The School of Magic's Gradebook Level: Medium Objective: Develop a system to manage students' grades.

You are the headmaster of a school for magical creatures. Each year, you need to track the student's progress, calculate averages, and keep parents updated. Create a grade management system to store and manage students' grades efficiently, allowing easy input and review. Remember, even a dragon deserves fair grades!

Task:

Use a dictionary to store student names and grades. Allow input of new grades, updating existing ones, and viewing a summary. Calculate average grades for individual students and the class. Key Concepts: Dictionaries, loops, input/output handling.

To create the grade management system for the School of Magic, we can use a Python program that utilizes a dictionary to store student names as keys and their grades as values (which can be a list of grades). This program will allow you to input new grades, update existing ones, and view summaries of grades and averages. Below is a complete implementation of the system:

In [ ]:
```
class Gradebook:
    def __init__(self):
        # Initialize the dictionary to store student grades
        self.grades = {}

    def add_grade(self, student_name, grade):
        """Add a new grade for a student or update existing grades."""
        if student_name in self.grades:
            self.grades[student_name].append(grade)
        else:
            self.grades[student_name] = [grade]
        print(f"Grade '{grade}' added for {student_name}.")

    def update_grade(self, student_name, old_grade, new_grade):
        """Update an existing grade for a student."""
        if student_name in self.grades and old_grade in self.grades[student_name]:
            self.grades[student_name].remove(old_grade)
            self.grades[student_name].append(new_grade)
            print(f"Grade '{old_grade}' updated to '{new_grade}' for {student_name}.")
        else:
            print(f"Grade '{old_grade}' not found for {student_name}.")

    def view_summary(self):
        """View a summary of student grades and their averages."""
        if not self.grades:
```

```
            print("No grades available.")
            return

        for student_name, grades in self.grades.items():
            average = sum(grades) / len(grades) if grades else 0
            print(f"{student_name}: Grades: {grades}, Average: {average:.2f}")

    def class_average(self):
        """Calculate and display the class average for all students."""
        total_grades = []
        for grades in self.grades.values():
            total_grades.extend(grades)

        if not total_grades:
            print("No grades available for class average.")
            return

        overall_average = sum(total_grades) / len(total_grades)
        print(f"Class Average: {overall_average:.2f}")


def main():
    gradebook = Gradebook()
    while True:
        print("\nWelcome to the School of Magic Gradebook!")
        print("1. Add Grade")
        print("2. Update Grade")
        print("3. View Summary")
        print("4. Class Average")
        print("5. Exit")

        choice = input("Choose an option: ")

        if choice == "1":
            student_name = input("Enter student name: ")
            grade = float(input("Enter grade (0-100): "))
            gradebook.add_grade(student_name, grade)

        elif choice == "2":
            student_name = input("Enter student name: ")
            old_grade = float(input("Enter old grade: "))
            new_grade = float(input("Enter new grade: "))
            gradebook.update_grade(student_name, old_grade, new_grade)
```

```
        elif choice == "3":
            gradebook.view_summary()

        elif choice == "4":
            gradebook.class_average()

        elif choice == "5":
            print("Exiting the Gradebook. Goodbye!")
            break

        else:
            print("Invalid choice. Please try again.")

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

```
Welcome to the School of Magic Gradebook!
1. Add Grade
2. Update Grade
3. View Summary
4. Class Average
5. Exit
Choose an option: 1
Enter student name: Sujay Naskar
Enter grade (0-100): 50
Grade '50.0' added for Sujay Naskar.

Welcome to the School of Magic Gradebook!
1. Add Grade
2. Update Grade
3. View Summary
4. Class Average
5. Exit
Choose an option: 2
Enter student name: Sujay Naskar
Enter old grade: 50
Enter new grade: 75
Grade '50.0' updated to '75.0' for Sujay Naskar.

Welcome to the School of Magic Gradebook!
1. Add Grade
2. Update Grade
3. View Summary
4. Class Average
5. Exit
```

In [ ]: