# Lab 2

**Code**

```
import pandas as pd
from collections import Counter

def entropy(data):
    labels = data['label'].tolist()
    counts = Counter(labels)
    probabilities = [count / len(labels) for count in counts.values()]
    entropy_value = -sum(p * math.log2(p) for p in probabilities if p > 0)  # Handle
probabilities of 0
    return entropy_value

def gain(data, feature):
    initial_entropy = entropy(data)
    feature_values = data[feature].unique()
    weighted_entropy = 0
    for value in feature_values:
        subset = data[data[feature] == value]
        weighted_entropy += (len(subset) / len(data)) * entropy(subset)
    return initial_entropy - weighted_entropy

def id3(data, features, target_attribute):
    # Check if all examples have the same label
    if len(data['label'].unique()) == 1:
        return data['label'].iloc[0]

    # Check if features is empty
    if len(features) == 0:
      return data['label'].value_counts().index[0]  # Return the most common label

    best_feature = max(features, key=lambda feature: gain(data, feature))
    tree = {best_feature: {}}
    features = [f for f in features if f != best_feature]

    for value in data[best_feature].unique():
        subset = data[data[best_feature] == value].drop(columns=[best_feature])
        if len(subset) == 0:
            tree[best_feature][value] = data['label'].value_counts().index[0]  # Handle empty
subsets
```

```
        else:
            tree[best_feature][value] = id3(subset, features, target_attribute)

    return tree

import math
# Sample dataset
data = {'outlook': ['sunny', 'sunny', 'overcast', 'rainy', 'rainy', 'rainy', 'overcast', 'sunny',
'sunny', 'rainy', 'sunny', 'overcast', 'overcast', 'rainy'],
        'temperature': ['hot', 'hot', 'hot', 'mild', 'cool', 'cool', 'cool', 'mild', 'cool', 'mild', 'mild',
'mild', 'hot', 'mild'],
        'humidity': ['high', 'high', 'high', 'high', 'normal', 'normal', 'normal', 'high', 'normal',
'normal', 'normal', 'high', 'normal', 'high'],
        'wind': ['weak', 'strong', 'weak', 'weak', 'weak', 'strong', 'strong', 'weak', 'weak',
'weak', 'strong', 'strong', 'weak', 'strong'],
        'label': ['no', 'no', 'yes', 'yes', 'yes', 'no', 'yes', 'no', 'yes', 'yes', 'yes', 'yes', 'yes', 'no']}

df = pd.DataFrame(data)

# Features and target attribute
features = ['outlook', 'temperature', 'humidity', 'wind']
target_attribute = 'label'

# Create the ID3 decision tree
decision_tree = id3(df, features, target_attribute)

# Print the decision tree
decision_tree
```

**Output**

```
{'outlook': {'sunny': {'humidity': {'high': 'no', 'normal': 'yes'}},
  'overcast': 'yes',
  'rainy': {'wind': {'weak': 'yes', 'strong': 'no'}}}}
```