2 For a given set of training data examples stored in a .csv file, implement and demostrate the candidate Elimination algorithm to output a description of the set of all hypothesis consistent with the training examples

```
import csv
with open('C:\Users\user1\Desktop\ML Lab\Book3.csv')as
                                                f:
        csv_file = csv.reader(f)
        data = list(csv_file)
print(data)
S = data[1][:-1]
print(s)
g = [['?' for i in range(len(S)] for j in range
                                    (len(S))]

for i in data:
    if i[-1] == "yes":
        for j in range(len(S)):
            if i[j] != S[j]:
                S[j] = '?'
                g[j][j] = '?'
    elif i[-1] == "no":
        for j in range(len(S)):
            if i[j] != S[j]:
                g[i][j] = S[j]
            else:
                g[i][j] = "?"
```

Teacher's Signature :

```
print("Steps of candidate elimination algorithm",
                                    data.index(i)+1)
print(s)
print(g)
gh = []

for i in g:
    for j in f:
        if j != '?':
            gh.append(i)
            break
print("\n Final specific hypothesis \n",s)
print("\n Final general hypothesis\n", gh)
```

output :-

[['sunny', 'warm', 'normal', 'strong', 'warm', 'same', 'yes']
['sunny', 'warm', 'high', 'strong', 'warm', 'same', 'yes']
['rainy', 'cold', 'high', 'strong', 'warm', 'change', 'no']
['sunny', 'warm', 'high', 'strong', 'cool', 'change', 'yes']]

['sunny', 'warm', 'high', 'strong', 'warm', 'same']

Steps of candidate elimination algorithm 1:

['sunny', 'warm', '?', 'strong', 'warm', 'same']

[['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?']]

Steps of candidate elimination algorithm 2:

['sunny', 'warm', '?', 'strong', 'warm', 'same']

[['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']]

Steps of candidate algorithm 3:

['sunny', 'warm', '?', 'strong', 'warm', 'same']

[['sunny', '?', '?', '?', '?', '?'] ['?', 'warm', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?', 'same']]

Steps of candidate elimination algorithm 4:

['sunny', 'warm', '?', 'strong', '?', '?']

[['sunny', '?', '?', '?', '?', '?'] ['?', '?warm', '?', '?', '?', '?']
['?', '?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']
['?', '?', '?', '?', '?'] ['?', '?', '?', '?', '?', '?']]

Final hypothesis

['sunny', 'warm', '?' 'strong', '?', '?']

Final general hypothesis

[['sunny', '?', '?', '?', '?', '?'] ['?', 'warm', '?', '?', '?', '?']]