**Interview Questions:**

**1. How does a decision tree work?**

A **Decision Tree** is a model that splits data into smaller groups based on feature values to make predictions.

* It starts from the **root node** (the whole dataset).
* At each step, it picks the **best feature** to split the data into subsets (using metrics like *Information Gain* or *Gini Index*).
* The process continues until the tree reaches **leaf nodes** (final decisions or classes).

Example:  
If you’re predicting whether to play tennis:

* Root: “Is it sunny?”
  + Yes → “Is humidity high?” → No → Play
  + No → Don’t play

**2. What is entropy and information gain?**

* **Entropy** measures **impurity or uncertainty** in data.
  + If all samples belong to one class → entropy = 0 (pure).
  + If samples are evenly split → entropy = 1 (impure).

Formula:

  
**Information Gain (IG)** measures how much a feature **reduces entropy** when splitting.  
  
IG = Entropy(parent) - Weighted average of entropy(children)  
  
The feature with **highest IG** is chosen for the split.

**3. How is random forest better than a single tree?**

A **Random Forest** builds **many decision trees** and combines their results (usually by majority voting).  
 **Advantages:**

* Reduces **overfitting** (trees are less correlated).
* Increases **accuracy and stability**.
* Handles **large datasets** and **missing values** well.

**4. What is overfitting and how do you prevent it?**

**Overfitting** happens when the model learns **noise** instead of patterns—works well on training data but poorly on new data.

**Prevention techniques:**

* **Pruning** the tree (limit depth, min samples per leaf).
* **Use Random Forest** or **Bagging** (reduces variance).
* **Cross-validation** (evaluate model on unseen data).

**5. What is bagging?**

**Bagging** = **Bootstrap Aggregating**.  
It trains multiple models (like decision trees) on **different random subsets** of the training data (with replacement).  
Final prediction = **average (for regression)** or **majority vote (for classification)**.

Random Forest = Decision Trees + Bagging + Random Feature Selection.

**6. How do you visualize a decision tree?**

You can visualize a tree using Python libraries:

from sklearn.tree import plot\_tree

plot\_tree(model, feature\_names=features, class\_names=classes, filled=True)

**7. How do you interpret feature importance?**

Feature importance tells **how much each feature contributes** to model decisions.  
In decision trees or random forests:

* It’s based on how much **information gain** (or Gini reduction) the feature provides across all splits.
* Higher importance = more influence on prediction.

Example:

| **Feature** | **Importance** |
| --- | --- |
| Age | 0.45 |
| Income | 0.30 |
| Education | 0.25 |

**8. What are the pros/cons of random forests?**

**Pros:**

* High accuracy and robustness.
* Handles missing data and categorical values.
* Reduces overfitting.
* Provides feature importance.

**Cons:**

* Slower for large datasets.
* Harder to interpret (many trees).
* Uses more memory and computation.