12/03/25

Lab - 2

Q1: Use an appropriate dataset for building the decision tree (ID3) & apply this knowledge to classify a new sample.

→
```
import pandas as pd
import numpy as np
from collections import Counter
import math.

def entropy(y):
    counts = Counter(y)
    prob = [count/len(y) for count in counts.values()]
    return -sum(p * math.log2(p) for p in prob)

def info_gain(data, feature, target):
    total_entropy = entropy(data[target])
    values = data[feature].unique()
    weighted_entropy = sum((len(data[feature]==v])/
    len(data)) * entropy(data[data[feature]==v][target])
    for v in values)
    return total entropy - weighted_entropy

def id3(data, features, target):
    if len(set(data[target]))==1:
        return data[target].iloc[0]
    if len(features)==0:
        return data[target].mode()[0]
    gains = {feature: info_gain(data, feature, target)
             for feature in features}
    best_feature = max(gains, key = gains.get)
```

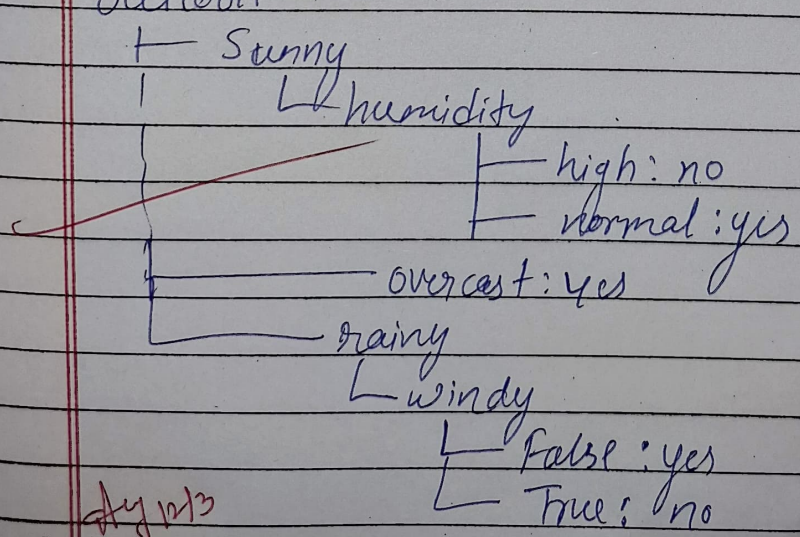
F54 5G

```
tree = {best_feature : {}}
return tree.

def print_tree(tree, indent = " "):
    if not isinstance(tree, dict):
        print(indent + "→" + str(tree))
        return
    for key, value in tree.items():
        print(indent + str(key))
        for sub_key, sub_tree in value.items():
            print(indent + "L" + str(sub_key))

path = '/content/tennis.csv'
data = pd.read_csv(filepath)
features = list(data.columns[:-1])
target = 'play'
decision_tree = id3(data, features, target)
print_tree(decision_tree).
```

→ Output :-



day 2/3