Complete Step-by-Step Guide for Using Weka (GUI-Based, No Coding)

# Part A: Preprocessing & Attribute Handling

## Q1. Categorical vs Real-Valued Attributes

1. Open Weka → Explorer → Preprocess tab → Open file (load .arff or .csv).  
2. In the right panel:  
 - Nominal = Categorical  
 - Numeric = Real-valued

## Q2. Identify Crucial Attributes

1. After loading dataset → click Visualize All.  
2. Observe attribute-class relationships.  
3. Commonly crucial attributes: credit\_history, checking\_status, credit\_amount, employment, purpose, age

## Finding Attribute Importance (Information Gain)

1. Go to Select attributes tab.  
2. Evaluator: InfoGainAttributeEval  
3. Search Method: Ranker  
4. Click Start → attributes ranked by IG.  
 Example:  
 0.281 credit\_history  
 0.245 checking\_status  
 0.210 credit\_amount  
 0.173 employment  
 0.000 num\_dependents  
5. Attributes with 0 IG can be removed.

# Part B: Classification (Decision Trees & Rules)

## Q3. Train Decision Tree (J48)

1. Go to Classify tab.  
2. Classifier: trees → J48  
3. Test option: Use training set  
4. Click Start  
- Output → Tree structure, accuracy, confusion matrix.

## Q4. Training Set Accuracy

Found in: % Correctly Classified Instances.  
⚠️ 100% accuracy = Overfitting

## Q5. Cross-Validation

1. In Classify → change test option to Cross-validation (10 folds).  
2. Run J48.  
3. Accuracy usually lower than training accuracy (more realistic).

## Q6. Use Fewer Attributes

1. Go to Preprocess → Remove all except: credit\_history, purpose, employment, other\_parties, housing, residence\_since, class.  
2. Run J48 with 10-fold cross-validation again.  
3. Compare performance with full-attribute model.

## Q7. Smaller Tree with Fewer Attributes

Expect: simpler tree, accuracy may be similar or slightly lower.  
Benefit: better interpretability.

## Q8. Simple vs Complex Trees

In J48 → adjust pruning or minNumObj.  
Trade-off:  
- Complex trees → high accuracy on training, poor generalization.  
- Simpler trees → lower variance, better generalization.

## Q9. Reduced Error Pruning

1. Classifier: trees → ReducedErrorPruning (or J48 with unpruned = false).  
2. Run with cross-validation.  
3. Output: simpler tree; compare accuracy vs original J48.

## Q10. Convert Tree to Rules + Rule Classifiers

A. Manual Conversion:  
 - J48 tree → convert branches into IF-THEN rules.  
 Example: IF credit\_history = good AND credit\_amount < 2000 THEN class = good.  
  
B. Rule-Based Classifiers:  
 - rules → PART → generates set of rules (cross-validation).  
 - rules → OneR → single attribute rule (very simple).  
  
C. Compare Results:  
- Compare accuracy: J48 vs PART vs OneR.

# Part C: Clustering

## Q11. K-Means Clustering

1. Go to Cluster tab.  
2. Classifier: SimpleKMeans → Set numClusters = 2.  
3. In Preprocess → remove class attribute (unsupervised learning).  
4. Run → view cluster assignments & visualize clusters.

# Part D: Other Classifiers

## Q12. SVM vs Decision Tree

1. Go to Classify tab.  
2. Choose functions → SMO (Weka’s SVM).  
3. Run 10-fold cross-validation.  
4. Compare accuracy & confusion matrix vs J48.

# Part E: Train/Test Splitting

## Percentage Split (e.g., 80/20, 60/40)

1. Load dataset → Classify tab.  
2. Choose classifier (e.g., J48).  
3. Test option → Percentage split:  
 - 80 → Train on 80%, Test on 20%  
 - 60 → Train on 60%, Test on 40%  
4. Click Start → accuracy, confusion matrix, precision/recall.  
5. ⚠️ Weka shuffles randomly → set random seed (e.g., 42) for reproducibility.

# Final Summary Table

|  |  |
| --- | --- |
| Task | Weka Tool/Step |
| View attributes | Preprocess → Check Nominal vs Numeric |
| Attribute ranking (IG) | Select attributes → InfoGain + Ranker |
| Train Decision Tree | Classify → J48 |
| Cross-validation | Classify → 10-fold CV |
| Feature selection | Preprocess → Remove unwanted attributes |
| Tree pruning | J48 options OR ReducedErrorPruning |
| Rule classifiers | Classify → rules → PART / OneR |
| Clustering | Cluster tab → SimpleKMeans |
| Compare SVM vs Tree | Classify → functions → SMO vs J48 |
| Train/test split | Classify → Test option → Percentage split |