# Quiz Bank: Data Cleaning, Preprocessing & Feature Engineering

*13 Steps × 5 Questions = 65 Medium Complexity Quizzes*

## Step 1: Schema & Dtype Validation

- 1. You receive a dataset where 'income' is stored as '40,000' with commas. How would you safely coerce this column to numeric and count how many entries became NaN?
- 2. A 'joined_on' column is mixed with '2021-07-12' and '12/07/2021'. How do you parse it consistently while preserving invalid entries for inspection?
- 3. After running astype('int') on a float column with NaNs, the code breaks. How should you fix it without losing values?
- 4. Which function would you use to auto-detect better dtypes and why is this useful?
- 5. A JSON import gives you 'id' as float64 (with trailing .0). How do you ensure it is strictly integer type without corrupting the keys?

## Step 2: Text Normalization

- 1. Customer 'North ' and 'north' appear as separate values. Write Python to normalize case and spaces.
- 2. In free-text 'email only', extra spaces exist. Show how you would normalize this using regex.
- 3. Why would you apply .lower() before one-hot encoding categorical variables?
- 4. A dataset has 'region': ['NORTH','South',' East ']. Show code to get unique values after cleaning.
- 5. What’s the risk of lowercasing free-text product reviews before sentiment analysis?

## Step 3: De-duplication

- 1. If two rows share the same age, region, and segment but differ in income, should you drop them? Why or why not?
- 2. Show code to count how many duplicates exist in columns ['customer_id'] while keeping the latest timestamp.
- 3. You find df.duplicated().sum() == 0, but still see 'visual duplicates'. What might be happening?
- 4. When would you prefer keep='first' vs keep='last' in drop_duplicates?
- 5. Write code to drop duplicate rows ignoring case in the region column.

## Step 4: Missing Values

- 1. A dataset has 15% missing in 'income'. Which imputation strategy is more robust: mean or median, and why?
- 2. Show code to add a boolean 'was_missing_income' column.
- 3. Why might KNNImputer outperform SimpleImputer for features like 'height' and 'weight'?
- 4. How can imputing with a constant (e.g., -999) introduce bias in linear models?
- 5. After imputing with median, accuracy improves but recall drops. What might be the cause?

## Step 5: Outliers

- 1. Show how to cap 'tenure_years' at the 99th percentile.
- 2. Why might you prefer RobustScaler over StandardScaler when outliers exist?
- 3. A LocalOutlierFactor flags 2% of rows. Should you always drop them? Why?
- 4. In fraud detection, are extreme outliers always 'bad data'? Explain.
- 5. Show code to visualize 'income' before and after outlier capping.

## Step 6: Business Rule Validation

- 1. You expect age >= 18. Show code that raises an error if violated.
- 2. Why are assert checks useful at ingestion time?
- 3. A 'segment' column unexpectedly has 'D'. What is the best first step—drop, replace, or flag?
- 4. What’s the difference between schema validation and data validation?
- 5. Write a function that returns a dict of schema violations in income and age.

## Step 7: Categorical Encoding

- 1. When encoding 'education' = ['High School', 'Bachelors', 'Masters'], why might ordinal encoding mislead linear regression?
- 2. Write code to one-hot encode 'region', dropping the first category.
- 3. What is the risk of one-hot encoding 'city' with 10,000 unique values?
- 4. When would you use target encoding instead of one-hot?
- 5. In decision trees, why might label encoding still work while failing in linear models?

## Step 8: Scaling

- 1. Show code to apply StandardScaler to numeric columns.
- 2. Which scaler would you prefer for 'salary' with extreme outliers?
- 3. Why is scaling important for PCA?
- 4. Does scaling matter for decision trees? Why or why not?
- 5. Show how to compare histograms before/after scaling.

## Step 9: Datetime Features

- 1. Write code to extract year, month, and weekday from 'joined_on'.
- 2. Why is 'days_since_signup' often more predictive than raw 'joined_on'?
- 3. How would you create a 'is_weekend' flag?
- 4. Why might 'quarter' be useful in retail datasets?
- 5. What’s the pitfall of one-hot encoding year?

## Step 10: Binning

- 1. Use KBinsDiscretizer to bin 'income' into 5 quantile bins.
- 2. Why might binning help logistic regression capture non-linearity?
- 3. Compare pd.cut vs pd.qcut for age. Which ensures equal counts per bin?
- 4. In credit scoring, why are variables often binned?
- 5. What is a drawback of fixed binning across different populations?

## Step 11: Feature Selection

- 1. Why does VarianceThreshold remove near-constant features?
- 2. Show code to select top-5 features using mutual_info_classif.
- 3. How does MI differ from correlation in capturing feature importance?
- 4. When might wrapper methods (e.g., RFECV) be preferred?
- 5. What’s the risk of selecting features before splitting data?

## Step 12: Leakage-Safe Pipelines

- 1. Why must scaling/encoding happen inside cross-validation folds?
- 2. Show code to build a pipeline with imputation → scaling → logistic regression.
- 3. What’s the difference between fitting a transformer on full data vs only training data?
- 4. Why does ColumnTransformer prevent errors with mixed types?
- 5. Why is leakage especially dangerous in time-series data?

## Step 13: End-to-End Integration

- 1. Design a pipeline that includes KNNImputer, StandardScaler, and RandomForestClassifier.
- 2. Your pipeline works on training data but fails on unseen categories in test. How do you fix it?
- 3. Why is using Pipeline preferable to writing sequential fit_transform steps manually?
- 4. How would you add PCA into an existing pipeline after scaling?
- 5. You deployed a model pipeline last year. New data has unseen categories in region. What’s your strategy?