Week 5: Introduction to Pandas and Data Cleaning

Assignment Question: •Use Pandas to clean and preprocess a messy dataset, documenting the steps taken during the cleaning process.

SOLUTION TO THE ASSIGNMENT

IMPORTING THE PANDAS LIBRARY: Below is how to import pandas.

```
# Importing the pandas library
import pandas as pd
```

CHECKING THE VERSION OF THE PANDAS

```
# Checking the version of the panda
print(pd._version)
```

```
<module 'pandas._version' from '/usr/local/lib/python3.10/dist-packages/pandas/
```

UPLOADING THE DATASET

```
# Uploading a dataset
Dsonr =pd.read_excel('/content/sample_data/Roni dataset.xls')
```

INSFECTING THE DATASET

```
# Inspect the dataset to understand its structure and identify issues
print('Initial dataset shape:', Dsonr.shape)
```

```
Initial dataset shape: (257, 11)
```

```
print("Column names:", Dsonr.columns)
```

```
Column names: Index(['Name', 'Email', 'Phone', 'PQE NUMBER', 'RRR Number', 'Sta
       'Expiry Days', 'Entry Date', 'Status', 'ID', 'Category'],
      dtype='object')
```

```
print("Data types:\n", Dsonr.dtypes)
```

```
Data types:
 Name            object
Email           object
```

```
Phone          float64
PQE NUMBER      object
RRR Number      object
State           object
Expiry Days     object
Entry Date      object
Status          object
ID               int64
Category        object
dtype: object
```

CATEGORY FILTERING

```
# Filtering the category of D
Filtered_Dsonr = Dsonr[Dsonr['Category'] =="D"]
print(Filtered_Dsonr)
```

```
                          Name            Email         Phone  \
0    A'ISHATU ALIYU ABDULLAHI              NaN  8.037415e+09
1                ABBA DAHIRU  sqqm@gmail.com  8.104987e+09
2                 ABBA DAUDA              NaN  8.065337e+09
3             ABDU ADAMU ZAI              NaN  8.039644e+09
5                   ABDU MUSA             NaN  8.110297e+09
..                        ...             ...           ...
249              YUNUSA SAADU   ysa@gmail.com  9.024332e+09
250             YUSIF A. MATI              NaN  8.132526e+09
253              ZAINAB ALIYU              NaN  8.067227e+09
254           ZAINAB SULE ALI              NaN  8.163621e+09
255               ZAINAB UMAR              NaN  8.138245e+09

               PQE NUMBER      RRR Number State Expiry Days  Entry Date  \
0    JG/PQE/A021/P/00161  3504-5981-0231     F   Unlimited  03-03-2021
1    JG/PQE/A021/P/00008  3504-2187-6627    M    Unlimited  28-09-2020
2    JG/PQE/A021/P/00135  1304-5794-3145    M    Unlimited  10-02-2021
3    JG/PQE/A021/S/00060  1504-5747-9821    M    Unlimited  09-02-2021
5    JG/PQE/A021/P/00093  3204-4049-7041    M    Unlimited  06-02-2021
..                   ...             ...   ...         ...         ...
249  JG/PQE/A021/P/00004  3504-2187-6627    M    Unlimited  28-09-2020
250  JG/PQE/A021/P/00054  3004-3369-8992    M    Unlimited  10-02-2021
253  JG/PQE/A021/P/00148  1804-5931-7343     F   Unlimited  18-02-2021
254  JG/PQE/A021/P/00090  3204-4049-7041     F   Unlimited  06-02-2021
255  JG/PQE/A021/P/00087  3204-4049-7041     F   Unlimited  06-02-2021

     Status    ID Category
0    Active  5137        D
1    Active  4888        D
2    Active  5076        D
3    Active  5030        D
5    Active  4958        D
..      ...   ...      ...
249  Active  4884        D
250  Active  5060        D
253  Active  5089        D
254  Active  4955        D
```

```
      255  Active  4952        D

      [166 rows x 11 columns]
```

```
# Filtering the category of C
Filtered_Dsonr = Dsonr[Dsonr['Category'] =="C"]
print(Filtered_Dsonr)
```

```
                                Name           Email         Phone  \
      4               ABDU MAKAMA KIYAWA             NaN  8.065880e+09
      7                  ABDULKARIM MUDI  amu@gmail.com  8.081014e+09
      8       ABDULLAHI ABDULKADIR AHMAD             NaN  7.063322e+09
      11         ABDULLAHI WURNO IBRAHIM             NaN  8.038556e+09
      16   ABIGAIL AYODELE OLORUNKUNLE   ayo@gmail.com  8.135279e+09
      ..                             ...            ...           ...
      244                     YA'U ISAH  yis@gmail.com  7.033620e+09
      245     YAHAYA ABDULLAHI MUHAMMAD             NaN  8.064640e+08
      248                   YAKUBU ISAH             NaN  8.065062e+09
      251                  YUSUF HALADU             NaN  8.039382e+09
      256                ZAKIYYU HALLIRU             NaN  7.034280e+09

                  PQE NUMBER      RRR Number State Expiry Days  Entry Date  \
      4    JG/PQE/A021/S/00047  2204-3963-7432    M    Unlimited  09-02-2021
      7    JG/PQE/A021/S/00010  1404-2659-2348    M    Unlimited  12-10-2020
      8    JG/PQE/A021/S/00079  3504-5894-7474    M    Unlimited  18-02-2021
      11   JG/PQE/A021/S/00024  3304-3156-3658    M    Unlimited  10-02-2021
      16   JG/PQE/A021/S/00006  2304-2600-4647     F   Unlimited  11-10-2020
      ..                   ...            ...   ...          ...         ...
      244  JG/PQE/A021/S/00004  3303-5595-8740    M    Unlimited  02-10-2020
      245  JG/PQE/A021/S/00091  3504-5894-7474    M    Unlimited  03-03-2021
      248  JG/PQE/A021/S/00052  3204-4665-9027    M    Unlimited  09-02-2021
      251  JG/PQE/A021/S/00082  1804-5931-7343    M    Unlimited  18-02-2021
      256  JG/PQE/A021/S/00050  3204-4665-9027    M    Unlimited  09-02-2021

           Status    ID Category
      4    Active  4996        C
      7    Active  4917        C
      8    Active  5107        C
      11   Active  5037        C
      16   Active  4913        C
      ..      ...   ...      ...
      244  Active  4900        C
      245  Active  5131        C
      248  Active  4999        C
      251  Active  5092        C
      256  Active  4994        C

      [89 rows x 11 columns]
```

HANDLE MISSING VALUES

```
# Check for missing values
print("Missing values:\n", Dsonr.isnull().sum())
```

```
Missing values:
 Name             0
Email           182
Phone            10
PQE NUMBER        0
RRR Number        0
State             0
Expiry Days       0
Entry Date        0
Status            0
ID                0
Category          0
dtype: int64
```

REMOVING REPLICATES

```
# Remove duplicates if any
Dsonr.drop_duplicates(inplace=True)


# Identify outliers using domain knowledge or statistical methods
# Remove outliers using z-score or IQR method


# Rename columns if needed
Dsonr.rename(columns={'Old_column_name': 'new_column_name'}, inplace=True)


# Reorder columns if needed
#Dsonr= Dsonr[['column1', 'column2', ...]]
```

DESCRIPTION OF THE DATASET

```
# Display the description of the dataset
Dsonr.describe()
```

	Phone	ID
count	2.470000e+02	257.000000
mean	8.277320e+09	5009.914397
std	7.131125e+09	74.643636
min	7.030390e+05	4881.000000
25%	7.063322e+09	4946.000000
50%	8.038325e+09	5010.000000
75%	8.099677e+09	5074.000000
max	9.064412e+10	5139.000000

```
# Summary of cleaning process
print("Final dataset shape:", Dsonr.shape)
print("Cleaning process completed.")
```

```
Final dataset shape: (257, 11)
Cleaning process completed.
```