Interview Questions Related To Above Task

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. They can result from errors in data collection, entry, or transmission.

Handling strategies:

- Deletion: Remove rows or columns with missing values (e.g., dropna()).
- Imputation: Replace with mean, median, mode, or predictive methods (e.g., fillna()).
- Flagging: Create an indicator variable that flags missing values.

---

2. How do you treat duplicate records?

Duplicates are repeated rows that can distort analysis.

Treatment:

- Detection: Use df.duplicated() in Pandas.
- Removal: Use df.drop_duplicates() to remove them.
- Sometimes, duplicates are valid (e.g., multiple purchases by same user), so context matters.

---

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

- dropna(): Removes rows or columns with missing values.
- fillna(): Fills missing values with a specified value (mean, median, constant, etc.).

---

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

Outliers are extreme values that deviate significantly from other observations.

Treatment:

- Detection: Using z-scores, IQR, or visualization (boxplots, scatter plots).

- Handling: Remove, cap (winsorization), or transform them.

Importance: Outliers can skew statistical analyses and model performance.

---

5. Explain the process of standardizing data.

Standardization involves rescaling features so they have a mean of 0 and standard deviation of 1.

Steps:

python

CopyEdit

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

scaled_data = scaler.fit_transform(data)

This is especially important for algorithms sensitive to scale (e.g., SVM, KNN).

---

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

Inconsistent formats can cause parsing and analysis issues.

Solution:

- Convert formats using pd.to_datetime() for dates.
- Normalize units (e.g., converting all measurements to the same unit).
- Apply string formatting and parsing tools (e.g., regex) for text fields.

---

7. What are common data cleaning challenges?

- Missing, duplicate, or inconsistent data
- Outliers and noise
- Data type mismatches
- Encoding issues (e.g., special characters)
- Unstructured or semi-structured formats (e.g., JSON, HTML)

---

8. How can you check data quality?

- Summary statistics (df.describe())
- Data types (df.dtypes)
- Missing values check (df.isnull().sum())
- Uniqueness and duplication checks
- Visual inspection (e.g., histograms, boxplots)
- Validation rules (e.g., age cannot be negative)