IS-5960-04: MRP

Employability Analytics Project

Data Validation & Transformation Documentation

Group Name: Team 16

Group Members:

Ananya Chowdary Bheemaneni

Maneesha Kakarla

Bala Krishna Kalavakunta

Laya Kalva

Manohar Kancharla

Sai Venkata Sriram Chowdary Karicheti

Tools Used:

- Python
- Power BI Power Query Editor
- DAX Measures

Python Data Cleaning Steps

Below are the Python scripts which we have executed for data cleaning.

Step 1: Load the Dataset

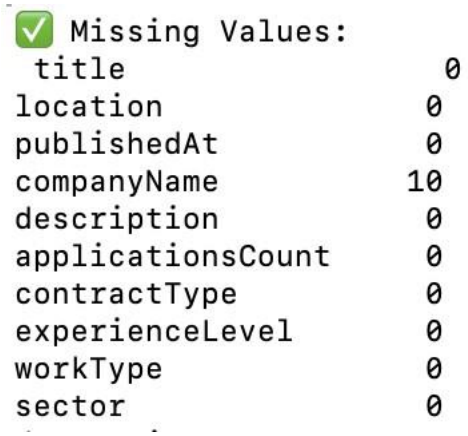
import pandas as pd file_path
=
"/Users/ananya/Documents/TEAM16_MRP/Business_analyst_job_listings_linkedin.csv"
df = pd.read_csv(file_path, dtype=str) **Output:**

```
✅ Dataset Loaded Successfully!
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 921 entries, 0 to 920
Data columns (total 10 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   title              921 non-null    object
 1   location           921 non-null    object
 2   publishedAt        921 non-null    object
 3   companyName        911 non-null    object
 4   description        921 non-null    object
 5   applicationsCount  921 non-null    object
 6   contractType       921 non-null    object
 7   experienceLevel    921 non-null    object
 8   workType           921 non-null    object
 9   sector             921 non-null    object
```

Step 2: Check for Missing Values Before

Cleaning: print("Missing Values:\n",

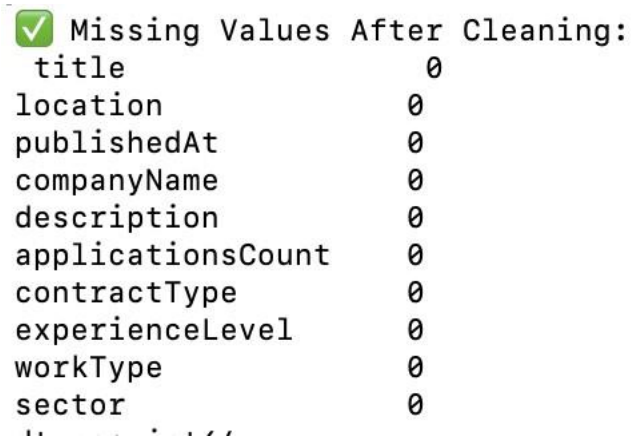
df.isnull().sum()) **Output:**



After Cleaning:

df["companyName"] = df["companyName"].fillna("Unknown") print("Missing Values After Cleaning:\n", df.isnull().sum())

Output:



Step 3: Convert applicationsCount to Numeric

import re def extract_number(val):

match = re.search(r'\d+', str(val)) return

int(match.group()) if match else None

df["applicationsCount"] = df["applicationsCount"].apply(extract_number)

Step 4: Convert publishedAt to Standard Date Format

df["publishedAt"] = pd.to_datetime(df["publishedAt"], format="%m/%d/%y", errors="coerce")

Output:

```
  publishedAt
0  2024-09-04
1  2024-08-23
2  2024-08-02
3  2024-08-20
4  2024-08-27
```

Step 5: Check and Fix Duplicate Job Listings Before fixing: duplicates =

df[df.duplicated(subset=["title", "location", "companyName"], keep=False)]

print("Duplicate Records Found:", len(duplicates))

Output:

```
... df.drop_duplicates(subset=["title", "location", "companyName"], keep="first", inplace=True)
[...
✅ Duplicate Records Found: 201
```

After fixing: df.drop_duplicates(subset=["title", "location", "companyName"], keep="first",

inplace=True) duplicates = df[df.duplicated(subset=["title", "location", "companyName"],

keep=False)] print("Duplicate Records Found:", len(duplicates)) **Output:**

```
0     200
1     200
2     170
3     200
4     200
5     200
6     200
8     200
9     200
10    200
Name: applicationsCount, dtype: int64
✅ Duplicate Records Found: 0
```

Step 6: Save the Cleaned Dataset cleaned_file_path

=

"/Users/ananya/Documents/TEAM16_MRP/CLEANED_Business_analyst_job_listings.csv"

df.to_csv(cleaned_file_path, index=False)

print(f"Cleaned dataset saved at: {cleaned_file_path}")

Transformations using Power BI (Power Query)

After validation, we imported the cleaned CSV into Power BI and used Power Query to apply key transformations:

- Renamed columns for readability (e.g., publishedAt → Published Date)
- Filtered out incomplete rows if any appeared during slicing
- Created calculated columns, such as:

  SimulatedSalary using conditional logic based on experience level

We used “Transform Data” to create and clean dimensions, especially for visual compatibility across pages.

DAX Measures in Power BI

To support dynamic visuals, we created several DAX measures. Here are a few examples:

- Total Job Listings = COUNT('jobs'[title])
- Total Applications = SUM('jobs'[applicationsCount])
- Applications per Job = DIVIDE([Total Applications], [Total Job Listings])
- Average Salary = AVERAGE('jobs'[simulatedSalary])
- Highest Salary = MAX('jobs'[simulatedSalary])

These DAX queries powered key card visuals and interactive metrics across all pages of the dashboard.