Geo Grades - Read Groups

```python
import sys

def read_groups():
    groups = sys.stdin.read().split("=\n")
    #print(groups)

    for group in groups:
        #print(group)
        my_list = group.split("\n")
        #print(my_list)
        flag = 0
        for my_value in my_list:
            #print(my_value)
            if my_value.find("_") != -1:
                flag = 1
                index_name = my_value.index("_")
                # print(line[:index_name])
                index_grades = my_value.rfind(" ")
                grades = my_value[index_grades + 1:]
                grades = grades.split()
                # print(grades)
                # print(index_name)
                sum_of_grades = 0
                for number in grades:
                    sum_of_grades += float(number)

                # print(sum_of_grades)
                print("%s has an average grade of %.1f" % (my_value[:index_name], avg_geo_grade(sum_of_grades)))
            else:
                if flag == 1:
                    print("End of report")
                print(my_value)
    print("End of report")




NUMBER_OF_GRADES = float(3)

def final_grade(avg_geo_grade):
    final_grades = round(avg_geo_grade)
    if final_grades - avg_geo_grade == 0.5:  # avg grade = 5.5 --> 6.0, avg grade
        return final_grades
    elif final_grades - avg_geo_grade < 0 and abs(final_grades - avg_geo_grade) <= 0.25:
        return final_grades
    elif final_grades - avg_geo_grade < 0 and abs(final_grades - avg_geo_grade) > 0.25:
        return final_grades + 0.5
    elif final_grades - avg_geo_grade > 0 and abs(final_grades - avg_geo_grade) <= 0.25:
        return final_grades
    elif final_grades - avg_geo_grade > 0 and abs(final_grades - avg_geo_grade) > 0.25:
        return final_grades - 0.5


def avg_geo_grade(sum_of_grades):
    avg_geo_grade = sum_of_grades / NUMBER_OF_GRADES
    return final_grade(avg_geo_grade)


def read_grades():
    for line in sys.stdin.readlines():
        index_name = line.index("_")
        # print(line[:index_name])
        index_grades = line.rfind(" ")
        grades = line[index_grades + 1:]
        grades = grades.split()
        # print(grades)
        # print(index_name)
        sum_of_grades = 0
        for number in grades:
            sum_of_grades += float(number)

        # print(sum_of_grades)
        print("%s has an average grade of %.1f" % (line[:index_name], avg_geo_grade(sum_of_grades)))


read_groups()
```