**Data Cleaning Documentation**

|  |  |
| --- | --- |
| **Project Details** | |
| Client | Retail Company |
| **Project** | Retail Company Sales Performance Analysis |
| **Start Data** | 2023-10-26 |
| **End Date** | 2023-11-25 |
| **Email** | [Alaamhassan2001@gmail.com](mailto:Alaamhassan2001@gmail.com) |
| **GitHub** | <https://github.com/alaamhassan> |
| **LinkedIn** | [www.linkedin.com/in/alaamhassan](http://www.linkedin.com/in/alaamhassan) |
| **Portfolio** |  |

Table of Contents

[Metadata 3](#_Toc152697241)

[Changelog 3](#_Toc152697242)

[Cleaning Process 5](#_Toc152697243)

[1. Business Logic (does the data make sense?) 5](#_Toc152697244)

[2. Check for Duplicates 8](#_Toc152697245)

[3. Check for Mistyping 10](#_Toc152697246)

[4. Check for Blank Fields 12](#_Toc152697247)

# Metadata

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| **Consumer ID** | Text | Unique identifier for each customer. |
| **Order ID** | Text | Unique identifier for each order, where the first part is the Consumer ID and the second is a unique alphabetic char. |
| **Month** | Text | The month when the order was placed. |
| **Year** | Text | The year when the order was placed. |
| **Total order value** | Number | The total value of the order before any discounts are applied. |
| **Discount** | Number | The amount of discount applied to the order. |
| **Line Value (net discount)** | Number | The value of the order after the discount has been applied. |
| **Line Category** | Text | The category of the product in the order. |
| **Line SKU** | Text | Unique identifier for each product. |
| **Line SKU Production Cost** | Number | The production cost of the product. |

# Changelog

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Change ID** | **Reason** | **Title of Change** | **Description** | **# Values** | **Process**  **Link** | **Version**  **Link** |
| **C1** | Data constraint violation. | Fixing mistyping of ‘Line SKU Production Cost’ Value. | Change Line SKU Production Cost from 432,000 € to 43.2 €. | 1 | [C1-P](#C1_P) | [C1-V](https://github.com/alaamhassan/RetailCompany_PerformanceAnalysis/tree/44bc2d7178ee4e43471d926f714bceb54e3c5ad2) |
| **C2** | Data Mistyping. | Fixing mistyping of ‘Line Category’ value. | Change ‘Line Category’ from Mini bag to Mini bags. | 3 | [C2-P](#C2_P) | [C2-V](https://github.com/alaamhassan/RetailCompany_PerformanceAnalysis/tree/4775555da2323c1b3277ec8e2369a74b49d53c44) |
| **C3** | Blank Data. | Filling Values for ‘Order ID’. | Fill Blank Fields to contain the ‘Consumer ID’ plus the two characters I and J. | 2 | [C3-P](#C3_P) | [C3-V](https://github.com/alaamhassan/RetailCompany_PerformanceAnalysis/tree/68a71176357a0a3d57d6c4cf6864acbff6a37464) |
| **C4** | Blank Data. | Filling Values for ‘Line SKU Production Cost’. | Fill Blank Field of ‘MIN3’ Line SKU to contain 247.26 €. | 1 | [C4-P](#C4_P) | C4-V |
| **C5** | Blank Data. | Filling Values for ‘Line SKU Production Cost’. | Fill Blank Field of ‘MIN4’ Line SKU to contain 293.22 €. | 1 | [C5-P](#C5_P) | C5-V |
| **C6** | Blank Data. | Filling Values for ‘Line SKU Production Cost’. | Fill Blank Field of ‘MIN2’ Line SKU to contain 222.25 €. | 1 | [C6-P](#C6_P) | C6-V |
| **C7** | Blank Data. | Filling Values for ‘Line SKU Production Cost’. | Fill Blank Field of ‘MIN8’ Line SKU to contain 210.4 €. | 1 | [C7-P](#C7_P) | C7-V |
| **C8** | Mismatched Data Type. | Converting the data type of “Consumer ID”. | Convert the data type of “Consumer ID” from General to Text. | 15 | [C8-P](#C8_P) | C8-V |

# Cleaning Process

## Business Logic (does the data make sense?)

Constrains based on data:

* Order ID = Consumer ID + Alphabetic Char
* Total order value >= Discount
* Line Value (net discount) = Total order value – Discount
* Line SKU Production Cost < Line Value (net discount)

Each constrain was checked using **conditional formatting**:

1. Order ID = Consumer ID + Alphabetic Char

To check for this constraint:

1. Add a new column called ‘Order ID - First Part’.
2. Extract the first part of the Order ID using both the **LEFT** and **LEN** function.
3. A screenshot of a computer

   Description automatically generatedUse **conditional formatting** to check the constraint.

**Output**:

There are 15 values highlighted as a violation, although they don’t seem as such.

A screenshot of a computer

Description automatically generated

**Observation**:

A further investigation has found that the highlighted values are because of different data types rather than violating the data constraint.

* A screenshot of a computer

  Description automatically generated“Consumer ID” that didn’t trigger a highlight in the “Order ID – First Part” was stored as text.
* A screenshot of a computer

  Description automatically generated“Consumer ID” that did trigger a highlight in the “Order ID – First Part” wasn’t stored as text.

**Correctness**:

**Steps**:

1. A screenshot of a computer

   Description automatically generatedA screenshot of a computer

   Description automatically generatedA screenshot of a computer

   Description automatically generatedConvert the data type of the “Consumer ID” from General to Text using **Text to Columns** feature.

**NOTE:**

* The Text to Columnsfeature was used rather than converting the type simply from the Data tap, because it would require re-entering the data again.
* A screenshot of a computer

  Description automatically generatedThis change occurred as the 8th change.

2) Total order value >= Discount

A screenshot of a computer

Description automatically generated

**Output**:

No value violates the constraint.

3) Line Value (net discount) = Total order value – Discount

A screenshot of a computer

Description automatically generated

**Output**:

No value violates the constraint.

A screenshot of a computer

Description automatically generated4) Line SKU Production Cost < Line Value (net discount)

**Output**:  
one value violates the constraint



**Correctness**:

**Steps**:

1. Filter for the ‘FRA5’ Line SKU

A screenshot of a computer

Description automatically generated

**Observations**:

170 records have the category ‘FRA5’. All these records have a Line SKU Production Cost of 43.2 €.

1. Change the Line SKU Production Cost of the first record from 432,000 € to 43.2 €.

A screenshot of a computer

Description automatically generated

## Check for Duplicates

Unique Constraints:

* A Record can’t be duplicated.
* Order ID can’t appear twice.

Check for each constrain using **Remove Duplicates** and **Conditional Formatting**:

1. A Record can’t be duplicated.

A screenshot of a computer

Description automatically generateduse **Remove Duplicates** to find and remove duplicate records.

**Output**:

A screenshot of a computer error

Description automatically generated No record violates the constraint.

1. Order ID can’t appear twice.

Check for duplicates using **conditional formatting**:

A screenshot of a computer

Description automatically generated

**Output**:

Two values violate the constraint.



as the value of the Order ID is N/A, no values will be removed.

## Check for Mistyping

In the ‘Line Category’ column there are two categories with the same name:

* Mini bags
* Mini bag

A screenshot of a computer

Description automatically generatedbut one is plural and the other is singular.

**Correctness:**

As the two ‘Line Category’ have records with the same ‘Line SKU’. Then the two are likely the same.

**steps:**

1. Filter for the ‘Mini bag’ Line Category.
2. A screenshot of a computer

   Description automatically generatedChange ‘Mini bag’ to ‘Mini bags’ using **Find and Replace**.

**Output**:

A screenshot of a computer error

Description automatically generatedThree values were changed.

1. Validate using the Filter Menu.

A screenshot of a computer

Description automatically generated

## 4. Check for Blank Fields

There are blanks fields in:

* Order ID
* Line SKU Production Cost

Let’s investigate each column:

1. Order ID

Use Filtration to find Blank fields.

**Correctness**:

**Steps:**

1. A screenshot of a computer

   Description automatically generatedAs the blanks field were for the ‘Consumer ID’ => 13134019, Filter only this one.

1. A screenshot of a computer

   Description automatically generatedChange the two blank fields to contain the Consumer ID plus the two characters I and J.
2. Line SKU Production Cost

Use Filtration to find Blank fields.

**Correctness**:

**Steps:**

1. Filter for ‘Line SKU’ equal MIN3.

A screenshot of a computer

Description automatically generated

**Observations**:

27 records have the Line SKU ‘MIN3’. All these records have a Line SKU Production Cost of 247.26 €.

A screenshot of a computer

Description automatically generatedFill the ‘Line SKU Production Cost’ of the ‘MIN3’ to 247.26 €.

1. A screenshot of a computer

   Description automatically generatedFilter for ‘Line SKU’ equal MIN4.

**Observations**:

33 records have the Line SKU ‘MIN4’. All these records have a Line SKU Production Cost of 293.22 €.

Fill the ‘Line SKU Production Cost’ of the ‘MIN4’ to 293.22 €.

A screenshot of a computer

Description automatically generated

1. A screenshot of a computer

   Description automatically generatedFilter for ‘Line SKU’ equal MIN2.

**Observations**:

29 records have the Line SKU ‘MIN2’. All these records have a Line SKU Production Cost of 222.25 €.

A screenshot of a computer

Description automatically generatedFill the ‘Line SKU Production Cost’ of the ‘MIN2’ to 222.25 €.

1. A screenshot of a computer

   Description automatically generatedFilter for ‘Line SKU’ equal MIN8.

**Observations**:

17 records have the Line SKU ‘MIN8’. All these records have a Line SKU Production Cost of 210.4 €.

A screenshot of a computer

Description automatically generatedFill the ‘Line SKU Production Cost’ of the ‘MIN8’ to 210.4 €.

## Check for mismatched data types

From the Metadata:

* Consumer ID => Text