NWU®

CMPG 321

GROUP GP10

Deliverable 2A: Physical Build and Implementation

Sponsor:	J. Pretorius
Executive Sponsor:	Mrs. J. Thandi

Group Members:

Jacques van Heerden (Student Number: 35317906)

Rohann Venter (Student Number: 25130757)

Francois Verster (Student Number: 40723380)

Christo Prinsloo (Student Number: 21052239)

Erika Haasbroek (Student Number: 37673149)

Contents

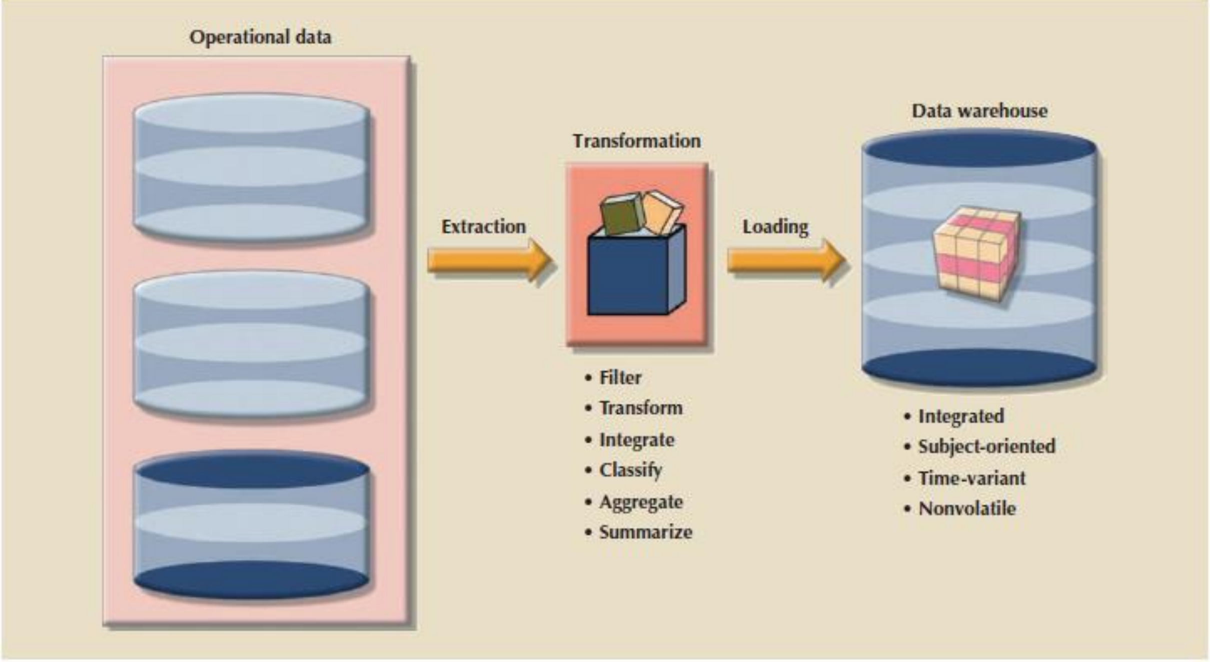
Executive Summary

Concise overview of the ETL process purpose and its role in the NoSQL-based BI system. (10 marks)

ETL Process Design

Detailed design of the ETL process, including extraction from provided data files, transformation rules (e.g., aligning with financial year), and loading into a NoSQL database (e.g., MongoDB). (20 marks)

FIGURE 13.4 THE ETL PROCESS



1. Extraction

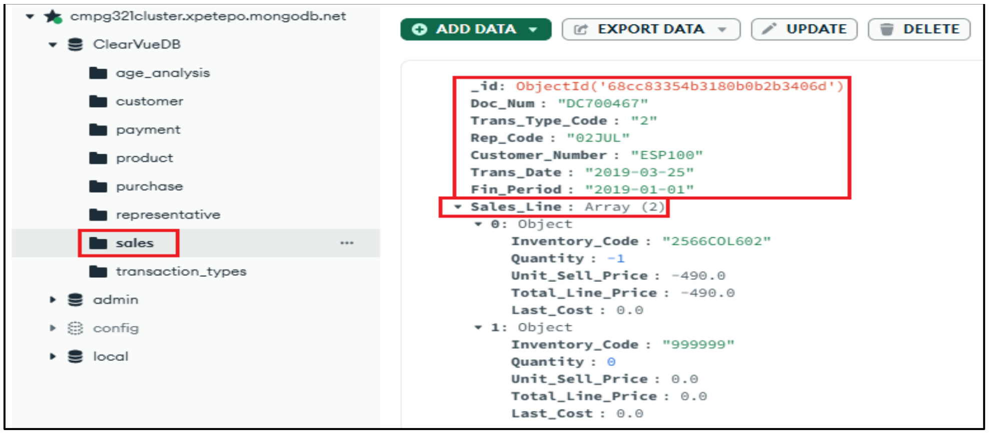
In the extraction phase, Python scripts utilizing the Pandas library were developed to collect data from 19 CSV files, including customer, payment, and product brand datasets. Validation rules were applied to ensure data quality, such as checking for missing values, duplicate records, and adherence to the expected schema, prior to loading into MongoDB. This approach ensures the system's flexibility to support hierarchical data (e.g., product categories) and scalability for future supplier analytics, offering real-time insights and a star schema-compatible design for efficient querying, surpassing the limitations of traditional relational databases.

2. Transformation

a. Aggregation 1: Sales Collection

i. Sales Header

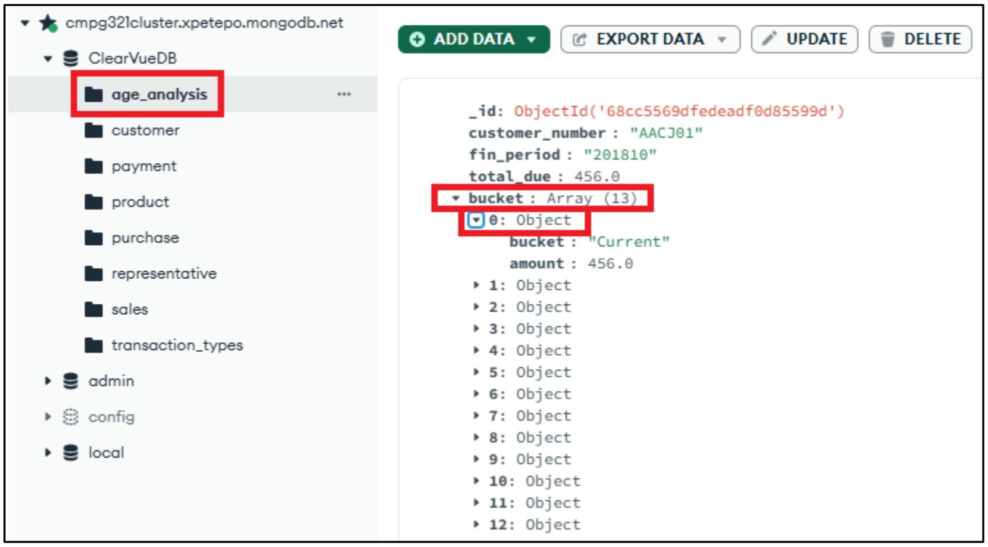
ii. Sales Line



```
_id: ObjectId('68cc83354b3180b0b2b3406d')
Doc_Num : "DC700467"
Trans_Type_Code : "2"
Rep_Code : "02JUL"
Customer_Number : "ESP100"
Trans_Date : "2019-03-25"
Fin_Period : "2019-01-01"
Sales_Line : Array (2)
  0: Object
    Inventory_Code : "2566COL602"
    Quantity : -1
    Unit_Sell_Price : -490.0
    Total_Line_Price : -490.0
    Last_Cost : 0.0
  1: Object
    Inventory_Code : "999999"
    Quantity : 0
    Unit_Sell_Price : 0.0
    Total_Line_Price : 0.0
    Last_Cost : 0.0
```

b. Aggregation 2: Age Analysis Collection

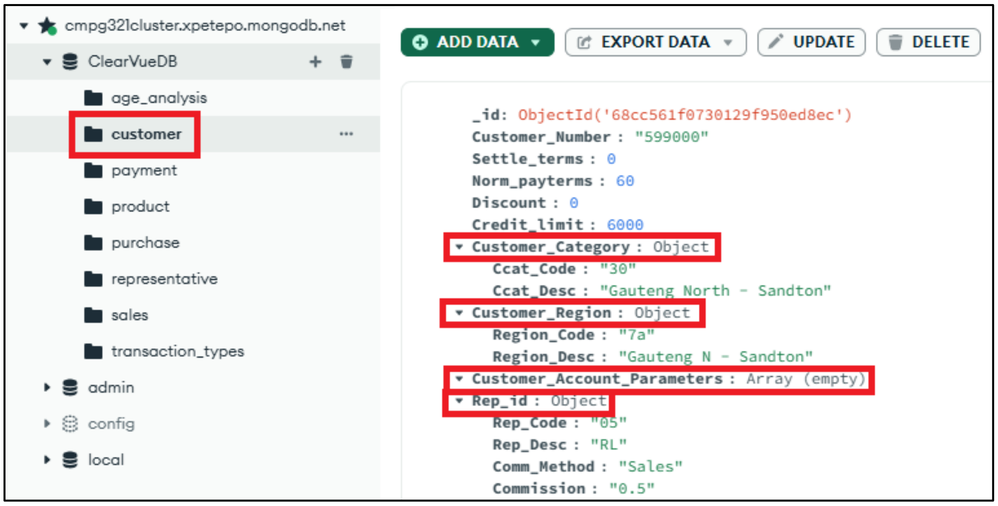
i. Age Analysis



```
_id: ObjectId('68cc5569dfedeadf0d85599d')
customer_number : "AACJ01"
fin_period : "201810"
total_due : 456.0
bucket : Array (13)
  0: Object
    bucket : "Current"
    amount : 456.0
  1: Object
  2: Object
  3: Object
  4: Object
  5: Object
  6: Object
  7: Object
  8: Object
  9: Object
  10: Object
  11: Object
  12: Object
```

c. Aggregation 3: Customer Collection

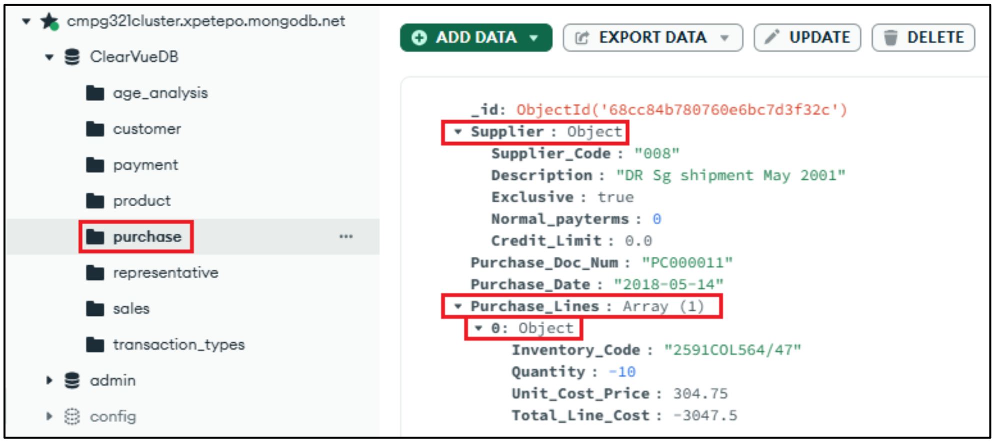
i. Customer
ii. Customer Categories
iii. Customer Regions
iv. Representative



```
_id: ObjectId('68cc561f0730129f950ed8ec')
Customer_Number : "599000"
Settle_terms : 0
Norm_payterms : 60
Discount : 0
Credit_limit : 6000
▾ Customer_Category : Object
   Ccat_Code : "30"
   Ccat_Desc : "Gauteng North - Sandton"
▾ Customer_Region : Object
   Region_Code : "7a"
   Region_Desc : "Gauteng N - Sandton"
▾ Customer_Account_Parameters : Array (empty)
▾ Rep_id : Object
   Rep_Code : "05"
   Rep_Desc : "RL"
   Comm_Method : "Sales"
   Commission : "0.5"
```

d. Aggregation 4: Purchase Collection

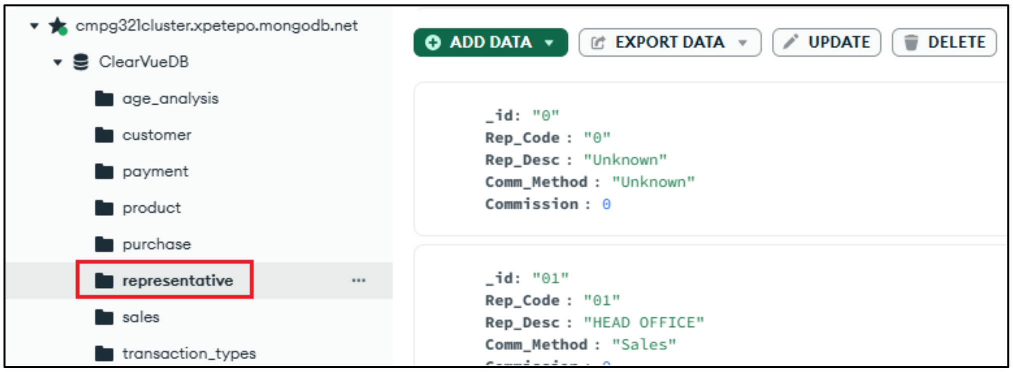
i. Suppliers
ii. Purchase Headers
iii. Purchase Lines



```
_id: ObjectId('68cc84b780760e6bc7d3f32c')
▾ Supplier : Object
   Supplier_Code : "008"
   Description : "DR Sg shipment May 2001"
   Exclusive : true
   Normal_payterms : 0
   Credit_Limit : 0.0
Purchase_Doc_Num : "PC000011"
Purchase_Date : "2018-05-14"
▾ Purchase_Lines : Array (1)
  ▾ 0: Object
     Inventory_Code : "2591COL564/47"
     Quantity : -10
     Unit_Cost_Price : 304.75
     Total_Line_Cost : -3047.5
```

e. Aggregation 5: Representative Collection

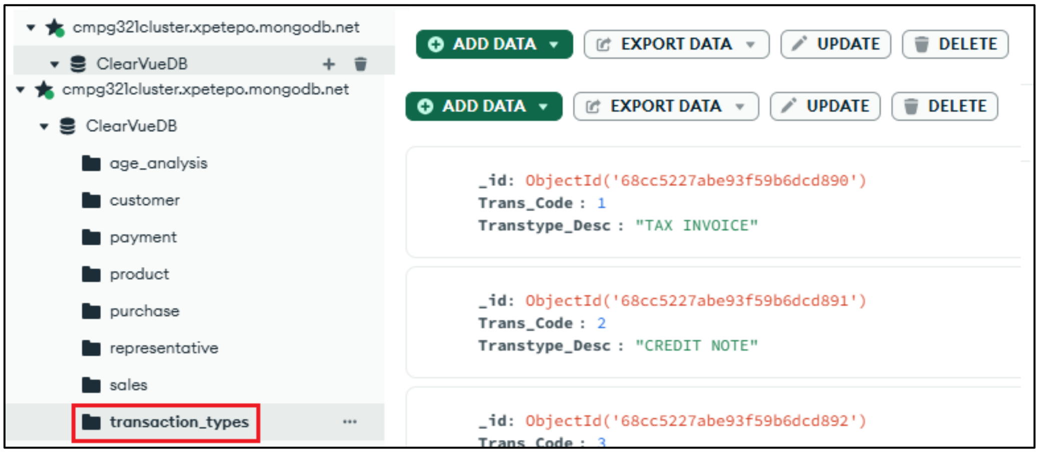
i. Representative



```
_id: "0"
Rep_Code : "0"
Rep_Desc : "Unknown"
Comm_Method : "Unknown"
Commission : 0

_id: "01"
Rep_Code : "01"
Rep_Desc : "HEAD OFFICE"
Comm_Method : "Sales"
```

f. Aggregation 6: Transaction Types Collection

i. Trans types

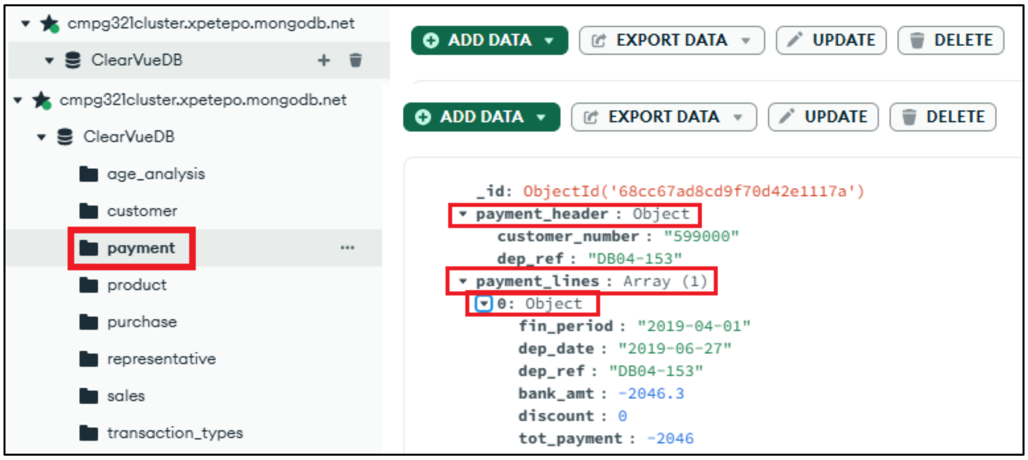


```
_id: ObjectId('68cc5227abe93f59b6dcd890')
Trans_Code : 1
Transtype_Desc : "TAX INVOICE"

_id: ObjectId('68cc5227abe93f59b6dcd891')
Trans_Code : 2
Transtype_Desc : "CREDIT NOTE"

_id: ObjectId('68cc5227abe93f59b6dcd892')
Trans_Code : 3
```

g. Aggregation 7: Payment Collection

i. Payment Headers

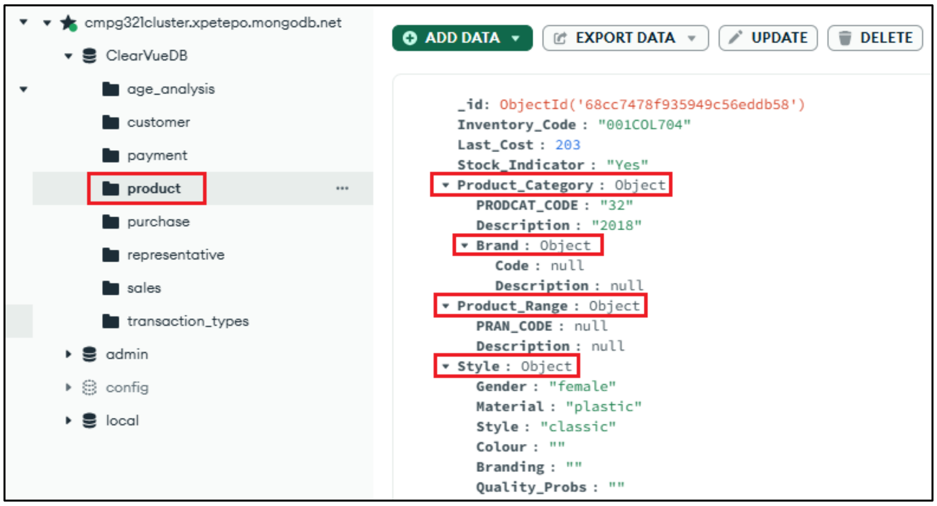
ii. Payment Lines



```
_id: ObjectId('68cc67ad8cd9f70d42e1117a')
▾ payment_header : Object
    customer_number : "599000"
    dep_ref : "DB04-153"
▾ payment_lines : Array (1)
  ▾ 0: Object
      fin_period : "2019-04-01"
      dep_date : "2019-06-27"
      dep_ref : "DB04-153"
      bank_amt : -2046.3
      discount : 0
      tot_payment : -2046
```

h. **Aggregation 8: Product Collection**

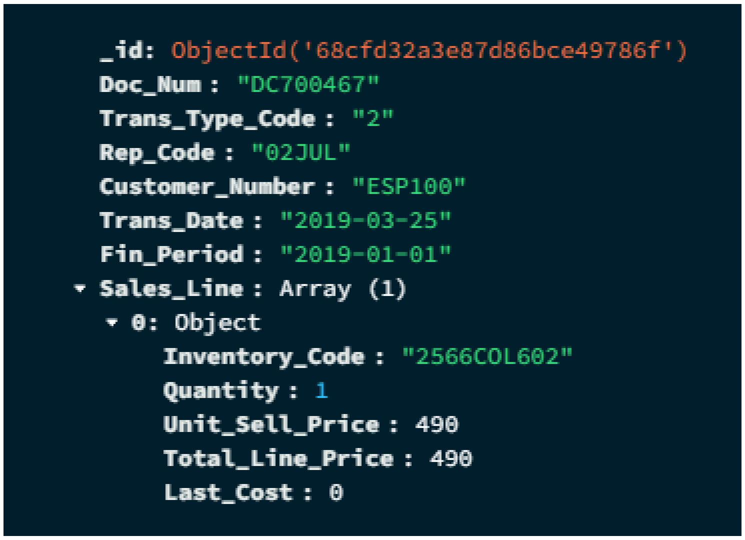
   i. Products
   ii. Products Styles
   iii. Product Range
   iv. Product Categories
   v. Product Brands



```
_id: ObjectId('68cc7478f935949c56eddb58')
Inventory_Code : "001COL704"
Last_Cost : 203
Stock_Indicator : "Yes"
▾ Product_Category : Object
    PRODCAT_CODE : "32"
    Description : "2018"
  ▾ Brand : Object
      Code : null
      Description : null
▾ Product_Range : Object
    PRAN_CODE : null
    Description : null
▾ Style : Object
    Gender : "female"
    Material : "plastic"
    Style : "classic"
    Colour : ""
    Branding : ""
    Quality_Probs : ""
```

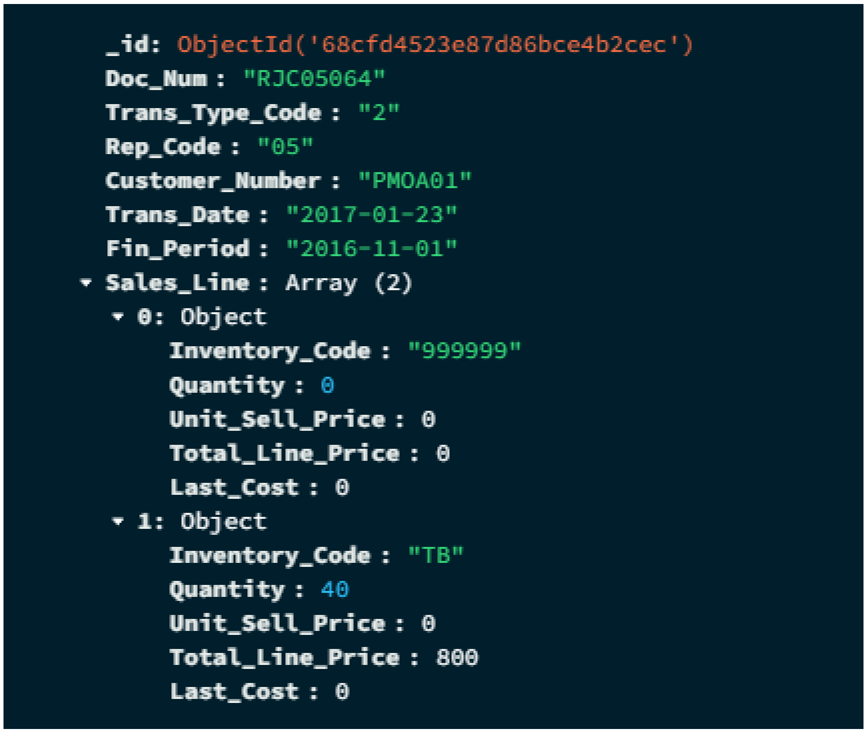
Francois Cleaning:

Only sales data with negative QUANTITY, UNIT_SELL_PRICE, TOTAL_LINE_PRICE and LAST_COST values were converted to positive values. Sales data with an empty value in QUANTITY or UNIT_SELL_PRICE or TOTAL_LINE_PRICE were removed from the sales collection and stored into a separate collection called SalesIncomplete. These transaction details are kept for data integrity reason, even if there are no inventory transactions connected to the sales transaction (due to removal).



```
    _id: ObjectId('68cfd32a3e87d86bce49786f')
    Doc_Num : "DC700467"
    Trans_Type_Code : "2"
    Rep_Code : "02JUL"
    Customer_Number : "ESP100"
    Trans_Date : "2019-03-25"
    Fin_Period : "2019-01-01"
  ▾ Sales_Line : Array (1)
    ▾ 0: Object
        Inventory_Code : "2566COL602"
        Quantity : 1
        Unit_Sell_Price : 490
        Total_Line_Price : 490
        Last_Cost : 0
```

Example of complete sales collection object.



```
    _id: ObjectId('68cfd4523e87d86bce4b2cec')
    Doc_Num : "RJC05064"
    Trans_Type_Code : "2"
    Rep_Code : "05"
    Customer_Number : "PMOA01"
    Trans_Date : "2017-01-23"
    Fin_Period : "2016-11-01"
  ▾ Sales_Line : Array (2)
    ▾ 0: Object
        Inventory_Code : "999999"
        Quantity : 0
        Unit_Sell_Price : 0
        Total_Line_Price : 0
        Last_Cost : 0
    ▾ 1: Object
        Inventory_Code : "TB"
        Quantity : 40
        Unit_Sell_Price : 0
        Total_Line_Price : 800
        Last_Cost : 0
```

Examples of incomplete inventory transactions being stored in the SalesIncomplete collection object.

3. Loading

The final step involved loading the transformed and cleaned data into MongoDB, a NoSQL database serving as the target destination for storing the structured and integrated dataset. This was achieved by implementing Python scripts that utilized libraries such as Pymongo and Pandas to read the processed CSV files and insert the records as documents into designated collections. This approach ensured efficient, scalable ingestion of the data, making it readily accessible for subsequent analytics, reporting, and machine learning app/locations.

Evaluation of ETL Process

Explicit evaluation of the ETL process in relation to the 17 provided data files, addressing data structures, hierarchies, and challenges (e.g., data quality, volume, variety). (20 marks)

1. **Extraction**
2. **Transformation**
   a. Data Cleaning
      i. Excel spreadsheet cleanup
      We had a look through all the Excel spreadsheets to look for any data that might impact performance negatively, i.e. sales transactions with negative values impact sales performance negatively, incorrect assigning of deposit references to customers impact deposits made by each customer, etc.

      1. No cleanup needed

      - Age Analysis
      - Customer Account Parameters
      - Customer
      - Product Brands
      - Products Styles
      - Products
      - Purchases Headers
      - Suppliers
      - Trans Types

2. **Cleanup needed**

- Customer Categories
    i. Category codes with unknown descriptions
    ii. Multiple category codes have descriptions with “No” or “no”
    iii. Category codes with “?” descriptions
- Customer Regions
    i. Multiple regions descriptions cover a huge geographic area, some covering cities in different provinces.
- Payment Header
    i. Inconsistent naming on deposit references, multiple have dates or different style of referencing
    ii. Multiple customers have the same deposit reference
- Payment Lines
    i. Multiple customers have the same deposit references.
- Product Categories
    i. The same name is used for different category codes
- Product Ranges
    i. The same description is used for different range codes
- Purchases Lines
    i. Negative values present in the “Quantity” columns which contribute to negative purchase totals
- Representatives
    i. The same representative has two different codes
    ii. Multiple codes have the same description
- Sales Header
    i. The financial period must be adjusted to the unique financial month rule that ClearVue Ltd has. For instance, the 2016/2017 financial year ends on 24 February 2017, and the 2017/2018 financial year starts on 25 February 2017. However, the data indicates that all transactions after 24 February 2017 falls under the 2016/2017 financial year.
- Sales Line
    i. Quantity column contains negative or “0” values
    ii. Inconsistent calculations in the total column:
        - ➢ Total Sales amount as per documents provided: R73 120 335
        - ➢ Total Sales amount as per corrections to negative values (changed from negative to positive): R105 922 736
        - ➢ Invalid values in Quantity, Sell Price, and Total columns:

a. Quantity column has 55882 entries containing “0”
b. Sell Price column has 79080 entries containing “0”
c. Total column has 78402 entries containing “0”

b. **Financial year alignment**

i. The unique financial year rule that ClearVue Ltd has is - a financial month starts on the last Saturday of the preceding month and ends on the last Friday of the current month.

ii. According to their files their financial year starts in March and ends the next year in February. To align the custom financial year with the unique rule the financial year will start on the last Saturday of February of the current year and run until the last Friday of February the next year.

Alignment with Financial Year

Clear demonstration of how the ETL process handles ClearVue Ltd.’s unique financial year structure (e.g., financial month from last Saturday of preceding month to last Friday of current month). (15 marks)

Prototype Implementation Plan

Detailed plan for implementing the ETL process in a prototype, including tools, technologies, and testing strategies. (15 marks)

Justification of NoSQL Approach

Clear justification of how the ETL process leverages NoSQL advantages (e.g., flexibility, scalability) over relational systems for the given data. (10 marks)

AI Usage Log

Comprehensive documentation of AI prompts/outputs used for ETL design or evaluation, with explanations of validations or improvements. (10 marks)

Overall Presentation and Clarity

Structure, clarity, and professionalism of the report (e.g., grammar, formatting, logical flow). (5 marks)

References