1. **What are the different types of missing data?**
   * MCAR (Missing Completely at Random): Missingness is unrelated to any variable.
   * MAR (Missing at Random): Missingness is related to observed variables but not the missing data itself.
   * MNAR (Missing Not at Random): Missingness is related to the missing values themselves.
2. **How do you handle categorical variables?**

Encoding techniques like:

* + - Label Encoding (for ordinal categories)
    - One-Hot Encoding (for nominal categories)
    - Target Encoding (for high cardinality features)
  + Or sometimes dropping if not informative.

1. **What is the difference between normalization and standardization?**
   * Normalization (Min-Max Scaling): Scales features to a range (typically 0 to 1).
   * Standardization: Scales features to have mean=0 and std=1. Better for algorithms assuming normally distributed data.
2. **How do you detect outliers?**
   * Visualization: Boxplots, scatter plots
   * Statistical methods: Z-score, IQR method
   * Domain knowledge: Understanding expected value ranges
3. **Why is preprocessing important in ML?**
   * Improves model performance and accuracy
   * Helps algorithms converge faster
   * Makes data compatible with algorithm requirements
   * Reduces bias from irrelevant data patterns
4. **What is one-hot encoding vs label encoding?**
   * One-hot creates new binary columns for each category (better for nominal data).
   * Label encoding assigns a number to each category (better for ordinal data or tree-based models).
5. **How do you handle data imbalance?**
   * Resampling: Oversampling minority class or undersampling majority class
   * Synthetic data generation (SMOTE)
   * Class weights in algorithms
   * Different evaluation metrics (precision, recall, F1)
6. **Can preprocessing affect model accuracy?**

Yes, significantly. Proper preprocessing can:

* + - Improve accuracy by removing noise
    - Prevent bias from missing data
    - Ensure features are on comparable scales
    - Make patterns more discernible to algorithms