2.For a given set of training data examples stored in a .CSV file, implement and demonstrate the

Candidate-Elimination algorithm in python to output a description of the set of all hypotheses

consistent with the training examples

```
import csv
# Load data from CSV file
def load_csv(filename):
    data = []
    with open(filename, 'r') as file:
        reader = csv.reader(file)
        headers = next(reader)  # skip header
        for row in reader:
            data.append(row)
    return data
# Candidate Elimination Algorithm
def candidate_elimination(data):
    num_attributes = len(data[0]) - 1

    # Initialize Specific and General boundaries
    S = ['0'] * num_attributes
    G = [['?'] * num_attributes]
  print("Initial Specific Hypothesis S:", S)
    print("Initial General Hypothesis G:", G)
    print("-" * 60)
for i, example in enumerate(data):
        instance = example[:-1]
        label = example[-1]
print(f"Training Example {i+1}: {example}")

        # POSITIVE EXAMPLE
        if label.lower() == 'yes':
```

```
            for j in range(num_attributes):
                if S[j] == '0':
                    S[j] = instance[j]
                elif S[j] != instance[j]:
                    S[j] = '?'

            # Remove hypotheses from G that do not satisfy S
            G = [g for g in G if all(g[j] == '?' or g[j] == S[j] for j in range(num_attributes))]

        # NEGATIVE EXAMPLE
        else:
            new_G = []
            for g in G:
                for j in range(num_attributes):
                    if g[j] == '?' and S[j] != '?' and S[j] != instance[j]:
                        new_hypothesis = g.copy()
                        new_hypothesis[j] = S[j]
                        new_G.append(new_hypothesis)

            G = new_G

        print("Specific Hypothesis S:", S)
        print("General Hypothesis G:", G)
        print("-" * 60)

    return S, G
# MAIN FUNCTION
if __name__ == "__main__":
    filename = "training_data.csv"   # CSV file name
    training_data = load_csv(filename)
```

```
    S_final, G_final = candidate_elimination(training_data)
print("\nFINAL RESULT")
    print("Final Specific Hypothesis S:", S_final)
    print("Final General Hypotheses G:", G_final)
```

```
>>>
========================================= RESTART: C:/Users/yasha/OneDrive/Desktop/ML LAB/id3_decision_tree.py ==================================
Decision Tree:
{'Outlook': {'Sunny': {'Humidity': {'High': 'No', 'Normal': 'Yes'}}, 'Rain': {'Wind': {'Weak': 'Yes', 'Strong': 'No'}}, 'Overcast': 'Yes'}}
>>>
====================================== RESTART: C:/Users/yasha/OneDrive/Desktop/ML LAB/candidate_elimination.py ==================================
Initial Specific Hypothesis S: ['0', '0', '0', '0', '0', '0']
Initial General Hypothesis G: [['?', '?', '?', '?', '?', '?']]
------------------------------------------------------------
Training Example 1: ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same', 'Yes']
Specific Hypothesis S: ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same']
General Hypothesis G: [['?', '?', '?', '?', '?', '?']]
------------------------------------------------------------
Training Example 2: ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same', 'Yes']
Specific Hypothesis S: ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
General Hypothesis G: [['?', '?', '?', '?', '?', '?']]
------------------------------------------------------------
Training Example 3: ['Rainy', 'Cold', 'High', 'Strong', 'Warm', 'Change', 'No']
Specific Hypothesis S: ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
General Hypothesis G: [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', 'Same']]
------------------------------------------------------------
Training Example 4: ['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change', 'Yes']
Specific Hypothesis S: ['Sunny', 'Warm', '?', 'Strong', '?', '?']
General Hypothesis G: [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?']]
------------------------------------------------------------

FINAL RESULT
Final Specific Hypothesis S: ['Sunny', 'Warm', '?', 'Strong', '?', '?']
Final General Hypotheses G: [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?']]
>>>
```