CODE

```
import csv

def predict(data, model):
	flag = True
	for i, ele in enumerate(model):
		if ele != "?" and ele != data[i]:
			flag = False
			break
	return flag

def train():
	model_s = ["", "", "", "", "", ""]
	model_g = [["?", "?", "?", "?", "?", "?"]]
	print("model_s(0): " + str(model_s))
	print("model_g(0): " + str(model_g) + "\n")

	with open("prog2_data.csv") as csv_file:
		data = csv.reader(csv_file, delimiter = ',')
		for i, row in enumerate(data):
			# positive example
			if row[-1] == "yes":

				# update model_s
				for j, col in enumerate(row[:-1]):
					if model_s[j] == "":
						model_s[j] = col
					elif col != model_s[j]:
						model_s[j] = "?"

				# eliminate from model_g
				n = len(model_g)
				m = 0
				while m<n:
					if(predict(row[:-1], model_g[m]) == False):
						model_g = model_g[:m] + model_g[m+1:]
						n = n - 1
					m = m+1

			# negative example
			else:
				# update model_g
				for m in range(len(model_g)):
					# generate candidates
					if(predict(row[:-1], model_g[m]) == True):
						toSpecific = model_g[m]
						model_g = model_g[:m] + model_g[m+1:]

						for j, col in enumerate(toSpecific):
							if model_s[j] != "?" and model_s[j] !=
col and model_s[j] != row[j]:

								temp = toSpecific[:]
								temp[j] = model_s[j]

								model_g += [temp]


			print("model_s({}): {}".format(i+1,model_s))
			print("model_g({}): {}\n".format(i+1,model_g))

train()
```

OUTPUT

```
model_s(0): ['', '', '', '', '', '']
model_g(0): [['?', '?', '?', '?', '?', '?']]

model_s(1): ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same']
model_g(1): [['?', '?', '?', '?', '?', '?']]

model_s(2): ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
model_g(2): [['?', '?', '?', '?', '?', '?']]

model_s(3): ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
model_g(3): [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?',
'?', '?'], ['?', '?', '?', '?', '?', 'Same']]

model_s(4): ['Sunny', 'Warm', '?', 'Strong', '?', '?']
model_g(4): [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?',
'?', '?']]
```