

**Name: Devesh Vengurlekar**

**Roll No: 9766**

**TE Comps A**

### **AI Experiment No. 9**

**Title : Simple prototype for Expert System**

**Program:**

```
# Devesh Vengurlekar
```

```
#Roll No: 9766
```

```
# TE Comps A
```

```
class ExpertSystem:
```

```
    def __init__(self):
```

```
        self.rules = {
```

```
            "low_calorie": "Focus on consuming fruits, vegetables, lean proteins, and whole grains.  
Limit added sugars and fats.",
```

```
            "high_protein": "Include plenty of protein-rich foods such as lean meats, fish, eggs, dairy,  
legumes, and nuts.",
```

```
            "low_carb": "Limit carbohydrate intake and focus on consuming non-starchy vegetables,  
lean proteins, and healthy fats.",
```

```
            "balanced_diet": "Eat a variety of foods from all food groups, including fruits, vegetables,  
grains, protein-rich foods, and healthy fats."  
        }
```

```
    def consult(self, dietary_needs):
```

```
        recommendations = []
```

```
        for need in dietary_needs:
```

```
            if need in self.rules:
```

```
                recommendations.append(self.rules[need])
```

```
            else:
```

```
                recommendations.append("Sorry, I'm not sure what to advise for '{}' dietary  
need.".format(need))
```

```
        return recommendations
```

```
def main():
```

```
    expert_system = ExpertSystem()
```

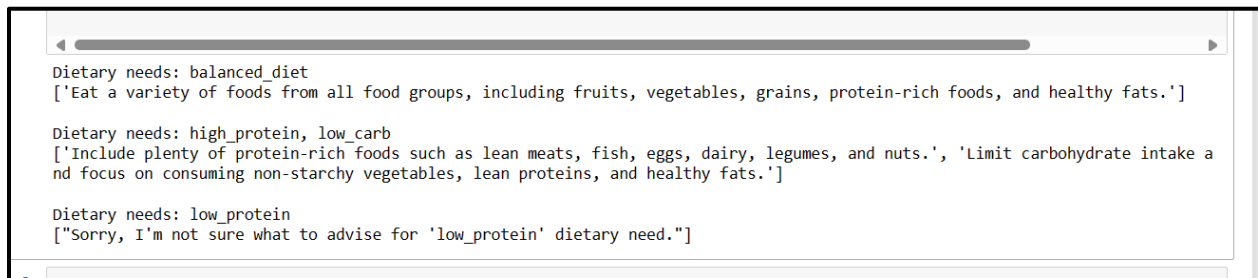
```
    # Example consultations
```

```
    print("Dietary needs: balanced_diet")
```

```
print(expert_system.consult(["balanced_diet"]))
print("\nDietary needs: high_protein, low_carb")
print(expert_system.consult(["high_protein", "low_carb"]))
print("\nDietary needs: low_protein")
print(expert_system.consult(["low_protein"]))

if __name__ == "__main__":
    main()
```

## Output:

A screenshot of a terminal window with a light gray background and a dark gray border. The terminal displays the output of a Python script. It shows three separate outputs, each starting with a label like 'Dietary needs: balanced\_diet' followed by a list of advice in single quotes. The first output is for 'balanced\_diet', the second for 'high\_protein, low\_carb', and the third for 'low\_protein'. The third output indicates that the system is unsure how to advise for the 'low\_protein' need.

```
Dietary needs: balanced_diet
['Eat a variety of foods from all food groups, including fruits, vegetables, grains, protein-rich foods, and healthy fats.']

Dietary needs: high_protein, low_carb
['Include plenty of protein-rich foods such as lean meats, fish, eggs, dairy, legumes, and nuts.', 'Limit carbohydrate intake and focus on consuming non-starchy vegetables, lean proteins, and healthy fats.']

Dietary needs: low_protein
["Sorry, I'm not sure what to advise for 'low_protein' dietary need."]
```