**import** pandas **as** pd

**import** numpy **as** np

data**=**pd**.**read\_csv("MusicData.csv")

data**.**head(3)

data**.**isnull()**.**sum()

data**.**shape

data\_droped**=**data**.**dropna()

data**.**isnull()**.**sum()

df**=**data['Value (Actual)']**.**bfill(**inplace=**True)

data**.**isnull()**.**sum()

data**.**drop\_duplicates()

data.shape

**for** col **in** data**.**select\_dtypes(**include=**'object')**.**columns**:**

    data[col]**=**data[col]**.**str**.**lower()

print(data[col])

print(data)

**for** col **in** data**.**select\_dtypes(**include=**'object')**.**columns**:**

    data[col] **=** data[col]**.**str**.**upper()

print(data[col])

import pandas as pd

# Step 1: Load the data

df = pd.read\_csv("MusicData.csv")

# Step 2: Clean column headers (lowercase, no spaces)

df.columns = df.columns.str.strip().str.lower().str.replace(' ', '\_')

# Step 3: Identify and count missing values

print("Missing values per column:\n", df.isnull().sum())

# Step 4: Handle missing values

# Example: Fill numeric columns with mean, text columns with 'Unknown'

for col in df.columns:

if df[col].dtype == 'object':

df[col] = df[col].fillna('Unknown')

else:

df[col] = df[col].fillna(df[col].mean())

# Step 5: Remove duplicates

df = df.drop\_duplicates()

# Step 6: Standardize text values (like gender or country)

# Example for gender:

if 'gender' in df.columns:

df['gender'] = df['gender'].str.strip().str.lower().replace({'f': 'female', 'm': 'male'})

# Step 7: Convert date columns to datetime

# Assuming there’s a column named 'date' (update with real column name)

if 'date' in df.columns:

df['date'] = pd.to\_datetime(df['date'], dayfirst=True, errors='coerce')

# Step 8: Fix data types

# Example: Convert 'age' to int

if 'age' in df.columns:

df['age'] = df['age'].fillna(0).astype(int)

# Preview cleaned data

print("\nCleaned Data Preview:\n", df.head())

Interview Question

**1. What are missing values and how do you handle them?**

**Missing values** occur when no data value is stored for a variable in an observation.  
**Handling techniques:**

* **Remove rows/columns** using dropna() (if missing values are high).
* **Impute values** using:
  + Mean/Median/Mode
  + Forward/Backward fill
  + Prediction models or KNN imputation.

**2. How do you treat duplicate records?**

To remove duplicate records:

* Use df.duplicated() to identify them.
* Use df.drop\_duplicates() to remove them. You may also consider domain knowledge to define what counts as a duplicate.

**3. Difference between dropna() and fillna() in Pandas?**

* dropna(): Removes missing values (NaNs) from the DataFrame.
* fillna(): Fills missing values with specified values (e.g., mean, median, 0, method='ffill').

**4. What is outlier treatment and why is it important?**

**Outlier treatment** involves detecting and handling extreme values that deviate significantly.  
**Importance:**

* Prevents skewed analysis
* Improves model accuracy  
  **Techniques:**
* Z-score, IQR, or visualizations (boxplots)
* Remove or cap (Winsorizing) outliers

**5. Explain the process of standardizing data.**

Standardization transforms data to have:

* Mean = 0
* Standard deviation = 1  
  Formula:  
  z = (x - mean) / std deviation  
  Helps in algorithms that are sensitive to feature scale (e.g., SVM, KNN).

**6. How do you handle inconsistent data formats (e.g., date/time)?**

* Use pd.to\_datetime() for converting strings to datetime.
* Normalize formats using:
  + String operations
  + Format detection tools
* Validate and replace incorrect formats

**7. What are common data cleaning challenges?**

* Missing or duplicate data
* Inconsistent formatting
* Outliers and noise
* Incorrect data types
* Human errors
* Encoding/categorical issues

**8. How can you check data quality?**

* **Summary stats** (df.describe())
* **Missing values check** (df.isnull().sum())
* **Data type consistency**
* **Value ranges and uniqueness**
* **Visual checks** (histograms, boxplots)