

**ANL503**

**Data Wrangling**

**Tutor-Marked Assessment - January Semester 2025**

**PI no. Y2510707**

**Name: Ng Chee Wee**

**Submission Date:22.03.2025**

Contents

[Create a new table named TMA\_data\_labelled 3](#_Toc193540319)

[Convert org\_date to calendar date variable 6](#_Toc193540320)

[ALTER TABLE to modify data type 7](#_Toc193540321)

[Summary table #1 – Coordination influenced Willingness to Recommend. 9](#_Toc193540322)

[Summary table #2 – Age Group and Gender based perception of services 11](#_Toc193540323)

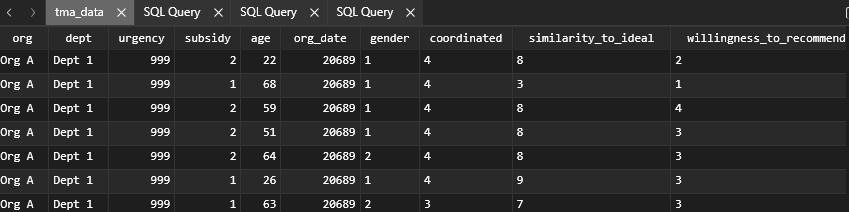
[References 13](#_Toc193540324)

[Data Appendix 14](#_Toc193540325)

# Create a new table named TMA\_data\_labelled

1a.

The given dataset TMA\_data.sql was imported into MySQL server. The TMA\_data table consist of 4858 rows and 10 columns.



MySQL code to create table TMA\_data\_labelled:

-- Create new table TMA\_data\_labelled based on TMA\_data

DROP TABLE IF EXISTS TMA\_data\_labelled;

CREATE TABLE TMA\_data\_labelled AS

SELECT

org,

dept,

-- map urgency numeric values to text labels described in data dictionary

CASE urgency

WHEN 1 THEN 'P1'

WHEN 2 THEN 'P2'

WHEN 3 THEN 'P3'

WHEN 4 THEN 'P4'

WHEN 5 THEN 'PX'

WHEN 999 THEN 'NA'

ELSE NULL

END AS urgency,

-- map subsidy numeric values to text labels described in data dictionary

CASE subsidy

WHEN 1 THEN 'Y'

WHEN 2 THEN 'N'

ELSE NULL

END AS subsidy,

age,

-- convert the org\_date variable into a calendar date variable

DATE\_ADD('1965-08-09',INTERVAL org\_date DAY) AS caldate,

-- map gender numeric values to text labels described in data dictionary

CASE gender

WHEN 1 THEN 'M'

WHEN 2 THEN 'F'

ELSE NULL

END AS gender,

-- map coordinated numeric values to text labels described in data dictionary

CASE coordinated

WHEN 1 THEN 'Never'

WHEN 2 THEN 'Sometimes'

WHEN 3 THEN 'Usually'

WHEN 4 THEN 'Always'

WHEN 5 THEN 'Unsure'

ELSE NULL

END AS coordinated,

-- map similarity\_to\_ideal numeric values to text labels described in data dictionary

CASE similarity\_to\_ideal

WHEN 11 THEN 'Unsure'

WHEN 999 THEN 'NA'

ELSE similarity\_to\_ideal

END AS similarity\_to\_ideal,

-- map willingness\_to\_recommend numeric values to text labels described in data dictionary

CASE willingness\_to\_recommend

WHEN 1 THEN 'Definitely No'

WHEN 2 THEN 'Probably No'

WHEN 3 THEN 'Probably Yes'

WHEN 4 THEN 'Definitely Yes'

WHEN 5 THEN 'Unsure'

WHEN 999 THEN 'NA'

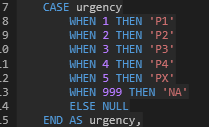
ELSE NULL

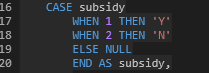
END AS willingness\_to\_recommend

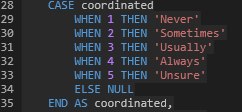
FROM tma\_data

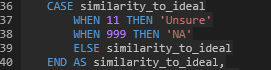
MySQL CREATE TABLE code explained:

To map the numeric values with their corresponding text labels according to the data dictionary, CASE statements were used in SQL query in Table Plus. The transformed columns are ‘urgency’ , ‘subsidy’, ‘gender’, ‘coordinated’, ‘similarity\_to\_ideal’ and ‘willingness\_to\_recommend’.

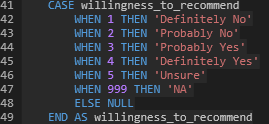
 For the ‘urgency’ variable, when the value is 1, then the corresponding text label is ‘P1’ so on and so forth.

For ‘subsidy’ variable, when the value is 1, then the corresponding text label is ‘Y’ so on and so forth.

 For ‘coordinated’ variable, when the value is 1, then the corresponding text label is ‘Never’, when value is 2, then ‘Sometimes’, so on and so forth.



For the similarity\_to\_ideal variable, the valid values are in numeric 0 – 10. The CASE statement was used to map when the value is 11, then the corresponding text label is ‘Unsure’ and ‘NA’ when the value is 999. In the query, “ELSE similarity\_to\_ideal” tells MySQL to keep 0-10 as is if the value is not 11 or 999.

For the willingness\_to\_recommend variable, when the value is 1, then the corresponding text label is ‘Definitely No’, when value is 2, then ‘Probably No’, so on and so forth.

# Convert org\_date to calendar date variable

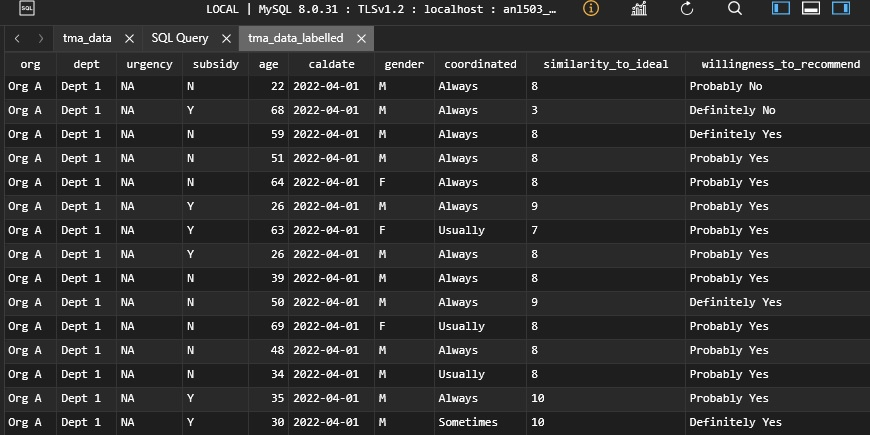
Besides replacing the numeric values with their corresponding text labels, the ‘org\_date’ column was converted into a calendar date variable named ‘caldate’ using DATE\_ADD() function.



According to the data dictionary, the ‘org\_date’ shows the “Number of days since 9 Aug 1965”. This means the base date is 1965-08-09 and the numeric values in each field shows the number of days since the base date. Using DATE\_ADD() function, this code convert the column into a DATE type.

End of code explain for CREATE TABLE code.

The new table was created as TMA\_data\_labelled. The first few rows of the new table is shown below. The numeric values in the identified columns have been mapped to their corresponding text labels and a new column named ‘caldate’ with MySQL DATE type ‘2022-04-01’ has been added.



# ALTER TABLE to modify data type

MySQL code:

-- Modify Data Types of columns in TMA\_data\_labelled using ALTER TABLE

ALTER TABLE TMA\_data\_labelled

MODIFY org VARCHAR(5), -- max length 'Org D'

MODIFY dept VARCHAR(6), -- max length 'Dept 3'

MODIFY urgency VARCHAR(3),

MODIFY subsidy CHAR(1), -- CHAR(1) more efficient than 'VARCHAR' for single-character field

MODIFY age TINYINT UNSIGNED, -- use 'UNSIGNED' as age is positive integer

MODIFY caldate DATE, -- set to DATE type format 'YYYY-MM-DD'

MODIFY gender CHAR(1), -- CHAR(1) more efficient than 'VARCHAR' for single-character field

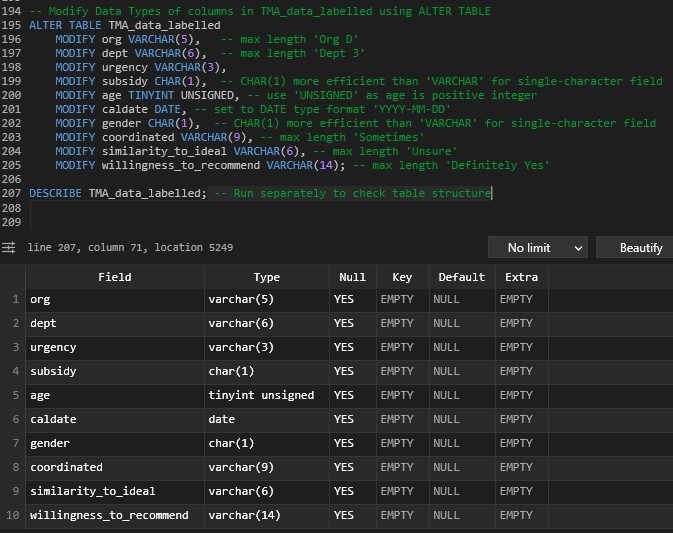
MODIFY coordinated VARCHAR(9), -- max length 'Sometimes'

MODIFY similarity\_to\_ideal VARCHAR(6), -- max length 'Unsure'

MODIFY willingness\_to\_recommend VARCHAR(14); -- max length 'Definitely Yes'

DESCRIBE TMA\_data\_labelled; -- Run separately to check table structure

The DESCRIBE TMA\_data\_labelled statement must be run separately from the ALTER TABLE query and the result is as follows:



ALTER TABLE code explain:

The ALTER TABLE statement was used to modify the data type in order to save storage space while not losing any information.

The valid values in column ‘org’ are ‘Org A’ to ‘Org D’, the maximum length is 5. Hence, using VARCHAR(5) is optimal

The valid values in column ‘dept’ are ‘Dept 1 to ‘Dept 3’, the maximum length is 6. Hence, using VARCHAR(6) is optimal

The valid values in column ‘subsidy’ are ‘Y’ and ‘N’. using CHAR(1) is more storage-efficient than VARCHAR for single-character field.

For the ‘age’ variable, tinyint UNSIGNED was used as UNSIGNED ensures that it only holds non-negative values. In the data dictionary, the valid values of age variable are “any integer greater than zero”.

The caldate variable is assigned the DATE type.

The valid values in column ‘gender’ are ‘M’ and ‘F’. hence, using CHAR(1) is more storage-efficient than VARCHAR for single-character field.

In the ‘coordinated’ variable, the maximum length of the valid values is ‘Sometimes’, which is 9 characters, hence VARCHAR(9) was applied.

In the ‘similarity\_to\_ideal’ variable, the maximum length of the valid values is ‘Unsure’, which is 6 characters, hence VARCHAR(6) was applied.

For the ‘willingness\_to\_recommend’ variable, the maximum length of the valid values is ‘Definitely Yes’, which is 14 characters, hence VARCHAR(14) was applied.

1b.

# Summary table #1 – Coordination influenced Willingness to Recommend.

This table investigate how the coordination level among the different parts of the organisation (measured by the column ‘coordinated’) influenced users’ willingness to recommend the services. I also tracked the result by each year so as to see the trend over time.

MySQL code:

SELECT

YEAR (caldate) AS survey\_year, -- to extract the Year from caldate

coordinated,

-- to calculate the average score of willingness\_to\_recommend round to 2 decimal places, have to convert/map the TEXT labels with a numeric scale using CASE WHEN statement

ROUND(AVG(CASE

WHEN willingness\_to\_recommend = 'Definitely Yes' THEN 4

WHEN willingness\_to\_recommend = 'Probably Yes' THEN 3

WHEN willingness\_to\_recommend = 'Probably No' THEN 2

WHEN willingness\_to\_recommend = 'Definitely No' THEN 1

WHEN willingness\_to\_recommend = 'Unsure' THEN 2.5 -- cannot use 5 as it would skew the result, have to use 2.5 as ‘Unsure’ is between 2 (Probably Yes) & 3 (Probably no)

ELSE NULL

END), 2) AS avg\_willingness\_score,

COUNT(\*) AS response\_count

FROM TMA\_data\_labelled

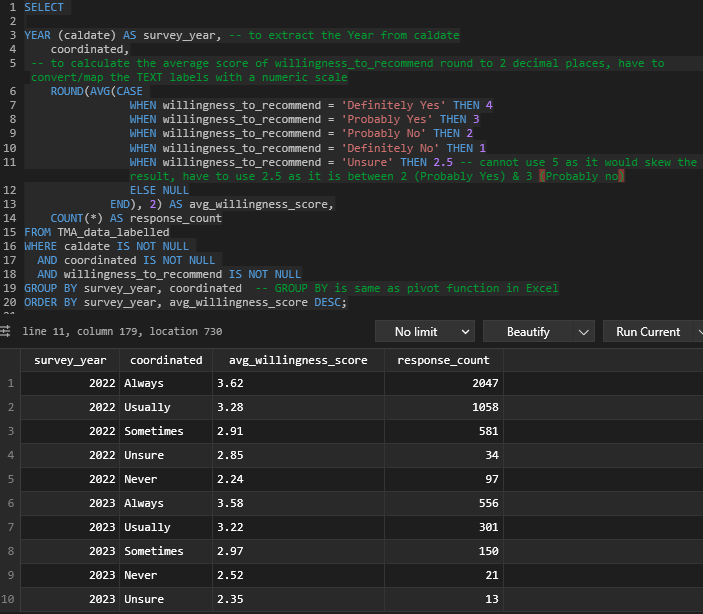
WHERE caldate IS NOT NULL

AND coordinated IS NOT NULL

AND willingness\_to\_recommend IS NOT NULL

GROUP BY survey\_year, coordinated -- GROUP BY statement groups rows that have the same values into summary rows. GROUP BY is same as pivot function in Excel

ORDER BY survey\_year, avg\_willingness\_score DESC;



The results revealed that when different parts of the organisations are coordinated (shown as ALWAYS), the average willingness score (which is average willingness\_to\_recommend) are the highest over the two surveyed year in 2022 and 2023.

On the other hand, when the different parts of the organisations are not coordinated (shown as NEVER), the average willingness score are the lowest (2.24 in 2022 and 2.52 in 2023).

As willingness to recommend can be linked to users’ satisfaction level, this table revealed that better coordination among different parts and departments correlates to the willingness\_to\_recommend score among users. There were not significant differences among the score for each coordinated category in each year.

This query provides actionable insights into the relationships between the coordination level among different departments and the users’ willingness\_to\_recommend and helping the organisation make data-driven decisions.

# Summary table #2 – Age Group and Gender based perception of services

MySQL code:

-- create age\_ranges using CASE WHEN statements

SELECT

CASE

WHEN age BETWEEN 0 AND 29 THEN '0-29'

WHEN age BETWEEN 30 AND 39 THEN '30-39'

WHEN age BETWEEN 40 AND 49 THEN '40-49'

WHEN age BETWEEN 50 AND 59 THEN '50-59'

WHEN age >= 60 THEN '60+'

ELSE 'Unknown'

END AS age\_range,

gender,

-- to calculate the average similarity\_to\_ideal score round to 2 decimal places

ROUND(AVG(CASE

WHEN similarity\_to\_ideal BETWEEN 0 AND 10 THEN similarity\_to\_ideal

WHEN similarity\_to\_ideal = 'Unsure' THEN 5.5 -- cannot use 11 as it would skew the result, have to use 5.5 as it is between the scale of 0 and 10

ELSE NULL

END), 2) AS avg\_similarity\_to\_ideal

FROM

tma\_data\_labelled

WHERE age IS NOT NULL

AND gender IS NOT NULL

AND similarity\_to\_ideal IS NOT NULL

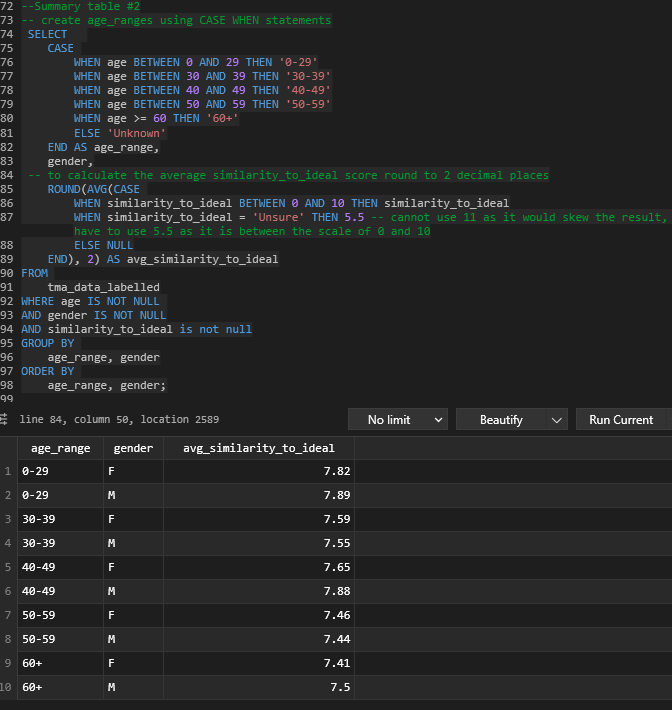
GROUP BY -- GROUP BY statement groups rows that have the same values into summary rows. GROUP BY is same as pivot function in Excel

age\_range, gender

ORDER BY

age\_range, gender;

This table investigate the average similarity\_to\_ideal score among users of various age groups and by their genders.



The results reveal how the different age groups and genders influence their perception of the services rendered by the organisations relative to their ideal. Generally the average similarity to ideal scores are quite high, suggesting the services rendered are relatively close to the perception of the users. However, if we zoomed in further, the lower age groups have higher scores than the older age groups. This could suggest that younger users have lower expectations than their older peers.

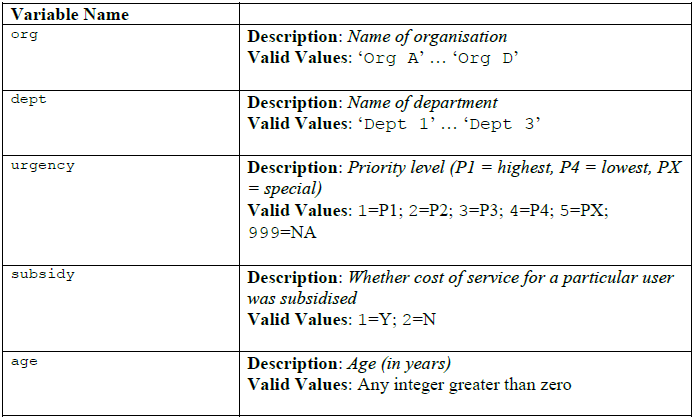
Across the different age groups, the differences in the scores between genders are rather small, and quite insignificant. This could suggest that the services provided by the organisations are not gender-specific.

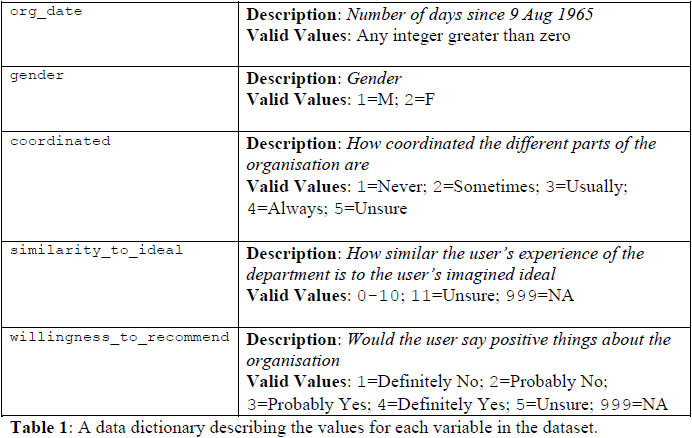
# References

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s/No. | AI tools | Prompt used | Output used in assignment | |
| 1 | ChatGPT | how to convert a numeric variable into a calendar date variable that uses MySQL Date type? the base date is given as 9 Aug 1965. | DATE\_ADD(‘1965-08-09, INTERVAL org\_date DAY) AS caldate | |
| Full output | |  | | |
| 2 ChatGPT | | why must DESCRIBE TMA\_data\_labelled be a separate statement and not as a part of ALTER TABLE statement? | |  |
|  | |  | |  |

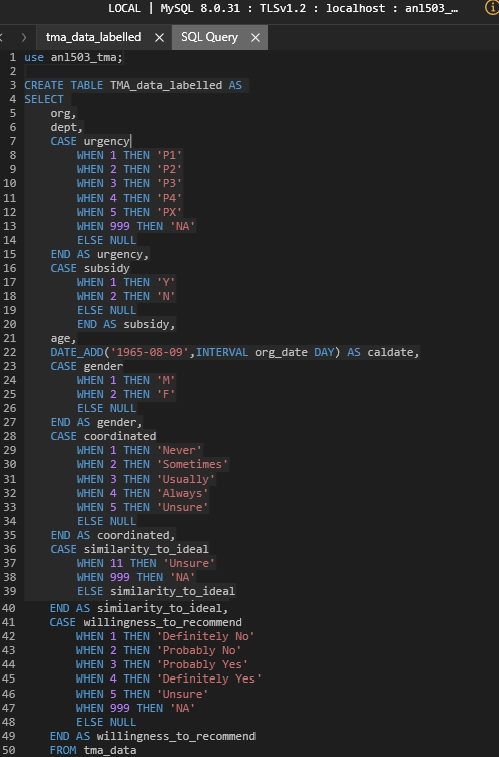
# Data Appendix

Table 1 shows the data dictionary describing the values of each variable.

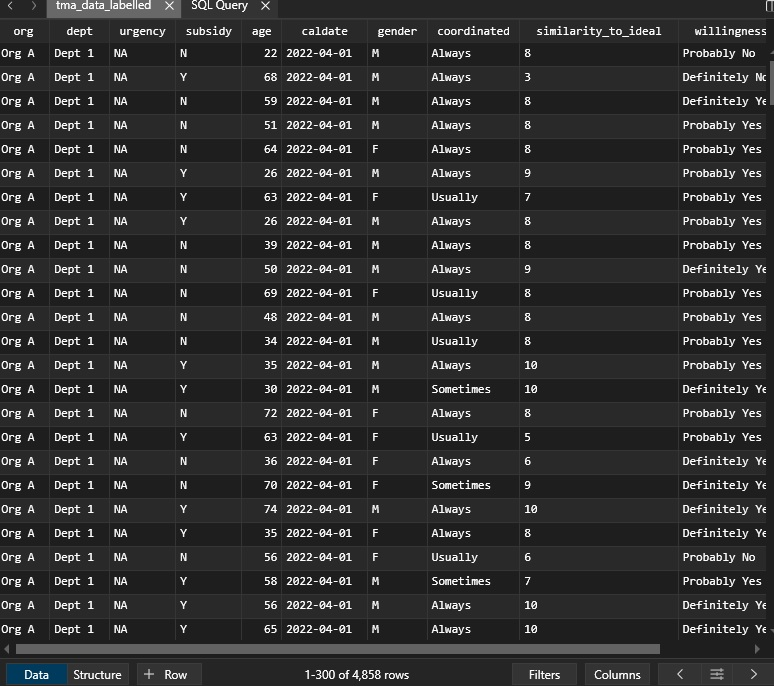




The MySQL query to create tma\_data\_labelled is shown below:



New table TMA\_data\_labelled output table consist of 4858 rows and 10 columns.



The data structure of TMA\_data\_labelled after performing ALTER TABLE is shown as follows:

