(Air\_System\_Pressure\_bar) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Air\_System\_Pressure\_bar), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Air\_System\_Pressure\_bar), 4)) AS kurtosis

FROM (

SELECT

Air\_System\_Pressure\_bar,

POWER(Air\_System\_Pressure\_bar - AVG(Air\_System\_Pressure\_bar) OVER (), 3) AS cubed\_diff,

POWER(Air\_System\_Pressure\_bar - AVG(Air\_System\_Pressure\_bar) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Coolant\_Temperature' as variable\_name,

AVG(Coolant\_Temperature)as mean,

var\_samp(Coolant\_Temperature)as variance,

STDDEV\_samp(Coolant\_Temperature) AS stddev,

MIN(Coolant\_Temperature) AS min,

MAX(Coolant\_Temperature) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Coolant\_Temperature), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Coolant\_Temperature), 4)) AS kurtosis

FROM (

SELECT

Coolant\_Temperature,

POWER(Coolant\_Temperature - AVG(Coolant\_Temperature) OVER (), 3) AS cubed\_diff,

POWER(Coolant\_Temperature - AVG(Coolant\_Temperature) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Hydraulic\_Oil\_Temperature\_Â°C' as variable\_name,

AVG(Hydraulic\_Oil\_Temperature\_Â°C)as mean,

var\_samp(Hydraulic\_Oil\_Temperature\_Â°C)as variance,

STDDEV\_samp(Hydraulic\_Oil\_Temperature\_Â°C) AS stddev,

MIN(Hydraulic\_Oil\_Temperature\_Â°C) AS min,

MAX(Hydraulic\_Oil\_Temperature\_Â°C) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Hydraulic\_Oil\_Temperature\_Â°C), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Hydraulic\_Oil\_Temperature\_Â°C), 4)) AS kurtosis

FROM (

SELECT

Hydraulic\_Oil\_Temperature\_Â°C,

POWER(Hydraulic\_Oil\_Temperature\_Â°C - AVG(Hydraulic\_Oil\_Temperature\_Â°C) OVER (), 3) AS cubed\_diff,

POWER(Hydraulic\_Oil\_Temperature\_Â°C - AVG(Hydraulic\_Oil\_Temperature\_Â°C) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Spindle\_Bearing\_Temperature\_Â°C' as variable\_name,

AVG(Spindle\_Bearing\_Temperature\_Â°C)as mean,

var\_samp(Spindle\_Bearing\_Temperature\_Â°C)as variance,

STDDEV\_samp(Spindle\_Bearing\_Temperature\_Â°C) AS stddev,

MIN(Spindle\_Bearing\_Temperature\_Â°C) AS min,

MAX(Spindle\_Bearing\_Temperature\_Â°C) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Bearing\_Temperature\_Â°C), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Bearing\_Temperature\_Â°C), 4)) AS kurtosis

FROM (

SELECT

Spindle\_Bearing\_Temperature\_Â°C,

POWER(Spindle\_Bearing\_Temperature\_Â°C - AVG(Spindle\_Bearing\_Temperature\_Â°C) OVER (), 3) AS cubed\_diff,

POWER(Spindle\_Bearing\_Temperature\_Â°C - AVG(Spindle\_Bearing\_Temperature\_Â°C) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Spindle\_Vibration\_Âµm' as variable\_name,

AVG(Spindle\_Vibration\_Âµm)as mean,

var\_samp(Spindle\_Vibration\_Âµm)as variance,

STDDEV\_samp(Spindle\_Vibration\_Âµm) AS stddev,

MIN(Spindle\_Vibration\_Âµm) AS min,

MAX(Spindle\_Vibration\_Âµm) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Vibration\_Âµm), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Vibration\_Âµm), 4)) AS kurtosis

FROM (

SELECT

Spindle\_Vibration\_Âµm,

POWER(Spindle\_Vibration\_Âµm - AVG(Spindle\_Vibration\_Âµm) OVER (), 3) AS cubed\_diff,

POWER(Spindle\_Vibration\_Âµm - AVG(Spindle\_Vibration\_Âµm) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Tool\_Vibration\_Âµm' as variable\_name,

AVG(Tool\_Vibration\_Âµm)as mean,

var\_samp(Tool\_Vibration\_Âµm)as variance,

STDDEV\_samp(Tool\_Vibration\_Âµm) AS stddev,

MIN(Tool\_Vibration\_Âµm) AS min,

MAX(Tool\_Vibration\_Âµm) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Tool\_Vibration\_Âµm), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Tool\_Vibration\_Âµm), 4)) AS kurtosis

FROM (

SELECT

Tool\_Vibration\_Âµm,

POWER(Tool\_Vibration\_Âµm - AVG(Tool\_Vibration\_Âµm) OVER (), 3) AS cubed\_diff,

POWER(Tool\_Vibration\_Âµm - AVG(Tool\_Vibration\_Âµm) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Spindle\_Speed\_RPM' as variable\_name,

AVG(Spindle\_Speed\_RPM)as mean,

var\_samp(Spindle\_Speed\_RPM)as variance,

STDDEV\_samp(Spindle\_Speed\_RPM) AS stddev,

MIN(Spindle\_Speed\_RPM) AS min,

MAX(Spindle\_Speed\_RPM) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Speed\_RPM), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Spindle\_Speed\_RPM), 4)) AS kurtosis

FROM (

SELECT

Spindle\_Speed\_RPM,

POWER(Spindle\_Speed\_RPM - AVG(Spindle\_Speed\_RPM) OVER (), 3) AS cubed\_diff,

POWER(Spindle\_Speed\_RPM - AVG(Spindle\_Speed\_RPM) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Voltage\_volts' as variable\_name,

AVG(Voltage\_volts)as mean,

var\_samp(Voltage\_volts)as variance,

STDDEV\_samp(Voltage\_volts) AS stddev,

MIN(Voltage\_volts) AS min,

MAX(Voltage\_volts) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Voltage\_volts), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Voltage\_volts), 4)) AS kurtosis

FROM (

SELECT

Voltage\_volts,

POWER(Voltage\_volts - AVG(Voltage\_volts) OVER (), 3) AS cubed\_diff,

POWER(Voltage\_volts - AVG(Voltage\_volts) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Torque\_Nm' as variable\_name,

AVG(Torque\_Nm)as mean,

var\_samp(Torque\_Nm)as variance,

STDDEV\_samp(Torque\_Nm) AS stddev,

MIN(Torque\_Nm) AS min,

MAX(Torque\_Nm) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Torque\_Nm), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Torque\_Nm), 4)) AS kurtosis

FROM (

SELECT

Torque\_Nm,

POWER(Torque\_Nm - AVG(Torque\_Nm) OVER (), 3) AS cubed\_diff,

POWER(Torque\_Nm - AVG(Torque\_Nm) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery

UNION ALL

select

'Cutting\_kN' as variable\_name,

AVG(Cutting\_kN)as mean,

var\_samp(Cutting\_kN)as variance,

STDDEV\_samp(Cutting\_kN) AS stddev,

MIN(Cutting\_kN) AS min,

MAX(Cutting\_kN) AS max,

SUM(cubed\_diff) / (COUNT(\*) \* POWER(STDDEV(Cutting\_kN), 3)) AS skewness,

SUM(fourth\_power\_diff) / (COUNT(\*) \* POWER(STDDEV(Cutting\_kN), 4)) AS kurtosis

FROM (

SELECT

Cutting\_kN,

POWER(Cutting\_kN - AVG(Cutting\_kN) OVER (), 3) AS cubed\_diff,

POWER(Cutting\_kN - AVG(Cutting\_kN) OVER (), 4) AS fourth\_power\_diff

FROM machine\_downtime

) subquery;

**-- Correlation**

SELECT

'Correlation' AS statistics,

NULL AS date,

NULL AS machine\_id,

NULL AS assembly\_line\_no,

CORR(Hydraulic\_Pressure\_bar, Coolant\_Pressure\_bar) AS hydraulic\_coolant\_corr,

CORR(Hydraulic\_Pressure\_bar, Air\_System\_Pressure\_bar) AS hydraulic\_air\_corr,

CORR(Hydraulic\_Pressure\_bar, Coolant\_Temperature) AS hydraulic\_coolant\_temp\_corr,

CORR(Hydraulic\_Pressure\_bar, Hydraulic\_Oil\_Temperature\_Â°C) AS hydraulic\_oil\_temp\_corr,

CORR(Hydraulic\_Pressure\_bar, Spindle\_Bearing\_Temperature\_Â°C) AS hydraulic\_bearing\_temp\_corr,

CORR(Hydraulic\_Pressure\_bar, Spindle\_Vibration\_Âµm) AS hydraulic\_spindle\_vib\_corr,

CORR(Hydraulic\_Pressure\_bar, Tool\_Vibration\_Âµm) AS hydraulic\_tool\_vib\_corr,

CORR(Hydraulic\_Pressure\_bar, Spindle\_Speed\_RPM) AS hydraulic\_spindle\_speed\_corr,

CORR(Hydraulic\_Pressure\_bar, Voltage\_volts) AS hydraulic\_voltage\_corr,

CORR(Hydraulic\_Pressure\_bar, Torque\_Nm) AS hydraulic\_torque\_corr,

CORR(Hydraulic\_Pressure\_bar, Cutting\_kN) AS hydraulic\_cutting\_corr,

CORR(Coolant\_Pressure\_bar, Air\_System\_Pressure\_bar) AS coolant\_air\_corr,

CORR(Coolant\_Pressure\_bar, Coolant\_Temperature) AS coolant\_temp\_corr,

CORR(Coolant\_Pressure\_bar, Hydraulic\_Oil\_Temperature\_Â°C) AS coolant\_oil\_temp\_corr,

CORR(Coolant\_Pressure\_bar, Spindle\_Bearing\_Temperature\_Â°C) AS coolant\_bearing\_temp\_corr,

CORR(Coolant\_Pressure\_bar, Spindle\_Vibration\_Âµm) AS coolant\_spindle\_vib\_corr,

CORR(Coolant\_Pressure\_bar, Tool\_Vibration\_Âµm) AS coolant\_tool\_vib\_corr,

CORR(Coolant\_Pressure\_bar, Spindle\_Speed\_RPM) AS coolant\_spindle\_speed\_corr,

CORR(Coolant\_Pressure\_bar, Voltage\_volts) AS coolant\_voltage\_corr,

CORR(Coolant\_Pressure\_bar, Torque\_Nm) AS coolant\_torque\_corr,

CORR(Coolant\_Pressure\_bar, Cutting\_kN) AS coolant\_cutting\_corr,

CORR(Air\_System\_Pressure\_bar, Coolant\_Temperature) AS air\_coolant\_temp\_corr,

CORR(Air\_System\_Pressure\_bar, Hydraulic\_Oil\_Temperature\_Â°C) AS air\_oil\_temp\_corr,

CORR(Air\_System\_Pressure\_bar, Spindle\_Bearing\_Temperature\_Â°C) AS air\_bearing\_temp\_corr,

CORR(Air\_System\_Pressure\_bar, Spindle\_Vibration\_Âµm) AS air\_spindle\_vib\_corr,

CORR(Air\_System\_Pressure\_bar, Tool\_Vibration\_Âµm) AS air\_tool\_vib\_corr,

CORR(Air\_System\_Pressure\_bar, Spindle\_Speed\_RPM) AS air\_spindle\_speed\_corr,

CORR(Air\_System\_Pressure\_bar, Voltage\_volts) AS air\_voltage\_corr,

CORR(Air\_System\_Pressure\_bar, Torque\_Nm) AS air\_torque\_corr,

CORR(Air\_System\_Pressure\_bar, Cutting\_kN) AS air\_cutting\_corr,

CORR(Coolant\_Temperature, Hydraulic\_Oil\_Temperature\_Â°C) AS coolant\_oil\_temp\_corr,

CORR(Coolant\_Temperature, Spindle\_Bearing\_Temperature\_Â°C) AS coolant\_bearing\_temp\_corr,

CORR(Coolant\_Temperature, Spindle\_Vibration\_Âµm) AS coolant\_spindle\_vib\_corr,

CORR(Coolant\_Temperature, Tool\_Vibration\_Âµm) AS coolant\_tool\_vib\_corr,

CORR(Coolant\_Temperature, Spindle\_Speed\_RPM) AS coolant\_spindle\_speed\_corr,

CORR(Coolant\_Temperature, Voltage\_volts) AS coolant\_voltage\_corr,

CORR(Coolant\_Temperature, Torque\_Nm) AS coolant\_torque\_corr,

CORR(Coolant\_Temperature, Cutting\_kN) AS coolant\_cutting\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Spindle\_Bearing\_Temperature\_Â°C) AS oil\_bearing\_temp\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Spindle\_Vibration\_Âµm) AS oil\_spindle\_vib\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Tool\_Vibration\_Âµm) AS oil\_tool\_vib\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Spindle\_Speed\_RPM) AS oil\_spindle\_speed\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Voltage\_volts) AS oil\_voltage\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Torque\_Nm) AS oil\_torque\_corr,

CORR(Hydraulic\_Oil\_Temperature\_Â°C, Cutting\_kN) AS oil\_cutting\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Spindle\_Vibration\_Âµm) AS bearing\_spindle\_vib\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Tool\_Vibration\_Âµm) AS bearing\_tool\_vib\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Spindle\_Speed\_RPM) AS bearing\_spindle\_speed\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Voltage\_volts) AS bearing\_voltage\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Torque\_Nm) AS bearing\_torque\_corr,

CORR(Spindle\_Bearing\_Temperature\_Â°C, Cutting\_kN) AS bearing\_cutting\_corr,

CORR(Spindle\_Vibration\_Âµm, Tool\_Vibration\_Âµm) AS spindle\_tool\_vib\_corr,

CORR(Spindle\_Vibration\_Âµm, Spindle\_Speed\_RPM) AS spindle\_speed\_vib\_corr,

CORR(Spindle\_Vibration\_Âµm, Voltage\_volts) AS spindle\_voltage\_corr,

CORR(Spindle\_Vibration\_Âµm, Torque\_Nm) AS spindle\_torque\_corr,

CORR(Spindle\_Vibration\_Âµm, Cutting\_kN) AS spindle\_cutting\_corr,

CORR(Tool\_Vibration\_Âµm, Spindle\_Speed\_RPM) AS tool\_spindle\_speed\_corr,

CORR(Tool\_Vibration\_Âµm, Voltage\_volts) AS tool\_voltage\_corr,

CORR(Tool\_Vibration\_Âµm, Torque\_Nm) AS tool\_torque\_corr,

CORR(Tool\_Vibration\_Âµm, Cutting\_kN) AS tool\_cutting\_corr,

CORR(Spindle\_Speed\_RPM, Voltage\_volts) AS spindle\_voltage\_corr,

CORR(Spindle\_Speed\_RPM, Torque\_Nm) AS spindle\_torque\_corr,

CORR(Spindle\_Speed\_RPM, Cutting\_kN) AS spindle\_cutting\_corr,

CORR(Voltage\_volts, Torque\_Nm) AS voltage\_torque\_corr,

CORR(Voltage\_volts, Cutting\_kN) AS voltage\_cutting\_corr,

CORR(Torque\_Nm, Cutting\_kN) AS torque\_cutting\_corr,

NULL AS downtime

FROM machine\_downtime;