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

CODE:

```
data = {
    'Sky': ['Sunny', 'Sunny', 'Rainy', 'Sunny'],
    'AirTemp': ['Warm', 'Warm', 'Cold', 'Warm'],
    'Humidity': ['Normal', 'High', 'High', 'High'],
    'Wind': ['Strong', 'Strong', 'Strong', 'Strong'],
    'Water': ['Warm', 'Warm', 'Warm', 'Cool'],
    'Forecast': ['Same', 'Same', 'Change', 'Change'],
    'EnjoySport': ['Yes', 'Yes', 'No', 'Yes']
}
df = pd.DataFrame(data)
df.to_csv('training_data.csv', index=False)
print("training_data.csv created successfully\n")
print(df)
# Load data
data = pd.read_csv('training_data.csv')
concepts = data.iloc[:, :-1].values
target = data.iloc[:, -1].values
# Initialize S and G
S = concepts[0].copy()
G = [['?' for _ in range(len(S))] for _ in range(len(S))]
print("\nInitial Specific Hypothesis (S):", S)
print("Initial General Hypothesis (G):", G)
# Candidate Elimination
for i, example in enumerate(concepts):
    print(f"\nTraining Example {i+1}: {example} → {target[i]}")

    if target[i] == 'Yes':  # Positive example
        for j in range(len(S)):
            if example[j] != S[j]:
                S[j] = '?'
                G[j][j] = '?'
        print("Updated S:", S)
    else:  # Negative example
        for j in range(len(S)):
            if example[j] != S[j]:
                G[j][j] = S[j]
            else:
                G[j][j] = '?'
        print("Updated G:", G)
# Remove overly general hypotheses
G_final = [g for g in G if g.count('?') < len(g)]
```

```
print("\n------------------------------")
print("Final Specific Hypothesis (S):")
print(S)
print("\nFinal General Hypotheses (G):")
for g in G_final:
    print(g)
```

OUTPUT:

```
training_data.csv created successfully

     Sky AirTemp Humidity    Wind Water Forecast EnjoySport
0  Sunny    Warm   Normal  Strong  Warm     Same        Yes
1  Sunny    Warm     High  Strong  Warm     Same        Yes
2  Rainy    Cold     High  Strong  Warm   Change         No
3  Sunny    Warm     High  Strong  Cool   Change        Yes

Initial Specific Hypothesis (S): ['Sunny' 'Warm' 'Normal' 'Strong'
'Warm' 'Same']
Initial General Hypothesis (G): [['?', '?', '?', '?', '?', '?'], ['?',
'?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?',
'?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?',
'?', '?', '?']]

Training Example 1: ['Sunny' 'Warm' 'Normal' 'Strong' 'Warm' 'Same'] →
Yes
Updated S: ['Sunny' 'Warm' 'Normal' 'Strong' 'Warm' 'Same']

Training Example 2: ['Sunny' 'Warm' 'High' 'Strong' 'Warm' 'Same'] →
Yes
Updated S: ['Sunny' 'Warm' '?' 'Strong' 'Warm' 'Same']

Training Example 3: ['Rainy' 'Cold' 'High' 'Strong' 'Warm' 'Change'] →
No
Updated G: [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?',
'?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?',
'?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?',
'Same']]

Training Example 4: ['Sunny' 'Warm' 'High' 'Strong' 'Cool' 'Change'] →
Yes
Updated S: ['Sunny' 'Warm' '?' 'Strong' '?' '?']

------------------------------
Final Specific Hypothesis (S):
['Sunny' 'Warm' '?' 'Strong' '?' '?']

Final General Hypotheses (G):
['Sunny', '?', '?', '?', '?', '?']
['?', 'Warm', '?', '?', '?', '?']
```