|  |  |
| --- | --- |
| **Submitted By:** | Gangula Karthik (223715Y) |
| **Module Group:** | AA2202 |
| **Tutor:** | Ms Lim Ai Huey |
| **Deadline:** | 25/June/2023 |

**IT2381 DATA WRANGLING ASSIGMENT**

Nanyang Land Holdings Data Wrangling Report

**Table of Contents**

[**Overview** 2](#_Toc138581674)

[**Extract Transform Load (ETL) steps** 2](#_Toc138581675)

[**Extract** 2](#_Toc138581676)

[Dataset Examination 2](#_Toc138581677)

[Data Quality Inspection 3](#_Toc138581678)

[**Transform** 3](#_Toc138581679)

[Data Cleansing (Pre-Merging Transformations) 3](#_Toc138581680)

[Schema Unification (Pre-Merging Transformations) 3](#_Toc138581681)

[Data Enrichment (Pre-Merging Transformations) 4](#_Toc138581682)

[Data Formatting (Pre-Merging Transformations) 4](#_Toc138581683)

[Data Merging 4](#_Toc138581684)

[Data Cleansing (Post-Merging Transformations) 4](#_Toc138581685)

[Data Enrichment (Post-Merging Transformations) 4](#_Toc138581686)

[Data Formatting (Post-Merging Transformations) 5](#_Toc138581687)

[Data Anonymization (Post-Merging Transformations) 5](#_Toc138581688)

[**Load** 6](#_Toc138581689)

[Final Adjustments 6](#_Toc138581690)

[Data Storage 6](#_Toc138581691)

[**Reflection** 6](#_Toc138581692)

[**Citations** 7](#_Toc138581693)

# **Overview**

This report will detail the data preparation process for Nanyang Land Holdings with the primary focus on **analysing total customer spending across the 4 different malls** (N, E, W, S).

The data preparation process was carried out using KNIME, involving the merging the four datasets into a unified dataset for analysis, detecting and correcting errors in the dataset, transforming the data to make it suitable for analysis, and saving the final cleansed dataset.

This report further documents the specific steps taken during the data wrangling process and the rationale behind it. It also outlines challenges faced and the learning outcome from these experiences.

# **Extract Transform Load (ETL) steps**

## **Extract**

* The data was read from the 4 CSV files using the CSV reader node.

### Dataset Examination

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **MALL N (13672 values)** | | | Field Name | Data Type | | customer\_id | string | | name | string | | gender | string | | birth\_date | string | | quantity | number(integer) | | price | number (double) | | payment\_method | string | | |  |  | | --- | --- | | **MALL E (11485 values)** | | | Field Name | Data Type | | customer\_id | string | | name | string | | gender | string | | birth\_date | string | | quantity | number(integer) | | price | number (double) | | payment\_method | string | |
| |  |  | | --- | --- | | **MALL W (11342 values)** | | | Field Name | Data Type | | customer\_id | string | | first\_name | string | | last\_name | string | | gender | string | | birth\_date | string | | quantity | number(integer) | | price | number (double) | | payment\_method | string | | |  |  | | --- | --- | | **MALL S (13672 values)** | | | Field Name | Data Type | | customer\_id | string | | name | string | | gender | string | | birth\_date | string | | quantity | number(integer) | | price | number (double) | | payment\_method | string | |

### Data Quality Inspection

|  |  |  |
| --- | --- | --- |
| **Mall** | **Schema** | **Initial Observations** |
| Mall N | Correct schema | * The lowest value for the price of an item is 0, indicating potential errors or promotions. |
| Mall E | Correct schema | * The lower bound value of quantity is 0, indicating instances where no units were bought. |
| Mall W | Incorrect schema | * The dataset contains separate 'first name' and 'last name' columns, differing from the standard schema. |
| Mall S | Correct schema | * Payment methods are listed as "Credit" and "Debit" instead of the standard "Credit Card" and "Debit Card" * Mall S also has 2 different date formats in the birth\_date column. |

## **Transform**

### Data Cleansing (Pre-Merging Transformations)

* **Mall N – Price value anomaly:** In the dataset, there were instances where price of an item was $0, which could be due to some promotional offerings. However, these transactions do not contribute to the overall sales analysis, so it was decided to remove these records.
* **Mall E – Quantity anomaly:** Similar to the previous error, there were records where quantity had the value 0. These records were most likely errors, so they were removed from the dataset.
* **Mall S – Payment Method Standardization:** Originally this dataset had the values Cash, Credit, and Debit as the payment method values. But this was not standardized with the other dataset values which used Credit Card and Debit Card.For the sake of uniformity, these values were transformed from Credit and Debit to Credit Card and Debit Card respectively. For this, the rule engine node was used.

### Schema Unification (Pre-Merging Transformations)

* **Mall W – Column Merge:** In this dataset there were 2 columns called first name and last name whereas other datasets only had a single column for name. To enable consistent dataset structure for subsequent operations, “first name” and “last name” were combined into a single column called “name”. The column combiner node was used to combine these columns together.

### Data Enrichment (Pre-Merging Transformations)

* **Mall Identifier:** The dataset was enriched by providing a feature called “mall\_name” that identifies the mall from which customers made their purchase. This additional column is important achieve our final analytical goal which is to find out the total sales per mall.

### Data Formatting (Pre-Merging Transformations)

* **Mall S - Standardizing Birth Date Format:** To maintain data consistency across all datasets, it was necessary to standardize the birth date format in the Mall S dataset which contained 'yy-mm-dd' and 'dd/mm/yyyy' formats. A multi-step process to convert the latter into the former, which was the standard format used in other mall datasets. The process involved:
* Row Splitter node to filter out the non-standard date format.
* Cell Splitter node to divide the date elements by "/".
* String Manipulation to reformat the date into the 'yy-mm-dd' standard.
* Then birth date list generated by the Cell Splitter node was removed from the dataset.
* Finally, the rows were combined back together using the concatenate node.

### Data Merging

* **Data Merging:** All the datasets were appended using the concatenate node. Additional input ports were incorporated to merge the four datasets together.

### Data Cleansing (Post-Merging Transformations)

* **Null values in birth date:** There are seven null values present in the birth date column. Given that this column isn't critical for the current analysis, these values can be ignored. However, if future analysis requires an understanding of spending based on age demographics, these fields could be filled using median or mode imputation. For the purposes of this assignment, the null values have been left as is.

### Data Enrichment (Post-Merging Transformations)

* **Total spending Calculation:** Total price feature is created and is derived from finding the product of the existing “quantity” and “price” columns. By doing this, a more direct measure of customer spending is created.

### Data Formatting (Post-Merging Transformations)

* **Converting birth date from string to date datatype:** Although the date format was now standardized across all datasets, the data type was a string instead of a date data type. This is problematic as it prevents us from performing sorting operations or date calculations on the date column. To convert this into a date data type, a series of steps were applied:
* The dates were divided into a list containing days, months, and years using the cell splitter node.
* Next, the month was padded with a leading zero if it was a single digit. For instance, if the month was "7", it was changed to "07". This task could not be accomplished using the string manipulation node as it lacked an if-else logic. Additionally, the rule engine couldn't be used as it didn't have a join logic. Therefore, the KNIME expressions extension, had to be installed and the column expressions node was utilized instead.
* Subsequently, a string manipulation node was used to reassemble the dates, replacing the existing birth date column with the date that now included a leading zero in the month.
* Following this, the birth\_date column was converted to a date datatype. However, a problem arose as the year only contained the last two digits, leading the date to assume all the years were from the current century. For instance, "99-03-12" was interpreted as "2099-03-12" rather than "1999-03-12".
* To fix this issue, the rule-based row splitter node was used to filter out dates greater than the current date (2023-06-02). Then, using the date and time shift node, these dates were rolled back by 100 years. The result was then concatenated back into the column.

### Data Anonymization (Post-Merging Transformations)

* **Customer ID masking:** The customer id was masked using the string manipulation node. The last 3 digits and an alphabet remained, and this was following PDPC guidelines.
* **Anonymization of name:** The anonymization of the name was executed using an anonymization node, employing a random seed value of 120 as the salt. A random salt was favoured over ‘salting using columns’ as it reduces the likelihood of decoding through column information. The anonymization approach also retains the ability to reverse the process to comply if any legal obligations. However, it's worth noting that securing the mapping table or key is extremely important to avoid potential re-identification.

## **Load**

### Final Adjustments

* **Row\_id reset:** The row id was reset to ensure that each record carries a unique and sequential identifier. This was achieved using the Row ID node.

### Data Storage

* The cleaned dataset was stored in the “Cleaned\_CustomerSpendingData\_NanyangLand.csv” csv file.

# **Reflection**

Reflecting on this project, it was definitely challenging but fulfilling. The task of managing real-world data was not easy, given its complexities and inconsistencies. But the experience was worthwhile, allowing me to learn valuable insights about data handling.

One of the primary challenges I encountered was identifying and fixing errors within a large dataset, particularly when these issues weren't readily apparent. In an attempt to address this, I utilized summary statistics to find abnormal or outlier values, and I examined every unique value in each column to identify any suspicious entries. While this method was successful in resolving many issues, it highlighted to me the potential advantages of utilizing more advanced techniques such as machine learning in future.

The data preparation stage had its own set of challenges. KNIME provided a lot of useful tools for pre-processing the data, but sometimes it was overwhelming with so many options to choose from. There were times when I had to search the KNIME community forum to figure out how to use certain features or how to write the correct expressions.

I also hit a roadblock in trying to standardize the date format in the Mall S dataset and converting the string to a date format. I had to go through a complex multi-step process using different tools in KNIME to get this done. This was a big learning for me, showing the importance of having good strategies for dealing with these kinds of data inconsistencies.

Once the datasets were merged, I was faced with a decision on how to address the null values in the birth date column. For the purposes of this project, I chose to leave them as is. However, this situation made me think about the potential consequences of such data, particularly if future analyses were to require age-based insights. This led me to explore potential imputation strategies, such as using median or mode values, to deal with missing values.

An important part of the project was also making sure to protect customer data. I learned a lot about data privacy laws and how to properly anonymize data. Previously I had learned about the theory behind these techniques from my data privacy and protection module, so this was a good opportunity to get hands on with these techniques.

In conclusion, this project was a great learning experience for me. It was challenging, but it helped me improve my skills in data processing and problem-solving. I know there's still a lot to learn, but I'm excited to use what I've learned in this project in the future.

# 

# **Citations**

1. <https://hub.knime.com/knime/extensions/org.knime.features.base/latest/org.knime.base.node.preproc.colcombine2.ColCombine2NodeFactory>
2. [https://hub.knime.com/kathrin/spaces/Forum Questions/latest/Convert Date&Time~CfajlONe6eCRHiN3](https://hub.knime.com/kathrin/spaces/Forum%20Questions/latest/Convert%20Date&Time~CfajlONe6eCRHiN3)
3. https://hub.knime.com/knime/extensions/org.knime.features.expressions/latest
4. <https://forum.knime.com/t/data-with-mixed-date-format/29777/2>
5. <https://forum.knime.com/t/how-to-combine-2-columns-as-one-column/62330/6>
6. <https://forum.knime.com/t/mess-with-datetime-format/34534/14?page=2>
7. <https://www.pdpc.gov.sg/-/media/Files/PDPC/PDF-Files/Other-Guides/Technical-Guide-to-Advisory-Guidelines-on-NRIC-Numbers---260819.pdf>
8. https://nodepit.com/node/org.knime.base.node.preproc.rowkey2.RowKeyNodeFactory2