AUTOMATED DATA CLEANING PIPELINE

1.Objective

The objective of this task is to design and implement an automated Python-based data cleaning pipeline that detects and corrects common data quality issues such as missing values, outliers, and inconsistent formats in a raw CSV dataset. The solution emphasizes **au**tomation, reproducibility, and scalability, while using OpenRefine as a validation reference.

2.Tools and Technologies

- Python
- Pandas
- NumPy
- OpenRefine
- Jupyter Notebook

3.Dataset Overview

The raw dataset contained several data quality issues, including:

- Missing values in numeric and categorical columns
- Inconsistent text formatting

- Presence of outliers in numeric attributes (e.g., salary and age)

- Potential anomalous records affecting data reliability

  The raw dataset is attached in submission for reference.

4.OpenRefine Cleaning (Reference Approach)

OpenRefine was used for initial exploration and manual cleaning to understand the nature of data issues.

- OpenRefine was used for manual cleaning tasks such as whitespace trimming and text normalization. Missing value imputation was intentionally not performed in OpenRefine, as such operations are subjective and not reproducible in manual tools. Instead, statistical imputation was implemented in the automated Python pipeline.

Note:

The OpenRefine-cleaned dataset was not used as input to the pipeline. It was retained only as a benchmark to validate the effectiveness of the automated Python solution.

The OpenRefine cleaned dataset is attached in submission for reference.

5.Python Automated Cleaning Pipeline

An automated pipeline was developed to clean the raw dataset end-to-end. The pipeline consists of modular and reusable functions applied sequentially.

Pipeline Steps

- **Raw Data Preservation**
  A copy of the raw dataset is created to ensure the original data remains unchanged.

- **Missing Value Imputation**

  - Numeric columns are imputed using the median

  - Categorical columns are imputed using the mode

- **Text Normalization**

  - Conversion to lowercase

  - Removal of leading and trailing whitespaces

  - Standardization of inconsistent categorical values

- **Outlier Detection**

  - Outliers are detected using the Z-Score method

  - Outliers are recorded per numeric column

- **Anomaly Flagging**
    - Records containing extreme or inconsistent values are flagged using an `Anomaly` indicator column

This pipeline is fully automated and can be applied to any similar raw CSV dataset without manual intervention.

6.Before and After Analysis

1. Missing Values

- Significant reduction in missing values across all columns after pipeline execution
- Pipeline results closely match (and in some cases improve upon) OpenRefine cleaning

2. Numeric Summary Statistics

- Improved distribution of numeric features
- Reduced variance caused by extreme values
- More stable and realistic mean and median values

3. Outlier Detection

- Outliers detected and quantified per numeric column
- Enables downstream decisions such as removal, capping, or further investigation

4. Anomaly Flagging

- Anomalous records explicitly flagged
- Improves transparency and traceability of data quality issues

Visual comparisons boxplots and distribution plots clearly demonstrate improvements from **raw** → **pipeline-cleaned** data and alignment with OpenRefine results.

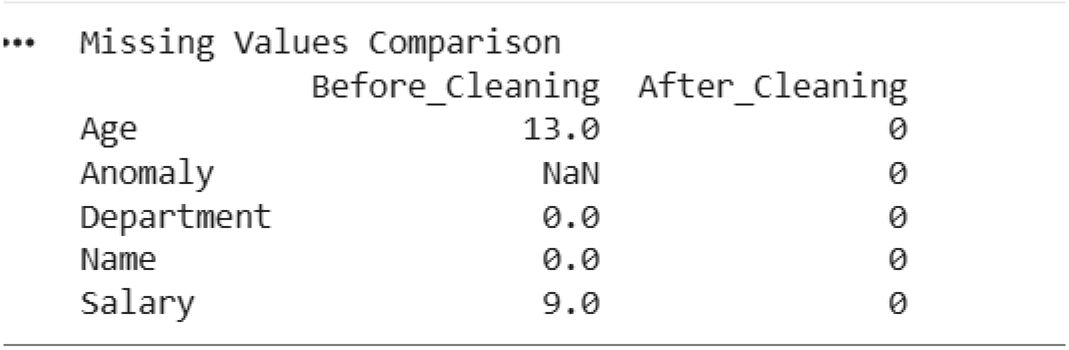
7.Results and Observations

- The Python pipeline successfully cleaned the raw dataset without manual intervention

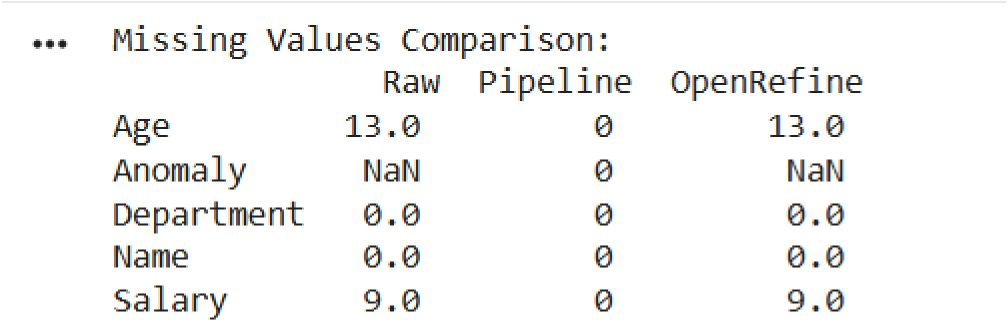
- Data quality improvements were measurable and consistent

- Pipeline performance matched or exceeded OpenRefine cleaning outcomes

- The solution is scalable, reproducible, and suitable for production workflows



```
•••  Missing Values Comparison
                Before_Cleaning  After_Cleaning
     Age                   13.0               0
     Anomaly                NaN               0
     Department             0.0               0
     Name                   0.0               0
     Salary                 9.0               0
```



```
•••  Missing Values Comparison:
                 Raw  Pipeline  OpenRefine
     Age        13.0         0        13.0
     Anomaly     NaN         0         NaN
     Department  0.0         0         0.0
     Name        0.0         0         0.0
     Salary      9.0         0         9.0
```

8.Conclusion

This project demonstrates the successful implementation of an automated data cleaning pipeline using Python, Pandas, and NumPy. While OpenRefine was useful for exploratory validation, the Python pipeline provides a fully automated, reusable, and scalable solution for real-world data preprocessing tasks. The before-and-after analysis confirms that the pipeline effectively improves data quality and is suitable for deployment in data engineering or machine learning workflows.