```
# Name: Fahmi Omer
# Course: ICS 4U1
# Project: Array Review Task: Make a basic student info program with at
least first name, last name, course and mark. Allow the user to add and
delete students, view the list and sort it.

#Setting some example students for the program and the lists that wil be
used to store information
first_names = ["FAHMI", "MIN", "MOBY", "OMAR"]
last_names = ["OMER", "LIM", "HALL", "El-Sawy"]
courses = ["SCH 4U1", "SPH 4U1", "ICS 4U1", "MCV 4U1"]
course_marks = [100, 12, 200, 97]

#Creating a function to add students that takes in their information
def add_student(firstName, lastName, course, mark):

    first_names.append(firstName)
    last_names.append(lastName)
    courses.append(course)
    course_marks.append(mark)

    print("Student Added!")

#Creating a function to remove students from the class using their first
name, If there are duplicated, the one occuring first will be removed
def remove_student(firstName):

    index = first_names.index(firstName)

    first_names.pop(index)
    last_names.pop(index)
    courses.pop(index)
    course_marks.pop(index)

    print("Student Removed!")

#Creating a function to be able to view the class at any desired time
def view_class():
    print("\nFirst Name\t\tLast Name\t\tCourse Code\t\tCourse
Grade\n----------\t\t---------\t\t-----------\t\t------------")
    for i in range(len(first_names)):
        print(first_names[i] + "\t\t\t" + last_names[i] + "\t\t\t" +
courses[i] + "\t\t\t" + str(course_marks[i]) +"%")

#Creating a function to be able to sort the class in some desired order
def sort_class(sortingMethod, first_names1, last_names1, courses1,
course_marks1):
    #Making the variables global so we can use them inside the function
and make edits to them
    global first_names
    global last_names
    global courses
    global course_marks

    #Create temporart lists to store the sorted
    sortedFname = []
    sortedLname = []
    sortedCourses = []
    sortedMarks = []

    #Sorting alphabetically by first names
    if sortingMethod == "1":
        sortedList = sorted(first_names1)

        #Sorting the desired way then order the sorted lists by appending
in the desired order
        for student in sortedList:
            index = first_names1.index(student)

            sortedFname.append(first_names1[index])
            sortedLname.append(last_names1[index])
            sortedCourses.append(courses1[index])
            sortedMarks.append(course_marks1[index])

        #Setting the sorted lists equal to our originala= lar video
        first_names = sortedFname
        last_names = sortedLname
        courses = sortedCourses
        course_marks = sortedMarks

        #Update user that the class has been sotred
        print("Class Has Been Sorted!")

    #Sorting alphabetically by last name
    elif sortingMethod == "2":
        sortedList = sorted(last_names1)

        for student in sortedList:
            index = last_names1.index(student)

            sortedFname.append(first_names1[index])
            sortedLname.append(last_names1[index])
            sortedCourses.append(courses1[index])
            sortedMarks.append(course_marks1[index])

        first_names = sortedFname
        last_names = sortedLname
        courses = sortedCourses
        course_marks = sortedMarks
        print("Class Has Been Sorted!")

    #Sorting class by course code
    elif sortingMethod == "3":
        sortedList = sorted(courses1)

        for student in sortedList:
            index = courses1.index(student)

            sortedFname.append(first_names1[index])
            sortedLname.append(last_names1[index])
            sortedCourses.append(courses1[index])
            sortedMarks.append(course_marks1[index])

        first_names = sortedFname
        last_names = sortedLname
        courses = sortedCourses
        course_marks = sortedMarks
        print("Class Has Been Sorted!")

    #Sorting class by highest to lowest marks
    elif sortingMethod == "4":
        sortedList = sorted(course_marks1)
        sortedList.reverse()

        for student in sortedList:
            index = course_marks1.index(student)

            sortedFname.append(first_names1[index])
            sortedLname.append(last_names1[index])
            sortedCourses.append(courses1[index])
            sortedMarks.append(course_marks1[index])

        first_names = sortedFname
        last_names = sortedLname
        courses = sortedCourses
        course_marks = sortedMarks
        print("Class Has Been Sorted!")

    else:
            #If one of the sorting methods is not entered, then say they must
    enter one of the options, then call the function again until they finally
    sort correctly
            print("\nYou have to choose one of the sorting methods!")

            sort_class(input("\nWhat Order would you like to sort the class
    in?\n\t1 - Alphabetically By First Name\n\t2 - Alphabeticlly By Last
    Name\n\t3 - Alphabeticlly By Course Code\n\t4 - Highest To Lowest By
    Course Mark\n"), first_names, last_names, courses, course_marks)

    #While loop to run program and allow user to view the commands and run
    them
    while True:
        #Provide the user with options
        action = input("\nWelcome to the classroom! What would you like to
    do?\n\t1 - Add Student To Class\n\t2 - Remove Student From Class\n\t3 -
    Sort The Class In An Order\n\t4 - View The Class\n\t5 - Exit The
    Program\n")

        #If they want to add a person
        if action == "1":
            add_student(input("What is the student's first
    name?\n\t").upper(),input("What is the student's last name?\n\t").upper(),
    input("What course is the student's enrolled in\n\t").upper(),
    int(input("What is the student's course mark?\n\t")))
        #If they want to remove a student
        elif action == "2":
            remove_student(input("What is the first name of the student you
    wish to remove?").upper())
        #If they want to Sort the class
        elif action == "3":
            sort_class(input("\nWhat Order would you like to sort the class
    in?\n\t1 - Alphabetically By First Name\n\t2 - Alphabeticlly By Last
    Name\n\t3 - Alphabeticlly By Course Code\n\t4 - Highest To Lowest By
    Course Mark\n"), first_names, last_names, courses, course_marks)
        #If they want to view the class
        elif action == "4":
            view_class()
       #To exit the program
        elif action == "5":
            break
        #If the user enters something that doesn't exist, tell them then allow
    them to go again
        else:
            print("You have to pick one of the options LOL, eg. 1, 2, 3")
```