```
import statistics as st
import pandas as pd
chemistry_studs = [] #list to save all chemistry students
biology_studs = [] #list to save all biology students
math_studs = [] #list to save all math students
physics_studs = [] #list to save all physics students


chemistry_scores = [] #list to save all chemistry students scores
biology_scores = [] #list to save all biology students
math_scores = [] #list to save all math students
physics_scores = [] #list to save all physics students

all_subjects = ["chemistry", "biology", "math", "physics"] #list of all subject offered in school
all_students = [chemistry_studs, biology_studs, math_studs, physics_studs]
all_scores = [chemistry_scores, biology_scores, math_scores, physics_scores]


#grading parameters
grade_a= 80
grade_b = 70
grade_c = 60
grade_d = 50
grade_f = 49

a = "grade A"
b = "grade B"
c = "grade C"
d = "grade D"
f = "grade F"

#creating a function that assigns students a grade according to subject score
def grade(score, subject):
    if score >= grade_a:
        return f"{a}"
    elif score >= grade_b and score < grade_a:
        return f"{b}"
    elif score >= grade_c and score < grade_b:
        return f"{c}"
    elif score >= grade_d and score < grade_c:
        return f"{d} retake exam"
    else:
        return f"{f}"



def data_input(): #function to input students data into school record
    course_name = input("enter course name") #enter student name
    course_name = course_name.lower() #converts course name entered to lowercase
    if course_name in all_subjects: #checking if course name in subjects offered in school
        subj_position = all_subjects.index(course_name) #seclecting subject
        course = all_students[subj_position]
        stud_name = input("enter the students name") #enter student name(s) record to add
        stud_name = stud_name.lower() #converting name entered to lowercase
        stud_name = stud_name.split(",") #seperating names entered individually
        for sd in stud_name: #iteratin throung names entered
            course.append(sd) #adding each name to course list

        course_score = all_scores[subj_position] #selecting subject ot add students score to
        exam_score = input(f"enter student(s) {course_name} score") #enter the student(s) course score
        exam_score = exam_score.split(",") #seperating student(s) score
        for sc in exam_score: #iterating through records
            sce = float(sc) #conveting score entered into float
            course_score.append(sce) #adding score to couser score list
    else:
        print(f"{course_name} does not exist in subjects offered") #prints as output if subject does not exist
```

```
# Create a function that returns student grades and overall average
def students_average(stud_name):
    stud_name = input("Enter the student's name: ").lower()
    student_found = False  # Check if the student is found in data
    student_status = []  # List to store the status for each subject
    subjects_taken = []  # List to store the subjects taken by the student
    average = [] #empty list to add hold student scores

    #iterate through students list to find student in school records
    for studs, subject, scores in zip(all_students, all_subjects, all_scores):
        for stud, score in zip(studs, scores):
            if stud_name == stud:
                student_found = True
                average.append(score)  # Add student's scores to a list
                grading = grade(score, subject) #call student grade function
                student_grade.append(grading)
                print(grading)

                # Determine student's status for each subject
                if score < grade_d:
                    student_status.append("failed")
                    subjects_taken.append(subject)
                elif score >= grade_d and score < grade_c:
                    student_status.append(f"retake {subject}")
                    subjects_taken.append(subject)
                else:
                    student_status.append("pass")

    if student_found:
        # Calculate the mean score for all subjects
        mean_score = st.mean(average)
        print(f"{stud_name}'s average score is {mean_score}")
        # Determine the overall student status
        if "failed" in student_status:
            student_status = "failed"
        elif any(subject.startswith("retake") for subject in student_status):
            student_status = ", ".join(subjects_taken) + " need to be retaken"
        else:
            student_status = "pass"

        print(f"{stud_name}'s Overall Status is {student_status}")
    else:
        print(f"{stud_name} does not exist in records")
```

```
# Create a function that returns student grades and overall average
def students_overall_status(stud_name):
    stud_name = input("Enter the student's name: ").lower()
    student_found = False  # Check if the student is found in data
    student_status = []  # List to store the status for each subject
    subjects_taken = []  # List to store the subjects taken by the student
    average = [] #empty list to add hold student scores
    student_grade = [] #list to hold student grade in each subjec
    student_s = [] #list to hold subject taken by student
    student_record = dict() #empty dictionary

    #iterate through students list to find student in school records
    for studs, subject, scores in zip(all_students, all_subjects, all_scores):
        for stud, score in zip(studs, scores):
            if stud_name == stud:
                student_found = True
                student_s.append(subject)
                average.append(score)  # Add student's scores to a list
                grading = grade(score, subject)
                student_grade.append(grading)

                # Determine student's status for each subject
                if score < grade_d:
                    student_status.append("failed")
                    subjects_taken.append(subject)
                elif score >= grade_d and score < grade_c:
                    student_status.append(f"retake {subject}")
                    subjects_taken.append(subject)
                else:
                    student_status.append("pass")

    if student_found:
        # Calculate the mean score for all subjects
        mean_score = st.mean(average)
        # Determine the overall student status
        if "failed" in student_status:
            student_status = "failed"
        elif any(subject.startswith("retake") for subject in student_status):
            student_status = ", ".join(subjects_taken) + " need to be retaken"
        else:
            student_status = "pass"
        student_r = dict #creating empty dictionary
        for sub, grd in zip(student_s, student_grade):
            student_r[sub] = grd
        student_r["average"] = mean_score #adding student average to dictionary
        student_r["Exam status"] = student_status #addign student overall exam status to dictionary
        student_record[stud_name] = student_r #adding student subject dictionary to student name
        #creating data frame using pandas
        df = pd.DataFrame.from_dict(student_record, orient='index')
        print(df)
    else:
        print(f"{stud_name} does not exist in records")



#calls fucntion that inputs students subjects, name and score
data_input()

     enter course name math
     enter the students name mike
     enter student(s) math score 54


status = students_average(stud_name)
status

     Enter the student's name:  saleem
     grade A
     grade A
     grade A
     grade B
     saleem's average score is 88.5
     saleem's Overall Status is pass


overall = students_overall_status(stud_name)
overall

     Enter the student's name:  saleem
            chemistry  biology     math  physics  average Exam status
     saleem   grade A  grade A  grade A  grade B     88.5        pass
```