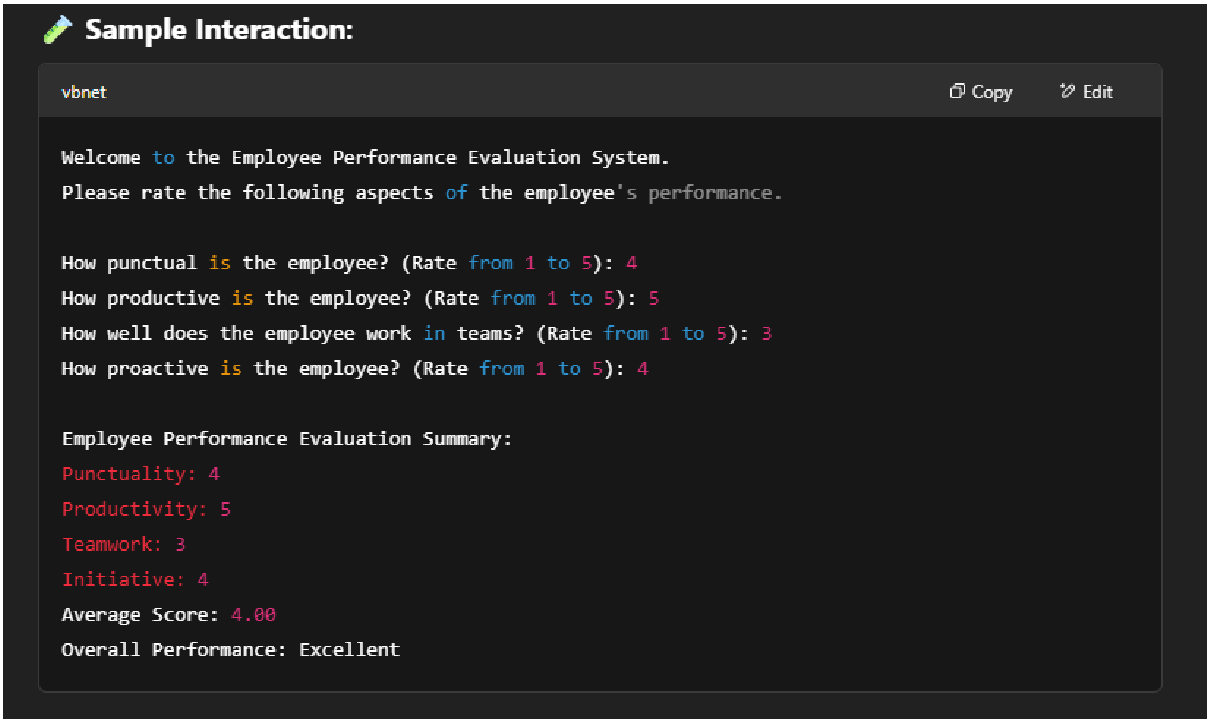
EXPERT SYSTEM-EMPLOYEE PERFORMANCE EVALUATION

```
class EmployeePerformanceExpertSystem:
    def __init__(self):
        self.performance_rating = ""

    def ask_question(self, question, low, high):
        """ Ask the user a question and return a score within a defined range """
        while True:
            try:
                response = int(input(f"{question} (Rate from {low} to {high}): "))
                if response < low or response > high:
                    print(f"Please enter a value between {low} and {high}.")
                else:
                    return response
            except ValueError:
                print("Invalid input, please enter a number.")

    def evaluate_performance(self):
        """ Evaluate performance based on responses """
        print("Welcome to the Employee Performance Evaluation System.\nPlease rate the following aspects of the employee's performance.")

        punctuality = self.ask_question("How punctual is the employee?", 1, 5)
        productivity = self.ask_question("How productive is the employee?", 1, 5)
        teamwork = self.ask_question("How well does the employee work in teams?", 1, 5)
        initiative = self.ask_question("How proactive is the employee?", 1, 5)

        # Calculate average score
        avg_score = (punctuality + productivity + teamwork + initiative) / 4

        # Determine performance rating
        if avg_score >= 4:
            self.performance_rating = "Excellent"
        elif avg_score >= 3:
            self.performance_rating = "Good"
        elif avg_score >= 2:
            self.performance_rating = "Needs Improvement"
        else:
            self.performance_rating = "Poor"

        # Display the performance evaluation
        print(f"\nEmployee Performance Evaluation Summary:")
        print(f"Punctuality: {punctuality}")
        print(f"Productivity: {productivity}")
        print(f"Teamwork: {teamwork}")
        print(f"Initiative: {initiative}")
        print(f"Average Score: {avg_score:.2f}")
        print(f"Overall Performance: {self.performance_rating}")

# Instantiate the expert system and evaluate an employee's performance
expert_system = EmployeePerformanceExpertSystem()
expert_system.evaluate_performance()
```



Sample Interaction:

```
Welcome to the Employee Performance Evaluation System.
Please rate the following aspects of the employee's performance.

How punctual is the employee? (Rate from 1 to 5): 4
How productive is the employee? (Rate from 1 to 5): 5
How well does the employee work in teams? (Rate from 1 to 5): 3
How proactive is the employee? (Rate from 1 to 5): 4

Employee Performance Evaluation Summary:
Punctuality: 4
Productivity: 5
Teamwork: 3
Initiative: 4
Average Score: 4.00
Overall Performance: Excellent
```

Possible Viva Questions for this Practical:

1. **What is an expert system?**
   - An expert system is a software application that mimics the decision-making abilities of a human expert. It uses predefined rules to evaluate situations and make decisions.
2. **What are the key components of this employee performance evaluation system?**
   - The key components include user input (ratings), calculations (average score), and decision-making (determining the performance rating based on the average score).
3. **What is the purpose of the ask_question() method in this system?**
   - The purpose of the ask_question() method is to interact with the user and gather ratings for different aspects of employee performance. It ensures that the input is valid and within the specified range.
4. **What happens if the user inputs an invalid value (e.g., a string or out-of-range value)?**
   - If the user inputs an invalid value, the system catches the error and prompts the user to enter a valid number within the specified range.
5. **How does the system determine the overall performance rating?**

   - The system calculates the average score from the ratings on different aspects and assigns a performance rating (e.g., "Excellent", "Good", "Needs Improvement", "Poor") based on the average score.

6. **What could be improved in this expert system?**
   - The system could be extended to include more parameters, provide detailed feedback for each parameter, or incorporate more advanced algorithms for performance evaluation.

7. **How is the performance rating categorized in this system?**
   - The performance rating is categorized as:
     - Excellent: Average score ≥ 4
     - Good: Average score between 3 and 4
     - Needs Improvement: Average score between 2 and 3
     - Poor: Average score < 2

8. **What data types are used in this expert system?**
   - The system uses integers for the ratings and floating-point numbers for the average score.

9. **Can this system handle multiple employees at once?**
   - No, this system is designed to evaluate one employee at a time. To evaluate multiple employees simultaneously, the system would need to be extended to handle multiple users and inputs.

10. **How could this system be expanded to evaluate more complex performance metrics?**
    - The system could incorporate more complex metrics such as work quality, leadership, innovation, etc., and integrate data from different sources like peer reviews or project outcomes.