



INDIVIDUAL ASSIGNMENT
TECHNOLOGY PARK MALAYSIA
CT108-3-1-PYP
PYTHON PROGRAMMING (PYP)
INTAKECODE
APD1F2009CS-IS-SE-APU1F2009CS-IS-SE

HAND OUT DATE: 2nd WEEK

HAND IN DATE: 12th WEEK

**Online Submission Time before = 7: 00 PM Malaysia
Time**

Lecturer : Tanveer Khaleel Shaikh

Table of Contents

Introduction	3
Assumption	4
Design - Pseudocode.....	5
Design – Flowchart.....	36
Sample Python Code	66
Screenshot of Input/Output	70
Additional Features.....	105
Conclusion.....	106
References	107

Introduction

Real Champions Sport Academy is one of the fast-growing Sport Centre in Malaysia, The Sport Centre requires a system/program to store the information about their coaches since it is difficult and inefficient for the admin staff to retrieve information about their coaches' records, sport's records, and student's records. The Sport Centers are located in various location and each of the Sport Centre is managed by an admin staff, all the sports' record, coaches' record, and students' records are managed and maintained by the Sport Centre Head Quarter Human Resources Department. The current filing system of records is by categorizing the coaches by their location and the problem with this is when searching for a record, the record could not be found easily as it might have been misplace in another Sport Centre.

To maintain quality and consistency, one coach is only allowed to coach one sport, which is their specialized sport, these sports include Swimming, Badminton, Football, Archery, Gymnastics, Volleyball, Basketball, Cricket, Tennis, and Table Tennis. Information needed about the coaches are Coach ID, Name, Date Joined, Date Terminated, Hourly Rate, Phone Address, Sport Centre Code, Sport Centre Name, Sport Code, Sport Name, and a Rating. These are necessary information that have to be included in the program.

The benefit of computerize this system is that it will be easier for the admin staff to manage and search for a file about coach, it will also include a system where students can register online and by registering, they can get information about Sport Centres, Sports, Coaches, Coach Performances, Sport Fees, Schedules of Sports, etc. The registration system will ask for the student's username, password, email, date of birth, place of birth, etc.

Assumption

1. It is assumed the system is able to ask the user to login to Student, Public, or Admin Account
2. It is assumed the admin can add records of :
 - a. Coaches
 - b. Sports
 - c. Sport Schedule
3. It is assumed the admin can display all records of:
 - a. Coaches
 - b. Sports
 - c. Registered Students
4. It is assumed the admin can search a specific records of:
 - a. Coaches by Coach ID
 - b. Coaches by overall Performance (Rating)
 - c. Sports by Sport ID
 - d. Students by Student ID
5. It is assumed the admin can sort and display records of:
 - a. Coaches in ascending order by names
 - b. Coaches hourly pay rate in ascending order
 - c. Coaches overall performance in ascending order
6. It is assumed the admin can modify records:
 - a. Coaches
 - b. Sports
 - c. Sport Schedule
7. It is assumed the public user can view details of:
 - a. Sports
 - b. Sport Schedules
8. It is assumed the public user can register as a new student
9. It is assumed the student user can view details of:
 - a. Coaches
 - b. Self-Record
 - c. Registered Sport Schedule
10. It is assumed the student user can modify self-record
11. It is assumed the student can provide feedback and star to coach

Design – Pseudocode

#MarcellAgungW

#TP058650

1. Main Menu Function (def main_menu())

Begin

declare profile

while True:

display "Select a Profile : 1. Admin 2. Student 3. Public (Un-Registered)

4. Exit Program "

read profile

if (profile == "1"):

call admin_login()

break

elif (profile == "2"):

call student_login()

break

elif (profile == "3"):

call public_function()

break

elif(profile == "4"):

display ""

break

else:

display "Invalid Input"

End if

End while

End

A>>>Public Functions-----

2. Public Function : General (def public_function())

Begin

declare option

while True:

display "Please select Public function : 1. View Details 2. Register as

Student 3. Exit"

read option

if (option == "1"):

```

        call view_detail_public()
        break
    elif (option == "2"):
        call register()
        break
    elif (option == "3"):
        call main_menu()
        break
    else:
        display "Invalid Input, Please Try Again"
    End if
End while
End

```

3. Public Function, Register (def register())

Begin

```

    declare student_data, student_registration
    student_data = ["studentID", "username", "email", "place of birth", "dateofbirth",
"password", "confirmedpassword"]
    display "Student Registration Enter 'b' on StudentID or Password to go back"
    while True:
        display "Student ID, Username, Email, Place of Birth, Date of Birth,
Password, Confirm your Password"
        read
        student_data[0], student_data[1], student_data[2], student_data[3], student_data[4], student_
data[5], student_data[6]
        if (student_data[5] == student_data[6])
            OPENFILE "studentdatabase.txt" for Append
            WRITEFILE "studentdatabase.txt",
"student_data[0]+"\""+student_data[1]+"\""+student_data[2]+"\""+student_data[3]+"\""+st
udent_data[4]+"\""+student_data[5]"
            CLOSEFILE "studentdatabase.txt"
            display "Registration Successful, Proceeding to Student Login

```

Function"

```

        call student_login()
        break
    elif (student_data[0] == "b" or student_data[5] == "b")
        call public_function()
        break
    else
        display "Username/Password doesn't match, Please try again or
type 'b' on StudentID or Password to go back"

```

```
        End if
    End while
End
```

4. Public Function, View Details (def view_detail_public())

```
Begin
    declare record
    display "Select a detail to be displayed : a. Sport b. Sport Schedule c. Back to
main menu"
    read record
    while True:
        if (record == "a"):
            call dsport_record()
            break
        elif (record == "b"):
            call sport_schedule()
            break
        elif (record == "c"):
            call public_function()
            break
        else:
            print ("Invalid Input, please select a, b, c, or d.")
    End if
End while
End
```

5. Public Function, View Details of : a. Sport (def dsport_record())

```
Begin
    declare sport_data, sports, order
    OPENFILE "sportdatabase.txt"
    for sports in sport_data
        order = sports.split()
        display " Sport Code \t: ",order[0])
    display "Sport Type \t: ",order[1])
    display
```

```
"Sport Center \t: ",order[2])
    display "Coach \t\t: ",order[3])
    display "Sport Fee \t: ",order[4])
```

```
        End for
    CLOSEFILE "sportdatabase.txt"
End
```

6. Public Function, View Details of : b. Sport Schedule (def sport_schedule())

```
Begin
    declare sportschedule, order
    OPENFILE "scheduledatabase.txt"
    for schedule in sportschedule:
        order = schedule.split()
        display "Sport Type \t: ", order[0])
    display "Sport Date \t: ",order[1])
        display "Sport Time \t: ", order[2])
        display "Sport Coach \t: ",order[3])
    End for
    CLOSEFILE "scheduledatabse.txt"
End
```

B>>>Student Functions-----

7. Student Login Function (def student_login())

```
Begin
    declare studentLogin, login_success, studentnm, studentpw, student_records,
student_data, order
    studentLogin = False
    display "Student Login, Type 'b' on Username or Password to go back"
    while (studentLogin == False)
        OPENFILE "studentdatabase.txt"
        display "Enter student username, Enter student password"
        read studentnm, studentpw
        if (studentnm == "b" or studentpw == "b")
            call main_menu()
            break
        End if
        for student_records in student_data:
            order = student_records.split()
            if (studentnm == order[1] and studentpw == order[5])
                display "Login Successful"
```



```

        studentLogin = True
        CLOSEFILE "scheduledatabase.txt"
        call student_function()
        break
    End if
End for
if (studentLogin == False)
    display "Invalid Username or Password, Enter 'b' to go back"
End if
End while
End
```

8. Student Function : General (def student_function())

```

Begin
    declare option
    display "Please select Student function : 1. View Details 2. Modify Self Record 3.
Provide feedback and Star to Coach 4. Back to main menu "
    read option
    while True:
        if (option == 1):
            call view_details_student()
            break
        elif (option == 2):
            call modify_self_record()
            break
        elif (option == 3):
            call feedback_and_star()
            break
        elif (option == 4):
            call main_menu()
            break
        else:
            display "Invalid Input, Please Try Again"
    End if
End while
End
```

9.Student Function, Provide Feedback and Star to Coach (def feedback_and_star())

```

Begin
    declare coachname, found, rating, totalrating, coaches, names, counter, order
    OPENFILE "coachdatabase.txt"
    display "Provide Feedback and Star to Coach, Enter 'b' to go back"
    display "List of Coaches: "
```

```

for names in coaches
    order = names.split()
    display order[1]
End for
CLOSEFILE "coachdatabase.txt"
found = False
while True
    display "Enter a Coach Name, or Enter 'b' to go back"
    read coachname
    OPENFILE "coachdatabase.txt"
    for coaches in "coachdatabase.txt"
        order = coaches.split()
        if (coachname == order[1])
            found = True
            display " Coach ID \t\t: ",order[0])
            display "Coach Name \t\t: ",order[1])
            display "Date Joined \t\t: ",order[2])
            display "Date Terminated \t: ",order[3])
            display "Hourly Rate (Rm/H) \t: ",order[4])
            display "Phone \t\t\t: ",order[5])
            display "Address \t\t: ",order[6])
            display "Sport Center Code \t: ",order[7])
            display "Sport Center Name \t: ",order[8])
            display "Sport Code \t\t: ",order[9])
            display "Sport Name \t\t: ",order[10])
            display "Rating \t\t\t: ",order[11])
            display "Total Reviewers \t: ", order[12])
            CLOSEFILE "coachdatabase.txt"
            break
        End if
    End for
    if (found == True)
        OPENFILE "coachdatabase.txt"
        counter = 0
        coaches = coachdata.readlines()
        for lines in coaches
            counter += 1
            if coachname in lines:
                order = lines.split()
                display "Enter your Rating (1-5): "
                read rating
                if (rating >= 1 and rating <= 5):

```

```

totalrating = rating + float(order[11])

* float(order[12])

order[12] = float(order[12]) + 1
order[11] = totalrating/float(order[12])
order[12] = str(order[12])
order[11] = str(order[11])
else:
    display "Invalid Input, Please enter
1-5"

    call student_function()
    break
order[12] = order[12] + "\n"
success = "\t".join(order)
coaches[counter-1] = success
OPENFILE "coachdatabase.txt" for write as
file:

    file.writelines(coaches)
    display "Rating Successfully Updated,
Updated Rating : ", order[11]

    break
CLOSEFILE "coachdatabase.txt"
elif (coachname == "b")
    call student_function()
    break
else
    display "Invalid Coach Name"
End if
End while
End

10. Student Function, Modify Self Record (def modify_self_record())
Begin
    declare login, counter, inputid, inputpw, verify, student_data, student_modify,
order
    student_data = ["studentID", "username", "email", "place of birth", "dateofbirth",
"password", "confirmedpassword"]
    display "Student Modify Function Enter 'b' in Username or Password to go back."
    login = False
    counter = 0
    while (login == False)
        display "Confirm your username, Confirm your password"
        read inputid, inputpw
        for password in student_modify:

```

```

        order = password.split()
        if (inputid == order[1] and inputpw == order[5])
            login = True
            break
        elif inputid == "b" or inputpw == "b"
            call student_function()
            break
        End if
    End for
    if (login == False)
        display "Invalid Username or Password, Please Retry or Enter 'b'
to go back"
    End If
End while
while (login = True)
    student_modify = open("studentdatabase.txt")
    student_modify_line = student_modify.readlines()
    student_data[0] = input(" Student ID \t: ")
    student_data[1] = input("Username \t: ")
    student_data[2] = input("Email \t\t: ")
    student_data[3] = input("Place of Birth \t: ")
    student_data[4] = input("Date of Birth \t: ")
    student_data[5] = input("Password \t: ")
    display "Enter an input : '1' to Save and exit program '2' to Save and go
back to Admin Function 'anything' to redo 'b' to go back and cancel modify"
    read verify
    if verify == "1"
        print (" Record Modified Successfully")
        student_modify_line[counter] =
student_data[0]+"\\t"+student_data[1]+"\\t"+student_data[2]+"\\t"+student_data[3]+"\\t"+st
udent_data[4]+"\\t"+student_data[5]+" "
        student_modify = open("studentdatabase.txt", "w")
        student_modify.writelines(student_modify_line)
        student_modify.close()
        break
    elif verify == "2"
        print (" Record Modified Successfully")
        student_modify_line[counter] =
student_data[0]+"\\t"+student_data[1]+"\\t"+student_data[2]+"\\t"+student_data[3]+"\\t"+st
udent_data[4]+"\\t"+student_data[5]+" "
        student_modify = open("studentdatabase.txt", "w")
        student_modify.writelines(student_modify_line)
        student_modify.close()

```

```
        student_function()
        break
    elif verify == "b"
        student_modify.close()
        call student_function()
        break
    else:
        login = True
    End if
End while
End
```

11. Student Function, View Details (def view_details_student())

```
Begin
    declare record
    while True:
        display "Select a detail to be displayed : a. Coach b. Self-Record c. Sport
Schedule d. Go Back"
        read record
        if record == "a"
            call dcoach_record()
            break
        elif (record == "b"):
            call self_record_detail()
            break
        elif (record == "c"):
            call sport_schedule()
            break
        elif (record == "d"):
            call student_function()
            break
        else:
            print (" Invalid Input, please select a, b, c, or d.")
    End if
End while
End
```

12. Student Function, View Details of : a. Coach (def dcoach_record())

```
Begin
    OPENFILE "coachdatabase.txt"
    for coaches in coach_data:
```

```

        order = coaches.split()
        display " Coach ID \t\t: ",order[0])
        display "Coach Name \t\t: ",order[1])
        display "Date Joined \t\t: ",order[2])
        display "Date Terminated \t: ",order[3])
        display "Hourly Rate (Rm/H) \t: ",order[4])
        display "Phone \t\t\t: ",order[5])
        display "Address \t\t: ",order[6])
        display "Sport Center Code \t: ",order[7])
        display "Sport Center Name \t: ",order[8])
        display "Sport Code \t\t: ",order[9])
        display "Sport Name \t\t: ",order[10])
        display "Rating \t\t\t: ",order[11])
        display "Total Reviewers \t: ",order[12])
    End for
    CLOSEFILE "coachdatabase.txt"
End

```

13. Student Function, View Details of : b. Self-Record (def self_record_detail())

```

Begin
    declare confirm, inputid, inputpw, order, selfrecord
    confirm = 1
    while (confirm == 1)
        display "Please Confirm your Username, Please Confirm your Password"
        read inputid, inputpw
        OPENFILE "studentdatabase.txt"
        for password in "studedntdatabase.txt"
            order = password.split()
            if (inputpw == order[5] and inputid == order[1])
                display "Student ID \t\t: ", order[0])
                display "Username \t\t: ", order[1])
                display "Email \t\t\t: ", order[2])
                display "Place of Birth \t\t: ", order[3])
                display "Date of Birth \t\t: ", order[4])
                CLOSEFILE "studentdatabase.txt"
                confirm = 0
            elif (inputid == "b" or inputpw == "b")
                confirm = 0
                call view_details_student()
            else
                display "Invalid Password, Please re-enter your password,
or type 'b' to go back"
            end if
        end for
    end while
End

```

```
                End if
            End for
        End while
    End
```

14. Student Function, View Details of : c. Registered Sport Schedule (def sport_schedule())

```
Begin
    declare sportschedule, order
    OPENFILE "scheduledatabase.txt"
    for schedule in sportschedule
        order = schedule.split()
        display "Sport Type \t: ", order[0]
        display "Sport Date \t: ",order[1]
        display "Sport Time \t: ",order[2]
        display "Sport Coach \t: ",order[3]
    End for
    CLOSEFILE "scheduledatabase.txt"
End
```

C>>Admin Functions-----

15. Admin Login Function (def admin_login())

```
Begin
    declare ad_nm, ad_pw, adLoginnm, adLoginpw
    ad_nm = "Admin", ad_pw = "Admin69"
    while True
        display "Enter Admin username and Admin password"
        read adLoginnm, adLoginpw
        if (adLoginnm == "b" or adLoginpw == "b")
            call main_menu()
            break
        elif (adLoginnm == ad_nm or adLoginpw == ad_pw)
            display "Login Successful"
            call admin_function()
            break
        else
            display "Invalid Username or Password"
```

```
        end if
    end while
End
```

16. Admin Function : General (def admin_function())

Begin

```
    declare option
    while True:
        display " Please select Admin function : 1. Add Records 2. Display Records 3.
Search Spesific Records 4. Sort and Display Records 5. Modify Records 6. Back to
main menu  ")
```

```
        read option
        if (option == 1):
            call add_records()
            break
        elif (option == 2):
            call display_records()
            break
        elif (option == 3):
            call search_records()
            break
        elif (option == 4):
            call sortdisplay_records()
            break
        elif (option == 5):
            call modifyRecords()
            break
        elif (option == 6):
            call main_menu()
            break
    else:
        display " Invalid Input, Please Try Again")
    End if
End while
```

End

17. Admin Functoin, Modify Records (def modifyRecords())

Begin

```
    declare record
    while True:
```



```

        display "Select a Record to be Modified : a. Coach b. Sport c. Sport
Schedule d. Go Back ")
        read record
        if (record == "a"):
            call modifyCoach()
            break
        elif (record == "b"):
            call modifySport()
            break
        elif (record == "c"):
            call modifySportSchedule()
            break
        elif (record == "d"):
            call admin_function()
            break
        else:
            print (" Invalid Input, please select a, b, c, or d.")
    End if
End while
End

```

18. Admin Function, Modify Records of : a. Coach (def modifyCoach())

```

Begin
    declare line, verify, counter, coach_data, data,coach_modify_line
    display "Modify Records of Coach"
    counter = 0
    OPENFILE "coachdatabase.txt"
    for data in coach_data:
        display "[" ,counter+1,"]", data
        counter += 1
    End for
    CLOSEFILE "coachdatabase.txt"
    coach_data = ["ID", "name", "datejoined", "dateterminated", "hourlyrate",
"phone", "address", "SPcode", "SPname", "sportcode", "sportname", "rating"]
    while True:
        display "Select a line to be modified"
        read line
        OPENFILE "coachdatabase.txt" as coach_modify_line
        coach_modify_line = coach_modify.readlines()
        coach_data[0] = input ("Coach ID \t\t: ")
        coach_data[1] = input ("Coach Name \t\t: ")
        coach_data[2] = input ("Date Joined \t\t: ")

```

```

coach_data[3] = input ("Date Terminated \t: ")
coach_data[4] = input ("Hourly Rate (RM/H)\t: ")
coach_data[5] = input ("Phone \t\t\t: ")
coach_data[6] = input ("Address \t\t: ")
coach_data[7] = input ("Sport Center Code \t: ")
coach_data[8] = input ("Sport Center Name \t: ")
coach_data[9] = input ("Sport Code \t\t: ")
coach_data[10] = input ("Sport Name \t\t: ")
coach_data[11] = input ("Rating \t\t\t: ")

display "Enter an input : '1' to Save and exit program '2' to Save and add
another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
back ")

read verify
if verify == "1"
    display "Record Modified Successfully")
    coach_modify_line[line-1] =
coach_data[0]+\t\t+coach_data[1]+\t\t+coach_data[2]+\t\t+coach_data[3]+\t\t+coach
_data[4]+\t\t+coach_data[5]+\t\t+coach_data[6]+\t\t+coach_data[7]+\t\t+coach_data[
8]+\t\t+coach_data[9]+\t\t+coach_data[10]+\t\t+coach_data[11]+\t\t"
    coach_modify = open("coachdatabase.txt", "w")
    coach_modify.writelines(coach_modify_line)
    coach_modify.close()
    break
elif verify == "2"
    display "Record Modified Successfully")
    coach_modify_line[line-1] =
coach_data[0]+\t\t+coach_data[1]+\t\t+coach_data[2]+\t\t+coach_data[3]+\t\t+coach
_data[4]+\t\t+coach_data[5]+\t\t+coach_data[6]+\t\t+coach_data[7]+\t\t+coach_data[
8]+\t\t+coach_data[9]+\t\t+coach_data[10]+\t\t+coach_data[11]+\t\t"
    coach_modify = open("coachdatabase.txt", "w")
    coach_modify.writelines(coach_modify_line)
    coach_modify.close()
elif verify == "3"
    display "Record Modified Successfully")
    coach_modify_line[line-1] =
coach_data[0]+\t\t+coach_data[1]+\t\t+coach_data[2]+\t\t+coach_data[3]+\t\t+coach
_data[4]+\t\t+coach_data[5]+\t\t+coach_data[6]+\t\t+coach_data[7]+\t\t+coach_data[
8]+\t\t+coach_data[9]+\t\t+coach_data[10]+\t\t+coach_data[11]+\t\t"
    coach_modify = open("coachdatabase.txt", "w")
    coach_modify.writelines(coach_modify_line)
    coach_modify.close()
    admin_function()
    break

```

```

        elif verify == "b"
            call modifyRecords()
            break
        End if
    End while
End

```

19. Admin Function, Modify Records of : b. Sport (def modifySport())

Begin

```

    declare line, verify, sport_data, counter, line
    display "Modify Records of Sport"
    OPENFILE "sportdatabase.txt"

    counter = 0
    for sports in sport_data:
        display "[" ,counter+1,"]",sports)
    End for
    CLOSEFILE "sportdatabase.txt"
    sport_data = ["id", "name", "center", "coach", "fees"]
    while True:
        display "Select a line to be modified : "
        read line
        OPENFILE "sportdatabase.txt" as sport_modify_line
        sport_modify_line = sport_modify.readlines()
        sport_data[0] = input("Sport ID\t: ")
        sport_data[1] = input("Sport Name \t: ")
        sport_data[2] = input("Sport Center \t: ")
        sport_data[3] = input("Sport Coach \t: ")
        sport_data[4] = input("Sport Fees \t: ")
        display " Enter an input : '1' to Save and exit program '2' to Save and add
another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
back " )

        read verify
        if verify == "1"
            print (" Record Modified Successfully")
            sport_modify_line[line] =
sport_data[0]+" \t\t"+sport_data[1]+" \t"+sport_data[2]+" \t"+sport_data[3]+" \t"+sport_data[4]+" "
            sport_modify = open("sportdatabase.txt", "w")
            sport_modify.writelines(sport_modify_line)
            sport_modify.close()
            break

```

```

        elif verify == "2"
            print (" Record Modified Successfully")
            sport_modify_line[line] =
sport_data[0]+"\\t\\t"+sport_data[1]+"\\t"+sport_data[2]+"\\t"+sport_data[3]+"\\t"+sport_data[4]+" "

            sport_modify = open("sportdatabase.txt", "w")
            sport_modify.writelines(sport_modify_line)
            sport_modify.close()
            break
        elif verify == "3"
            print (" Record Modified Successfully")
            sport_modify_line[line] =
sport_data[0]+"\\t\\t"+sport_data[1]+"\\t"+sport_data[2]+"\\t"+sport_data[3]+"\\t"+sport_data[4]+" "

            sport_modify = open("sportdatabase.txt", "w")
            sport_modify.writelines(sport_modify_line)
            sport_modify.close()
            admin_function()
            break
        elif verify == "b"
            call modifyRecords()
            break
    End if
End while
End

```

20. Admin Function, Modify Records of : c. Sport Schedule (def modifySportSchedule())
Begin

```

declare line, verify, counter, schedule_data
display "Modify Records of Sport Schedules"
OPENFILE "scheduledatabase.txt"
counter = 0
for schedules in schedule_data:
    display "[" ,counter+1,"]",schedules)
counter += 1
End for
CLOSEFILE "scheduledatabase.txt"
schedule_data = ["sport", "time", "coach"]
while True:
    display "Select a line to be modified"
    read line
    schedule_modify = open("scheduledatabase.txt")

```

```

schedule_modify_line = schedule_modify.readlines()
schedule_data[0] = input("Sport \t: ")
schedule_data[1] = input("Date \t: ")
    schedule_data[2] = input("Time \t: ")
    schedule_data[3] = input("Coach \t: ")
    display " Enter an input : '1' to Save and exit program '2' to Save and add
another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
back ")
        read verify
        if verify == "1"
            print (" Record Modified Successfully")
            schedule_modify_line[line] =
schedule_data[0]+"\\t"+schedule_data[1]+"\\t"+schedule_data[2]+schedule_data[3]+"\\n"
            schedule_modify = open("scheduledatabase.txt", "w")
            schedule_modify.writelines(schedule_modify_line)
            schedule_modify.close()
            break
        elif verify == "2"
            print (" Record Modified Successfully")
            schedule_modify_line[line] =
schedule_data[0]+"\\t"+schedule_data[1]+"\\t"+schedule_data[2]+schedule_data[3]+"\\n"
            schedule_modify = open("scheduledatabase.txt", "w")
            schedule_modify.writelines(schedule_modify_line)
            schedule_modify.close()
            break
        elif verify == "3"
            print (" Record Modified Successfully")
            schedule_modify_line[line] =
schedule_data[0]+"\\t"+schedule_data[1]+"\\t"+schedule_data[2]+schedule_data[3]+"\\n"
            schedule_modify = open("scheduledatabase.txt", "w")
            schedule_modify.writelines(schedule_modify_line)
            schedule_modify.close()
            admin_function()
            break
        elif verify == "b"
            call modifyRecords()
            break
    End if
End while
End

```

21. Admin Function, Sort and Display Records (def sortdisplay_records())

```

Begin
    declare record
    while True:
        display "Select a Record to be Sorted and Displayed : a. Coach in
Ascending Order by Names b. Coaches Hourly Pay Rate in Ascending Order c. Coaches
Overall Performance in Ascending Order d. Go Back "
        read record
        if (record == "a"):
            call coachNames()
            break
        elif (record == "b"):
            call coachPayRate()
            break
        elif (record == "c"):
            call coachRating()
            break
        elif (record == "d"):
            call admin_function()
            break
        else:
            print (" Invalid Input, please select a, b, c, or d.")
    End if
End while
End

```

22. Admin Function, Sort and Display Records of : a. Coaches in Ascending Order by Names (def coachNames())

```

Begin
    declare question, coach_database, sort_details, sort_name, sort_result,
coach_records, readl, i, j
    display "Displaying Coaches by Names in Ascending Order, enter 1 to continue
or 0 to go back:
    read question
    if (question == 1)
        OPENFILE "coachdatabase.txt"
        OPENFILE "sortname.txt" for Write
        for coach_records in coach_database:
            coach_details = coach_records.split()

            sort_name.write(coach_details[1]+"\\t"+coach_details[0]+"\\t"+coach_details[2]+"\\
t"+coach_details[3]+"\\t"+coach_details[4]+"\\t"+coach_details[5]+"\\t"+coach_details[6]+

```

```

"\t"+coach_details[7)+"\t"+coach_details[8)+"\t"+coach_details[9)+"\t"+coach_details[1
0)+"\t"+coach_details[11)+"\t"+coach_details[12]+" ")
    End for
        coach_database.close()
    elif (question == "0"):
        sortdisplay_records()
    End if
    CLOSEFILE "coachdatabase.txt"
    OPENFILE "coachdatabase.txt" as sort_name
    readl = sort_name.readlines()
    n = len(readl)
    x = ""
    for i in range (n-1):
        for j in range (0, n-i-1):
            if readl[j] > readl [j+1]:
                x = readl[j]
                readl[j] = readl[j+1]
                readl[j+1] = x
            End if
        End for
    End for
    OPENFILE "sortname.txt" for Write as file:
    file.writelines (readl)
    CLOSEFILE "sortname.txt"
    sort = open("sortnameresult.txt")
    for sort_records in sort:
        sort_records = sort_records.strip(" ")
        sort_details = sort_records.split()
        display " Coach ID \t\t: ",sort_details[1])
        display "Coach Name \t\t: ",sort_details[0])
        display "Date Joined \t\t: ",sort_details[2])
        display "Date Terminated \t: ",sort_details[3])
        display "Hourly Rate (Rm/H) \t: ",sort_details[4])
        display "Phone \t\t\t: ",sort_details[5])
        display "Address \t\t: ",sort_details[6])
        display "Sport Center Code \t: ",sort_details[7])
        display "Sport Center Name \t: ",sort_details[8])
        display "Sport Code \t\t: ",sort_details[9])
        display "Sport Name \t\t: ",sort_details[10])
        display "Rating \t\t\t: ",sort_details[11])
        display "Total Reviewers \t: ",sort_details[12])
    End for
End

```

23. Admin Function, Sort and Display Records of : b. Coaches Hourly Pay Rate in Ascending Order (def coachPayRate())

Begin

```
    declare question, coach_database, sort_pay, sort_result, coach_records, readl, n, y,
    i, j
```

```
    display "Displaying Coaches by Hourly Pay Rate in Ascending Order, enter 1 to
    continue or 0 to go back:
```

```
    if (question == "1"):
```

```
        OPENFILE "coachdatabase.txt"
```

```
        OPENFILE "sortpay.txt" for Write
```

```
        for coach_records in coach_database:
```

```
            coach_details = coach_records.split()
```

```
    sort_pay.write(coach_details[4]+"\\t"+coach_details[0]+"\\t"+coach_details[1]+"\\t"+coach
    _details[2]+"\\t"+coach_details[3]+"\\t"+coach_details[5]+"\\t"+coach_details[6]+"\\t"+coac
    h_details[7]+"\\t"+coach_details[8]+"\\t"+coach_details[9]+"\\t"+coach_details[10]+"\\t"+c
    oach_details[11]+"\\t"+coach_details[12]+" ")
```

```
    coach_database.close()
```

```
    elif (question == "0"):
```

```
        sortdisplay_records()
```

```
        CLOSEFILE "sortpay.txt"
```

```
        OPENFILE "sortpay.txt"
```

```
        readl = sort_pay.readlines()
```

```
        n = len(readl)
```

```
        y = ""
```

```
        for i in range (n-1):
```

```
            for j in range (0, n-i-1):
```

```
                if readl[j] > readl [j+1]:
```

```
                    y = readl[j]
```

```
                    readl[j] = readl[j+1]
```

```
                    readl[j+1] = y
```

```
        OPENFILE "sortpay.txt" for Write as file:
```

```
            file.writelines (readl)
```

```
            file.close
```

```
        sort_result.close()
```

```
        sort = open("sortpayresult.txt")
```

```
        for sort_records in sort:
```

```
            sort_records = sort_records.strip(" ")
```

```
            sort_details = sort_records.split()
```

```
            display " Coach ID \\t\\t: ",sort_details[1])
```

```
            display "Hourly Rate (Rm/H) \\t: ",sort_details[0])
```



```

display "Coach Name \t\t: ",sort_details[2])
display "Date Joined \t\t: ",sort_details[3])
display "Date Terminated \t: ",sort_details[4])
display "Phone \t\t\t: ",sort_details[5])
display "Address \t\t: ",sort_details[6])
display "Sport Center Code \t: ",sort_details[7])
display "Sport Center Name \t: ",sort_details[8])
display "Sport Code \t\t: ",sort_details[9])
display "Sport Name \t\t: ",sort_details[10])
display "Rating \t\t\t: ",sort_details[11])
display "Total Reviewers \t: ",sort_details[12])
End

24. Admin Function, Sort and Display Records of : c. Coaches Overall Performance in
Ascending Order (def coachRating())
Begin
    declare question, coach_database, sort_rate, sort_result, coach_records, readl, n, z,
    i, j
    display "Displaying Coaches by Overall Performance in Ascending Order, enter 1
to continue or 0 to go back:
    read question
    if (question == "1"):
        OPENFILE "coachdatabase.txt"
        OPENFILE "sortrate.txt" as Write
        for coach_records in coach_database:
            coach_details = coach_records.split()

            sort_rate.write(coach_details[11]+\t"+coach_details[0]+\t"+coach_details[1]+\t
"+coach_details[2]+\t"+coach_details[3]+\t"+coach_details[4]+\t"+\t"+coach_details
[5]+\t"+coach_details[6]+\t"+coach_details[7]+\t"+coach_details[8]+\t"+coach_detai
ls[9]+\t"+coach_details[10]+\t"+coach_details[12]+" ")
        End for
        coach_database.close()
    elif (question == "0"):
        sortdisplay_records()
    End if
    CLOSEFILE "sortrate.txt"
    OPENFILE "sortrate.txt"
    readl = sort_rate.readlines()
    n = len(readl)
    z = ""
    for i in range (n-1):

```

```

for j in range (0, n-i-1):
    if readl[j] > readl [j+1]:
        z = readl[j]
        readl[j] = readl[j+1]
        readl[j+1] = z
OPENFILE "sortrate.txt" for Write as file:
file.writelines (readl)
file.close
sort_result.close()
sort = open("sortrateresult.txt")
for sort_records in sort:
    sort_records = sort_records.strip(" ")
    sort_details = sort_records.split()
    display " Coach ID \t\t: ",sort_details[1])
    display "Rating \t\t\t: ",sort_details[0])
    display "Total Reviewers \t: ",sort_details[12])
    display "Coach Name \t\t: ",sort_details[2])
    display "Date Joined \t\t: ",sort_details[3])
    display "Date Terminated \t: ",sort_details[4])
    display "Hourly Rate (Rm/H) \t: ",sort_details[5])
    display "Phone \t\t\t: ",sort_details[6])
    display "Address \t\t: ",sort_details[7])
    display "Sport Center Code \t: ",sort_details[8])
    display "Sport Center Name \t: ",sort_details[9])
    display "Sport Code \t\t: ",sort_details[10])
    display "Sport Name \t\t: ",sort_details[11])
End for
End

```

25. Admin Function, Search Specific Records (def search_records())

Begin

```

    declare record
    while True:
        display "Select a Record to be Searched : a. Coach by Coach ID b. Coach
by Coach Rating c. Sport by Sport ID d. Go Back ")
        read record
        if (record == "a"):
            call search_coachID()
            break
        elif (record == "b"):
            call search_coachRating()
            break

```

```

        elif (record == "c"):
            call search_studentID()
            break
        elif (record == "d"):
            call admin_function()
            break
    else:
        print (" Invalid Input, please select a, b, c, or d.")
    End if
End while
End

```

26. Admin Function, Search Specific Records of : a. Coach by Coach ID (def search_coachID())

```

Begin
    declare coachID, found, coach_data, coaches, order
    found = False
    print ("Search Specific Records of Coach by Coach ID Enter 'b' to go Back")
    while found == False:
        display "Enter a Coach ID"
        read coachID
        OPENFILE "coachdatabase.txt"
        for coaches in coach_data:
            order = coaches.split()
            if (coachID == order[0])
                display "Coach ID \t\t: ",order[0]
                display "Coach Name \t\t: ",order[1])
            display "Date Joined \t\t: ",order[2])
            display "Date Terminated \t: ",order[3])
            display "Hourly Rate (Rm/H) \t: ",order[4])
            display "Phone \t\t\t: ",order[5])
            display "Address \t\t: ",order[6])
            display "Sport Center Code \t: ",order[7])
            display "Sport Center Name \t: ",order[8])
            display "Sport Code \t\t: ",order[9])
            display "Sport Name \t\t: ",order[10])
            display "Rating \t\t\t: ",order[11])
            display "Total Reviewers \t: ",order[12])
            found = True
        End if
    End for
    CLOSEFILE "coachdatabase.txt"

```

```

        if (coachID == "b")
            found = True
            search_records()
        elif (found == False)
            display "Invalid Coach ID, please retry or Enter 'b' to go back"
        End if
    End while
End

```

27. Admin Function, Search Specific Records of : b. Coach by Overall Performance (def search_coachRating())

```

Begin
    declare coachRating, found, coach_data, coaches, order, counter
    found = False
    counter = 1
    print ("Search Specific Records of Coach by Coach Rating Enter 'b' to go Back")
    while found == False:
        display "Enter a Coach Rating"
        read coachRating
        OPENFILE "coachdatabase.txt"
        for coaches in coach_data
            order = coaches.split()
            if (coachRating == order[11])
                display " [" +str(counter)+ "]"
                display "Coach ID \t\t: ",order[0])
                display "Coach Name \t\t: ",order[1])
                display "Date Joined \t\t: ",order[2])
                display "Date Terminated \t: ",order[3])
                display "Hourly Rate (Rm/H) \t: ",order[4])
                display "Phone \t\t\t: ",order[5])
                display "Address \t\t: ",order[6])
                display "Sport Center Code \t: ",order[7])
                display "Sport Center Name \t: ",order[8])
                display "Sport Code \t\t: ",order[9])
                display "Sport Name \t\t: ",order[10])
                display "Rating \t\t\t: ",order[11])
                display "Total Reviewers \t: ",order[12])
                counter += 1
                found = True
            End if
        End for
    CLOSEFILE "coachdatabase.txt"
End

```

```

        if (coachRating == "b")
            found = True
            search_records()
        elif (found == False)
            display "Invalid Coach ID, please retry or Enter 'b' to go back"
        End if
    End while
End

```

28. Admin Function, Search Specific Records of : c. Sport by Sport ID (def search_sportID())

```

Begin
    declare sportID, found, coach_data, coaches, order
    found = False
    print ("Search Specific Records of Sport by Sport ID Enter 'b' to go Back")
    while found == False:
        display "Enter a Sport ID"
        read sportID
        OPENFILE "sportdatabase.txt"
        for sports in sport_data
            order = coaches.split()
            if (sportID == order[0])
                display " Sport Code \t: ",order[0])
                display "Sport Type \t: ",order[1])
                display "Sport Center \t: ",order[2])
                display "Coach \t\t: ",order[3])
                display "Sport Fee \t: ",order[4])
                found = True
            End if
        End for
        CLOSEFILE "sportdatabase.txt"
        if (sportID == "b")
            found = True
            search_records()
        elif (found == False)
            display "Invalid Coach ID, please retry or Enter 'b' to go back"
        End if
    End while
End

```

29. Admin Function, Display Records (def display_records())

```

Begin
    declare record
    while True:
        display "Select a Record to be Displayed : a. Coach b. Sport c.
Registered Students d. Go Back"
        read record
        if (record == "a"):
            call dcoach_record()
            break
        elif (record == "b"):
            call dsport_record()
            break
        elif (record == "c"):
            call dregisteredstudent_record()
            break
        elif (record == "d"):
            call admin_function()
            break
        else:
            display "Invalid Input, please select a, b, c, or d."
    End if
End while
End

```

30. Admin Function, Display Records of : a. Coaches (def dcoach_record())

```

Begin
    declare coach_data, coaches, order
    OPENFILE "coachdatabase.txt"
    for coaches in coach_data
        order = coaches.split()
        display "\nCoach ID \t\t: ",order[0]
        display "Coach Name \t\t: ",order[1]
        display "Date Joined \t\t: ",order[2]
        display "Date Terminated \t: ",order[3]
        display "Hourly Rate (Rm/H) \t: ",order[4]
        display "Phone \t\t\t: ",order[5]
        display "Address \t\t: ",order[6]
        display "Sport Center Code \t: ",order[7]
        display "Sport Center Name \t: ",order[8]
        display "Sport Code \t\t: ",order[9]
        display "Sport Name \t\t: ",order[10]
        display "Rating \t\t\t: ",order[11]
    End for
End

```

```
        display "Total Reviewers \t: ",order[12]
    End for
    CLOSEFILE "coachdatabase.txt"
End
```

31. Admin Function, Display Records of : b. Sports (def dsport_record())

```
Begin
    declare sport_data, sports, order
    OPENFILE "sportdatabase.txt"
    for sports in sport_data
        order = sports.split()
        display "\nSport Code \t: ",order[0]
        display "Sport Type \t: ",order[1]
        display "Sport Center \t: ",order[2]
        display "Coach \t\t: ",order[3]
        display "Sport Fee \t: ",order[4]
    End for
    CLOSEFILE "sportdatabase.txt"
End
```

32. Admin Function, Display Records of : c. Registered Students (def dregisteredstudent_record())

```
Begin
    declare student_data, students, order
    OPENFILE "studentdatabase.txt"
    for students in student_data
        order = students.split()
        display "\nStudent ID\t: ",order[0]
        display "Username \t: ",order[1]
        display "Email \t\t: ",order[2]
        display "Place of Birth \t: ",order[3]
        display "Date of Birth \t: ",order[4]
        display "Password \t: ",order[5]
    End for
    CLOSEFILE "studentdatabase.txt"
End
```

33. Admin Function, Add Records (def add_records())

```
Begin
    declare record
```

```

while True:
    display "Select a Record to be Added : a. Coach b. Sport c. Sport
Schedule d. Go Back "
    read record
    if (record == "a"):
        call coach_record()
        break
    elif (record == "b"):
        call sport_record()
        break
    elif (record == "c"):
        call sportschedule_record()
        break
    elif (record == "d"):
        call admin_function()
        break
    else:
        display "Invalid Input, please select a, b, c, or d."
    End if
End while
End

```

34. Admin Function, Add Records of : a. Coaches (def coach_record())

```

Begin
    declare confirm, coach_data, verify
    display "Add Records of Coach : "
    confirm = 1
    coach_data = ["ID", "name", "datejoined", "dateterminated", "hourlyrate",
"phone", "address", "SPcode", "SPname", "sportcode", "sportname", "rating"]
    while (confirm == 1)
        display "Coach ID, Coach Name, Date Joined, Date Terminated, Hourly
Rate, Phone, Address, Sport Center Code, Sport Center Name, Sport Code, Sport Name,
Rating"
        read coach_data[1], coach_data[2],..., coach_data[11]
        display " Enter an input : '1' to Save and exit program '2' to Save and add
another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
back "
        read verify
        OPENFILE "coachdatabase.txt" for Append as coach_database
        if (verify == "1")
            confirm = 0
            print ("\nRecord Saved")

```



```

        coach_database.write(coach_data[0]+"\\t"+coach_data[1]+"\\t"+coach_data[2]+"\\t"
+coach_data[3]+"\\t"+coach_data[4]+"\\t"+coach_data[5]+"\\t"+coach_data[6]+"\\t"+coach
_data[7]+"\\t"+coach_data[8]+"\\t"+coach_data[9]+"\\t"+coach_data[10]+"\\t"+coach_data[
11]+"\\t1"+"\\n")

        CLOSEFILE "coachdatabase.txt"
        break
    elif (verify == "2")
        print ("\\nRecord Saved")

        coach_database.write(coach_data[0]+"\\t"+coach_data[1]+"\\t"+coach_data[2]+"\\t"
+coach_data[3]+"\\t"+coach_data[4]+"\\t"+coach_data[5]+"\\t"+coach_data[6]+"\\t"+coach
_data[7]+"\\t"+coach_data[8]+"\\t"+coach_data[9]+"\\t"+coach_data[10]+"\\t"+coach_data[
11]+"\\t1"+"\\n")

        CLOSEFILE "coachdatabase.txt"
    elif (verify == "3")
        confirm = 0
        print ("\\nRecord Saved")

        coach_database.write(coach_data[0]+"\\t"+coach_data[1]+"\\t"+coach_data[2]+"\\t"
+coach_data[3]+"\\t"+coach_data[4]+"\\t"+coach_data[5]+"\\t"+coach_data[6]+"\\t"+coach
_data[7]+"\\t"+coach_data[8]+"\\t"+coach_data[9]+"\\t"+coach_data[10]+"\\t"+coach_data[
11]+"\\t1"+"\\n")

        CLOSEFILE "coachdatabase.txt"
        admin_function()
        break
    elif (verify == "b")
        confirm = 0
        call add_records()
    End if
End while
End

```

35. Admin Function, Add Records of : b. Sports (def sport_record())

Begin

```

    declare confirm, sport_data, verify
    display "Add Records of Sport : "
    confirm = 1
    sport_data = ["id", "name", "center", "coach", "fees"]
    while (confirm == 1)
        display "Sport ID, Sport Name, Sport Center, Sport Coach, Sport Fees"
        read sport_data[1], sport_data[2], ....., sport_data[4]
    
```

```

        display " Enter an input : '1' to Save and exit program '2' to Save and add
another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
back "

        read verify
        OPENFILE "sportdatabase.txt" for Append as sport_database
        if (verify == "1")
            print ("\nRecord Saved")

            sport_database.write("\n"+sport_data[0]+"t"+sport_data[1]+"t"+sport_data[2]+"t"+sport_data[3]+"t"+sport_data[4])
            CLOSEFILE "sportdatabase.txt"
            break
        elif (verify == "2")
            print ("\nRecord Saved")

            sport_database.write("\n"+sport_data[0]+"t"+sport_data[1]+"t"+sport_data[2]+"t"+sport_data[3]+"t"+sport_data[4])
            CLOSEFILE "sportdatabase.txt"
        elif (verify == "3")
            print ("\nRecord Saved")

            sport_database.write("\n"+sport_data[0]+"t"+sport_data[1]+"t"+sport_data[2]+"t"+sport_data[3]+"t"+sport_data[4])
            CLOSEFILE "sportdatabase.txt"
            admin_function()
            break
        elif (verify == "b")
            confirm = 0
            call add_records()

        End if
    End while
End

```

36. Admin Function, Add Records of : c. Sport Schedules (def sportschedule_record())

```

Begin
    declare confirm, schedule_data, verify
    display "Add Records of Sport Schedule : "
    confirm = 1
    schedule_data = ["sport", "time", "coach"]
    while (confirm == 1)
        display "Sport, Date, Time, Coach"
    end while
End

```

```

        read schedule_data[0], schedule_data[1], schedule_data[2],
        schedule_data[3]
        display " Enter an input : '1' to Save and exit program '2' to Save and add
        another Record '3' to Save and go back to Admin Function 'anything' to redo 'b' to go
        back "

        read verify
        OPENFILE "scheduledatabase.txt" for Append as schedule_database
        if (verify == "1")
            display ("\nRecord Saved")

            schedule_database.write("\n"+schedule_data[0]+"t"+schedule_data[1]+"t"+sche
            dule_data[2]+schedule_data[3]+"n"
            CLOSEFILE "scheduledatabase.txt"
            break
        elif (verify == "2")
            display ("\nRecord Saved")

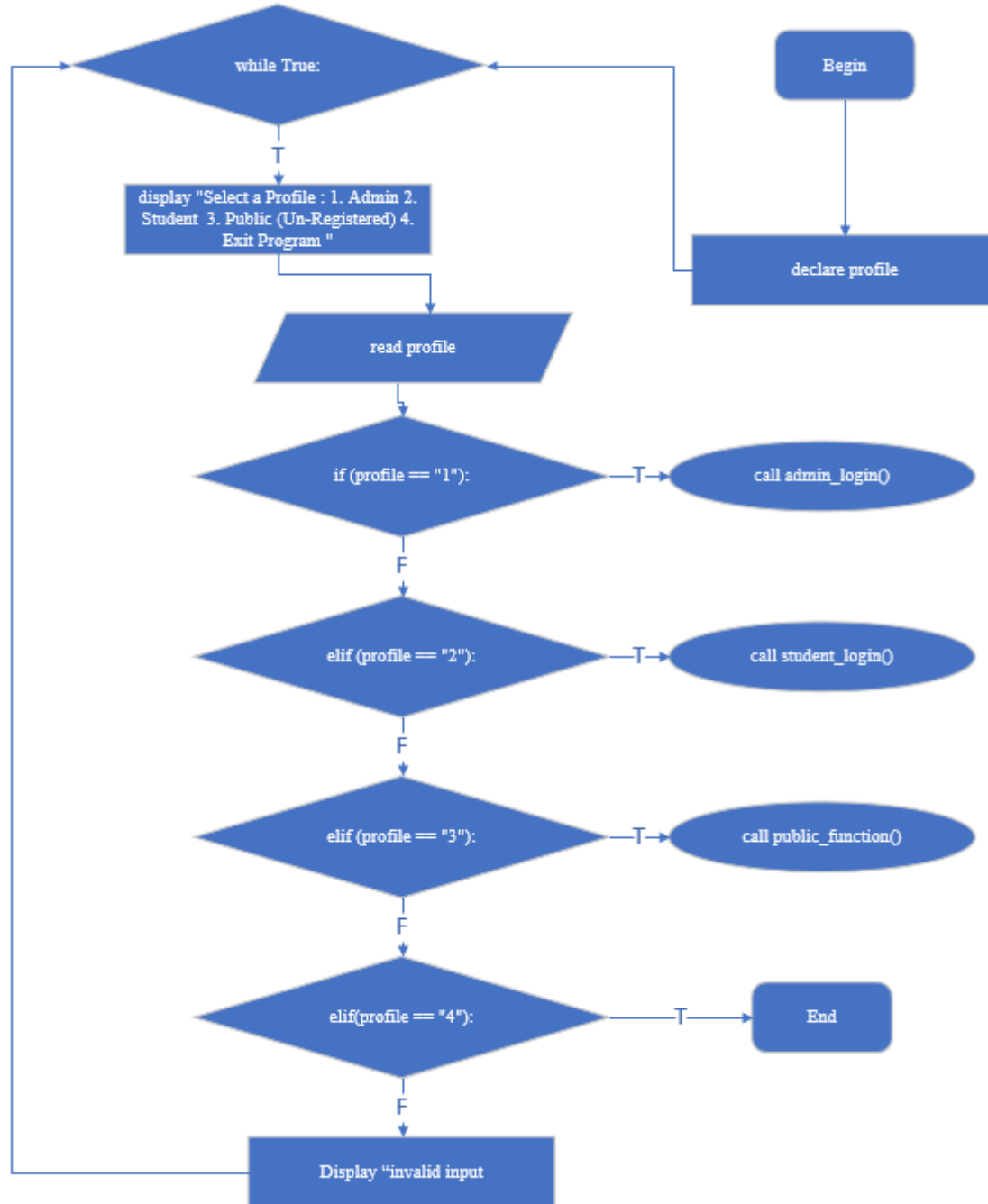
            schedule_database.write("\n"+schedule_data[0]+"t"+schedule_data[1]+"t"+sche
            dule_data[2]+schedule_data[3]+"n"
            CLOSEFILE "scheduledatabase.txt"
            elif (verify == "3")
                display ("\nRecord Saved")

                schedule_database.write("\n"+schedule_data[0]+"t"+schedule_data[1]+"t"+sche
                dule_data[2]+schedule_data[3]+"n"
                CLOSEFILE "scheduledatabase.txt"
                admin_function()
            break
        elif (verify == "b")
            confirm = 0
            call add_records()

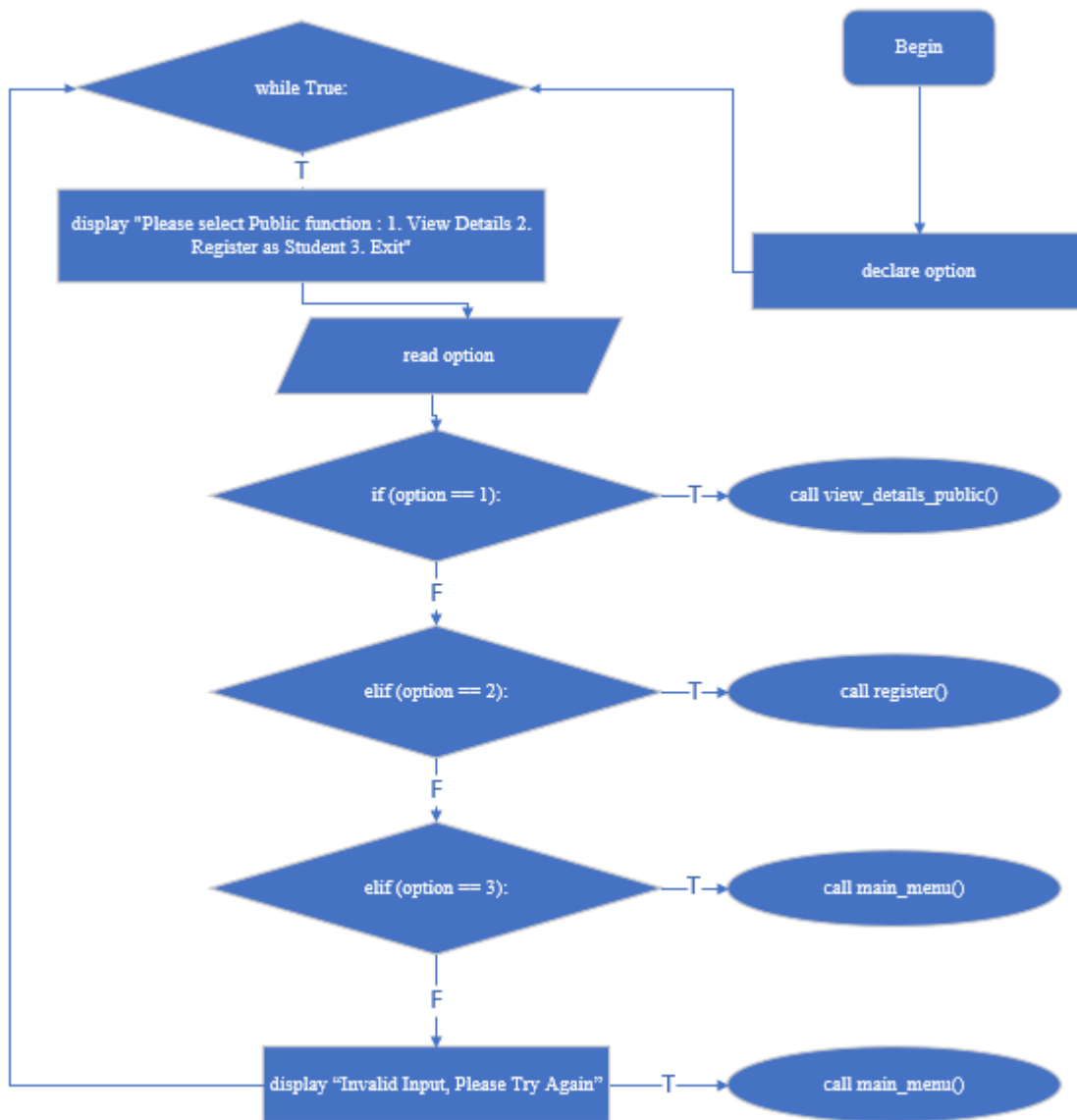
        End if
    End while
End

```

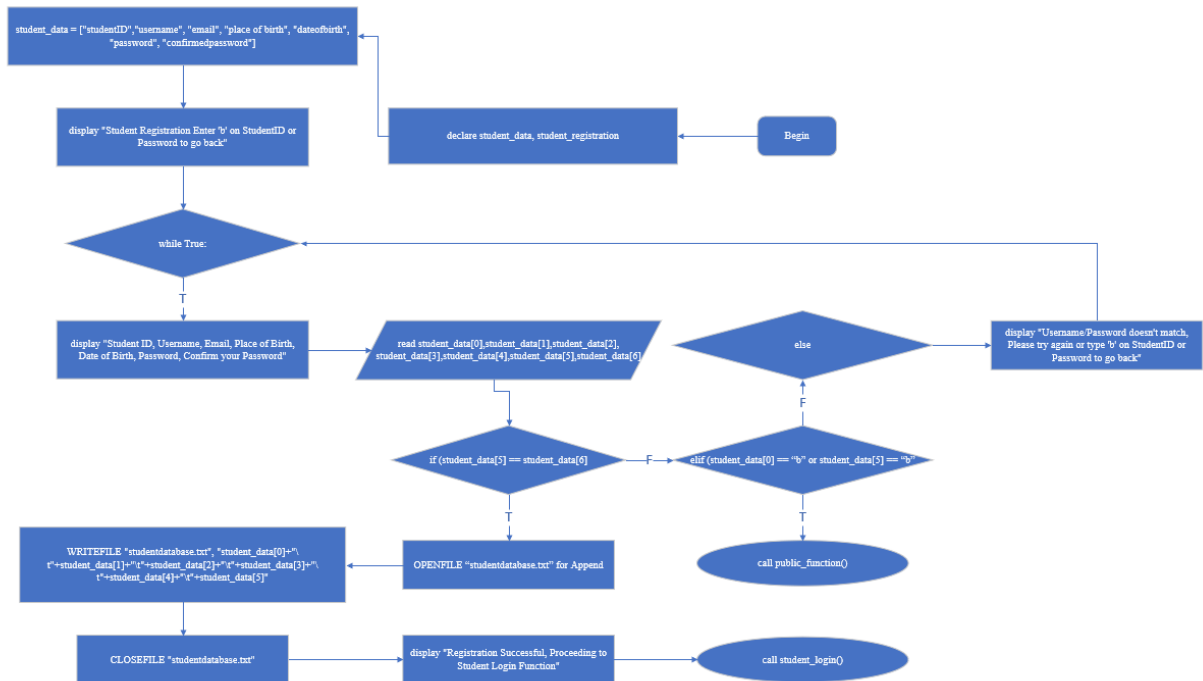
Design – Flowchart



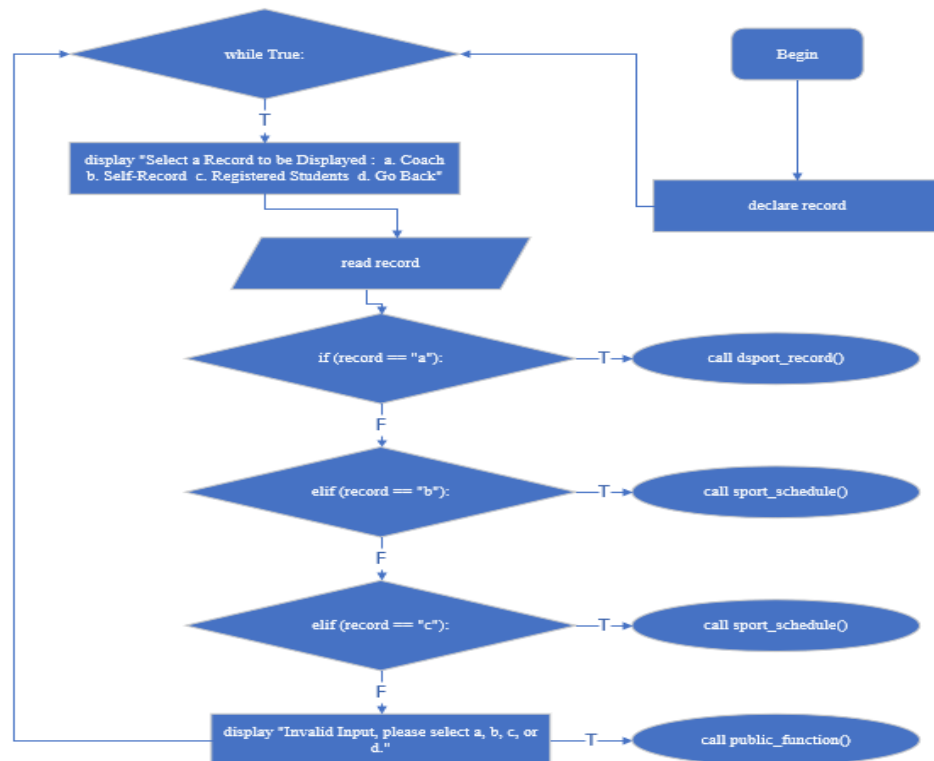
Main Menu Function (def main_menu())



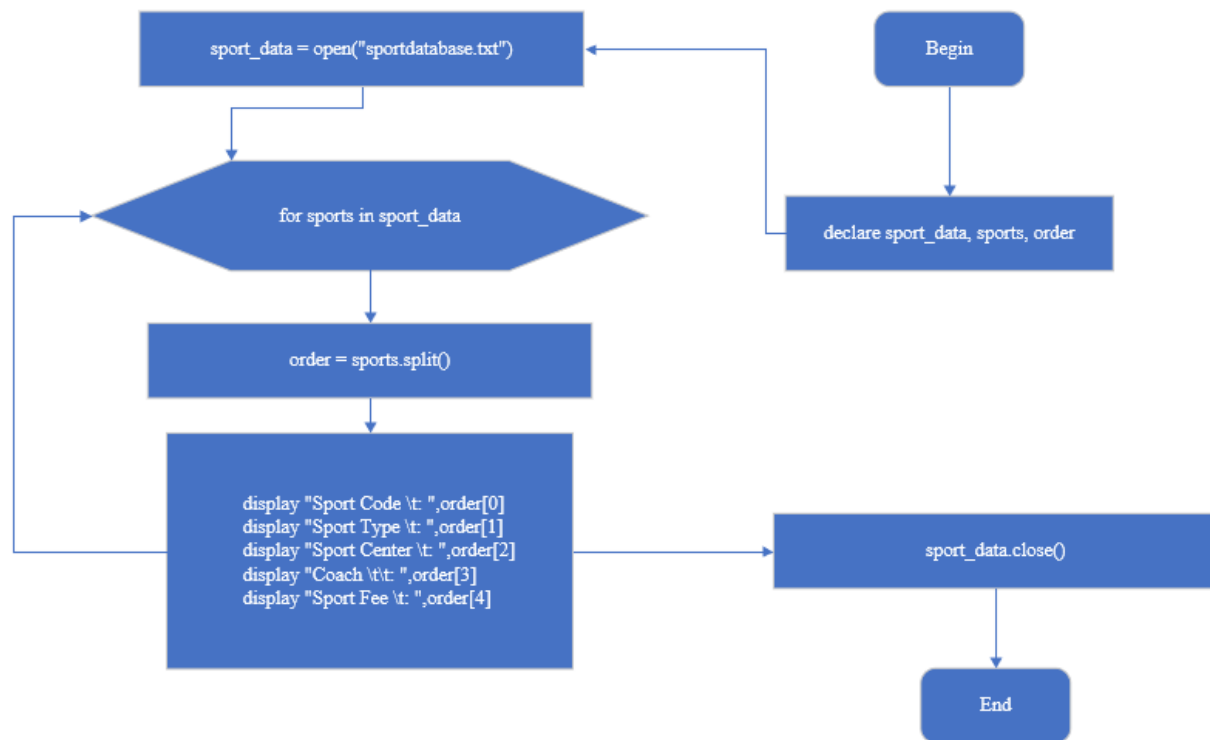
Public Function : General (def public_function())



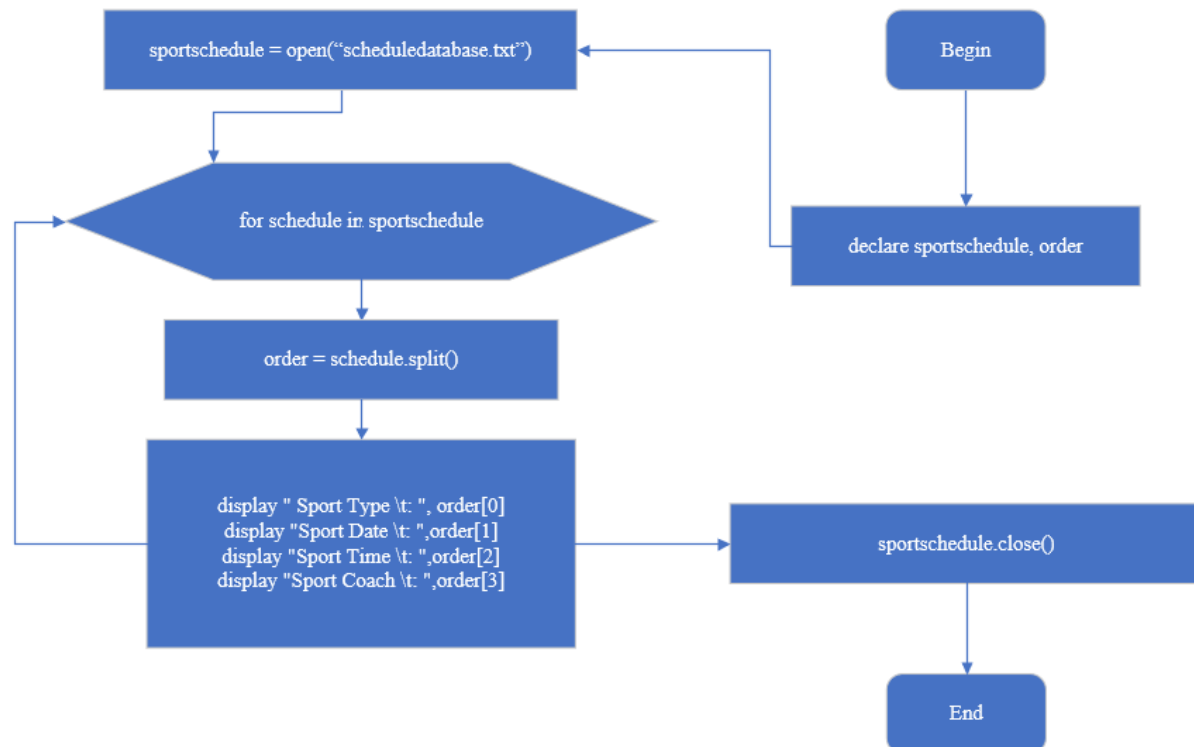
Public Function : Register (def register())



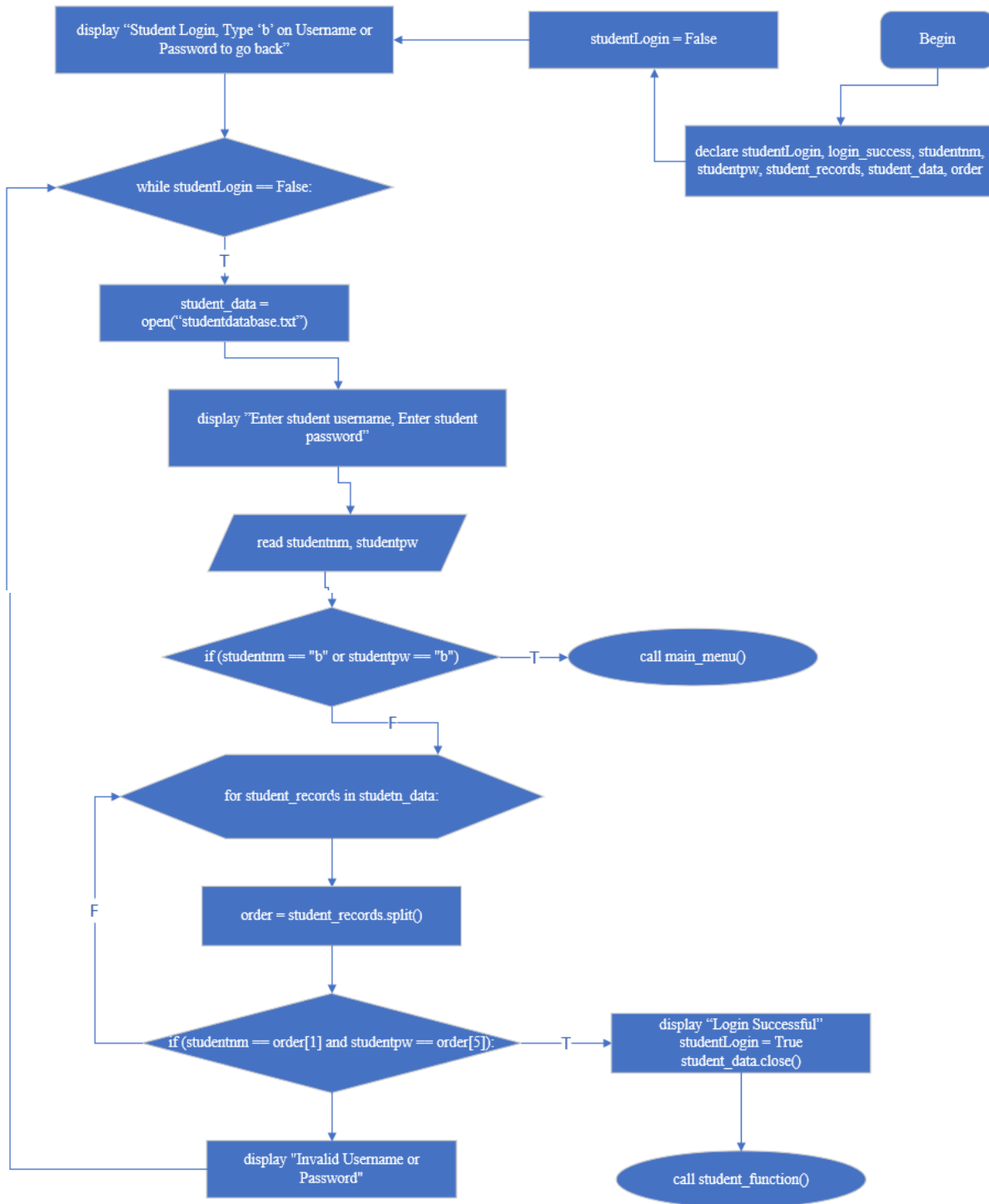
Public Function, View Details (def view_detail_public())



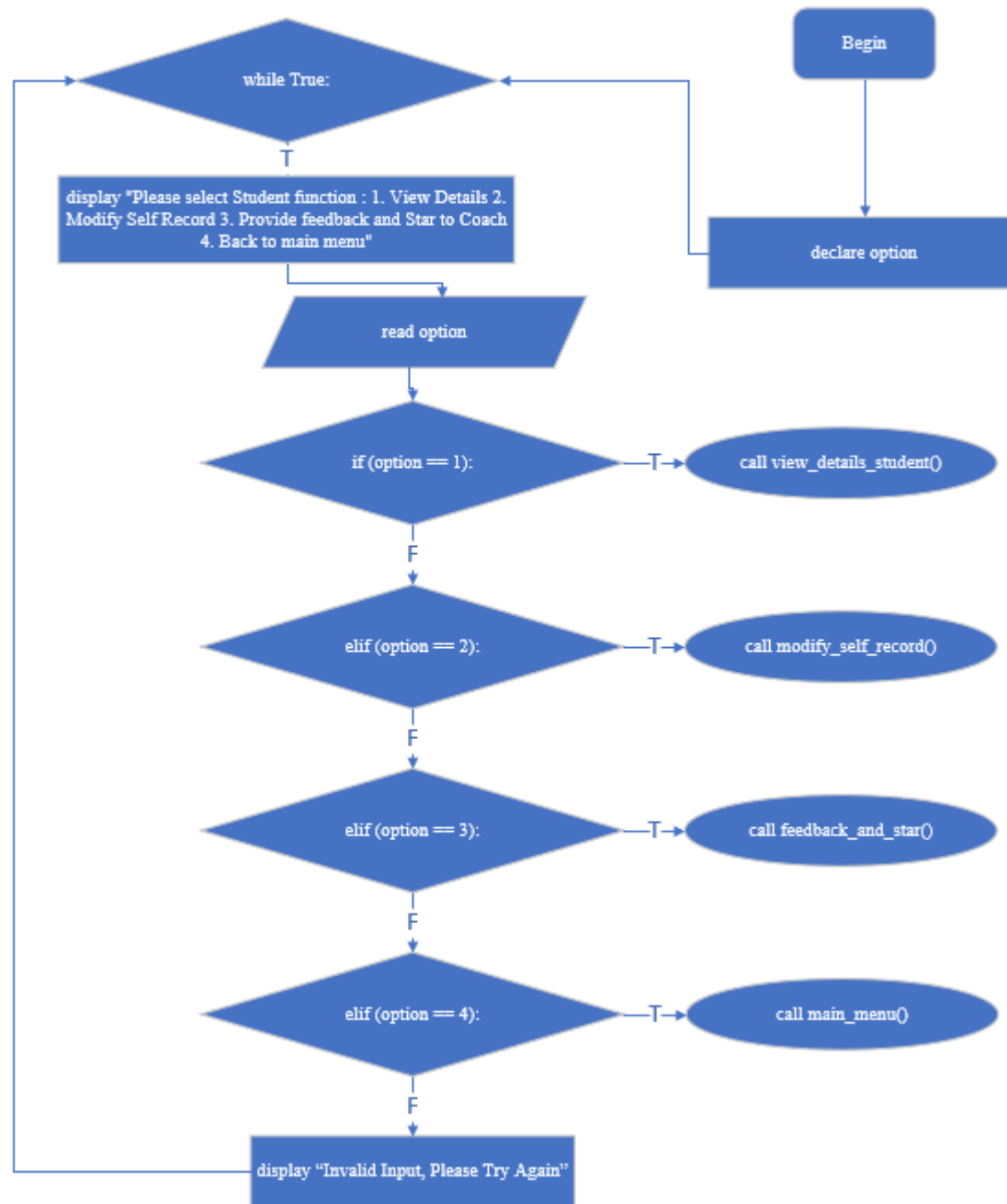
Public Function, View Details of : a. Sport (def dsport_record())



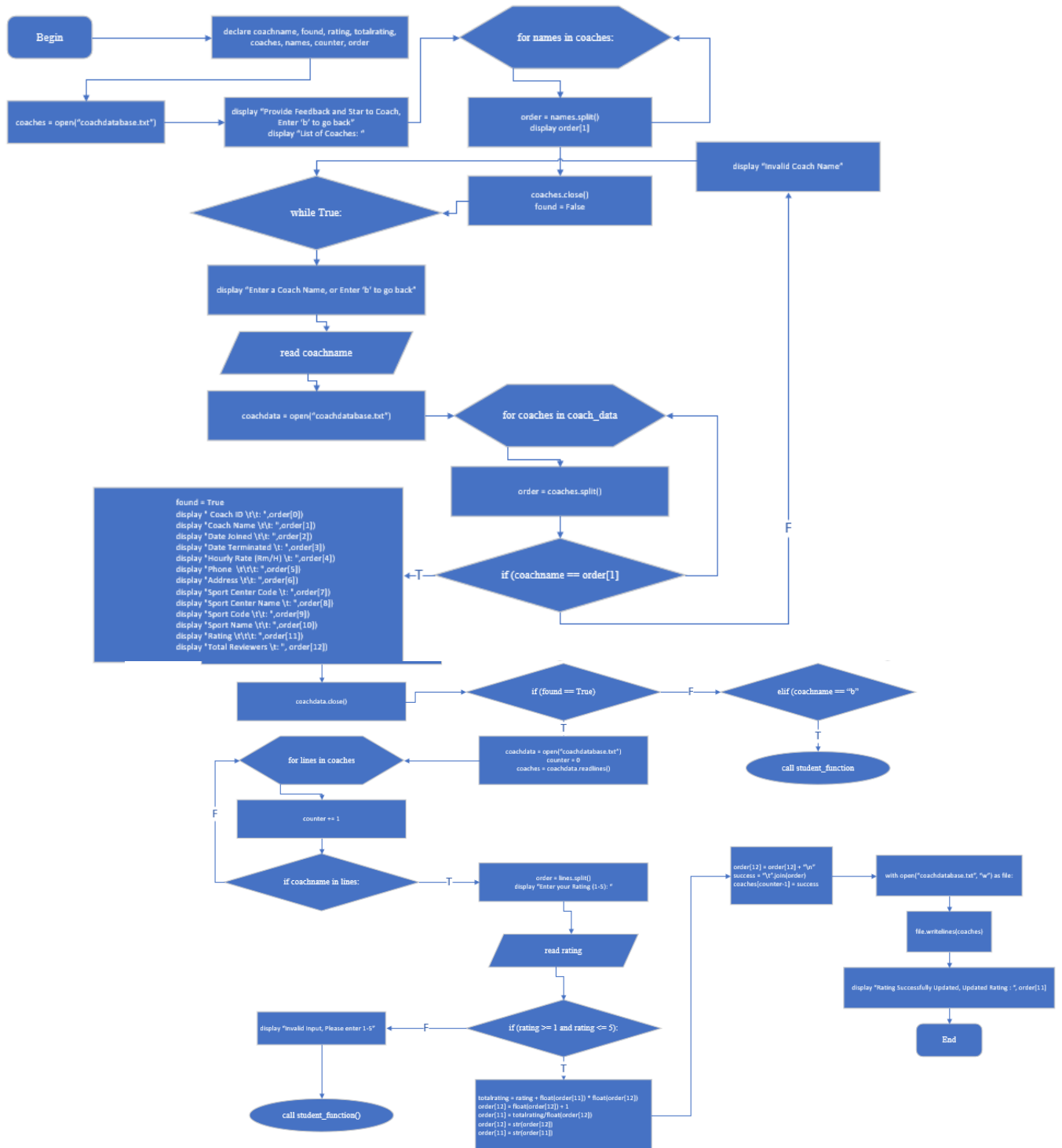
Public Function, View Details of : b. Sport Schedule (def sport_schedule())



Student Login Function (def student_login())

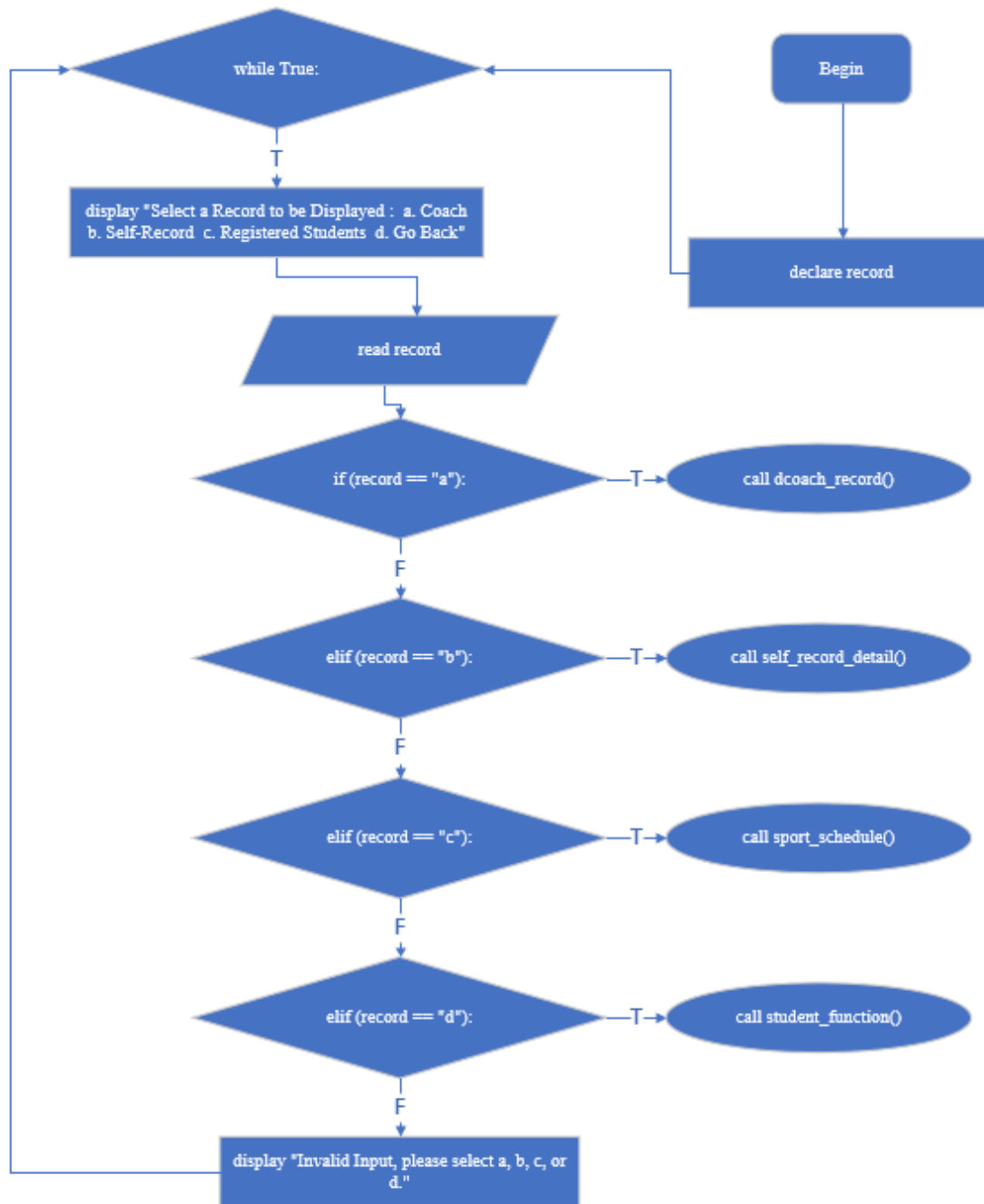


Student Function : General (def student_function())

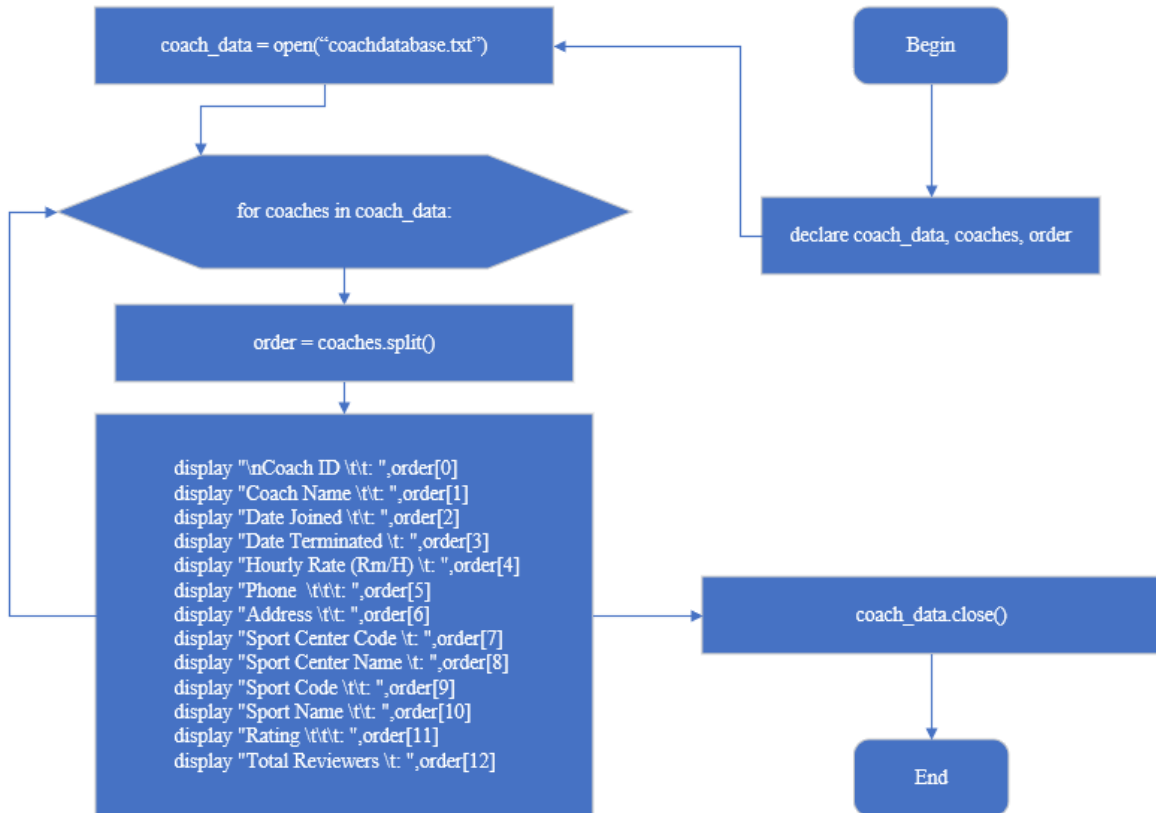


Student Function, Provide Feedback and Star to Coach (def feedback_and_star())

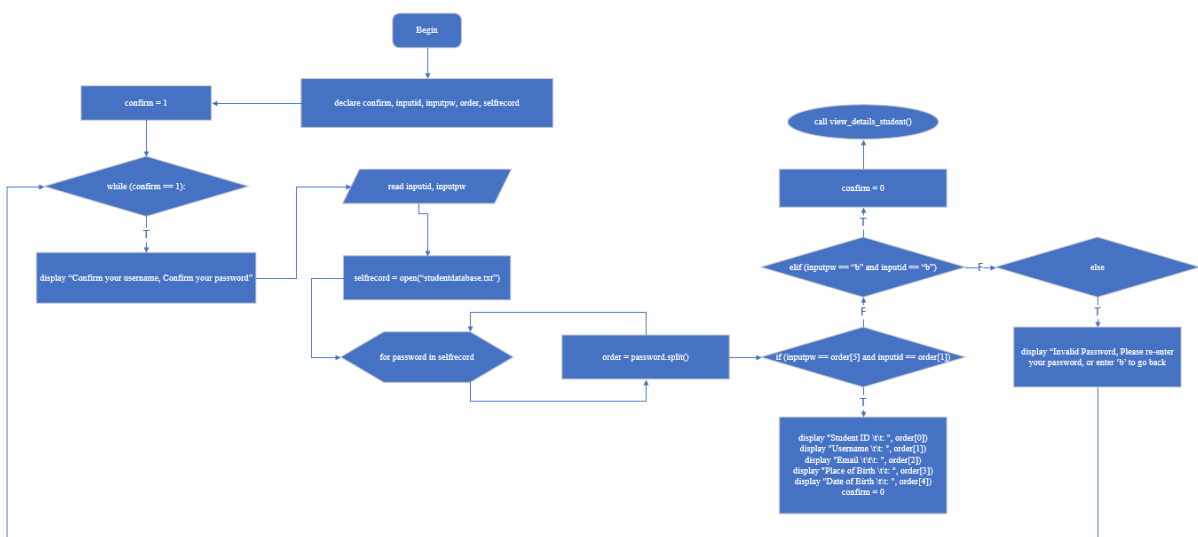




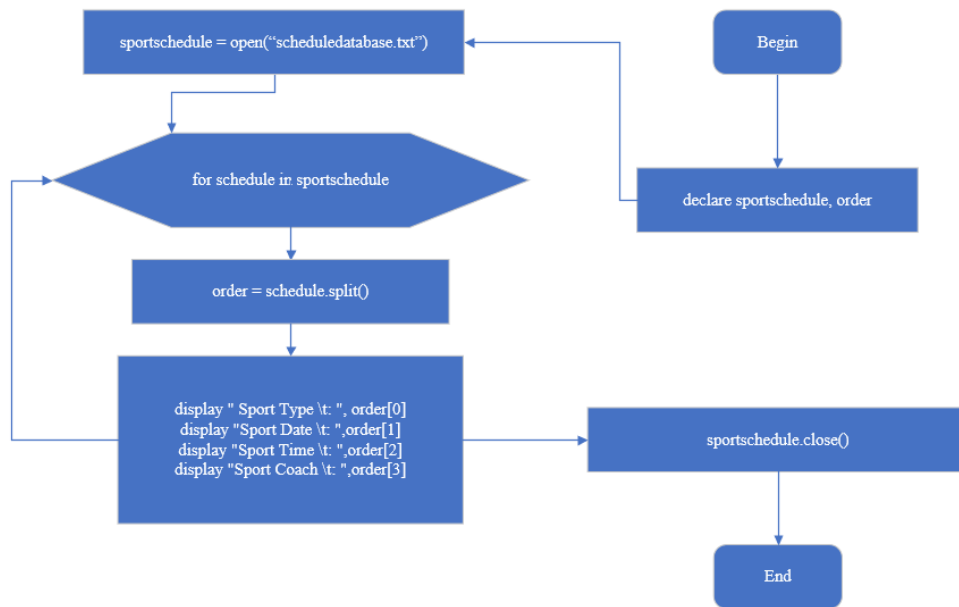
Student Function, View Details (def view_details_student())



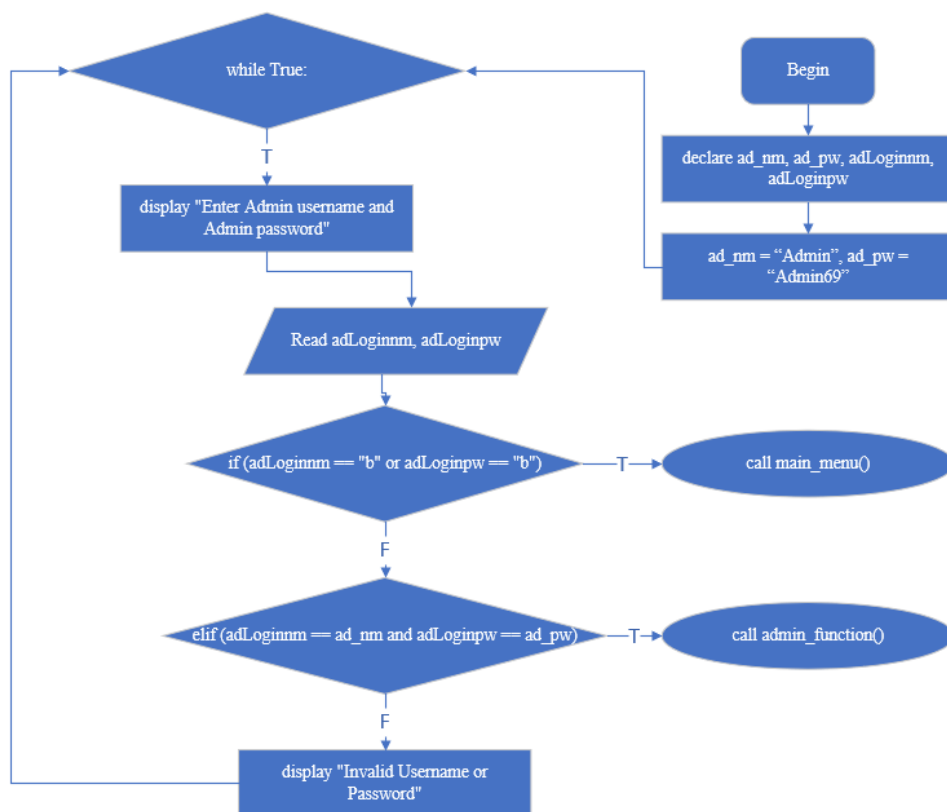
Student Function, View Details of : a. Coach (def dcoach_record())



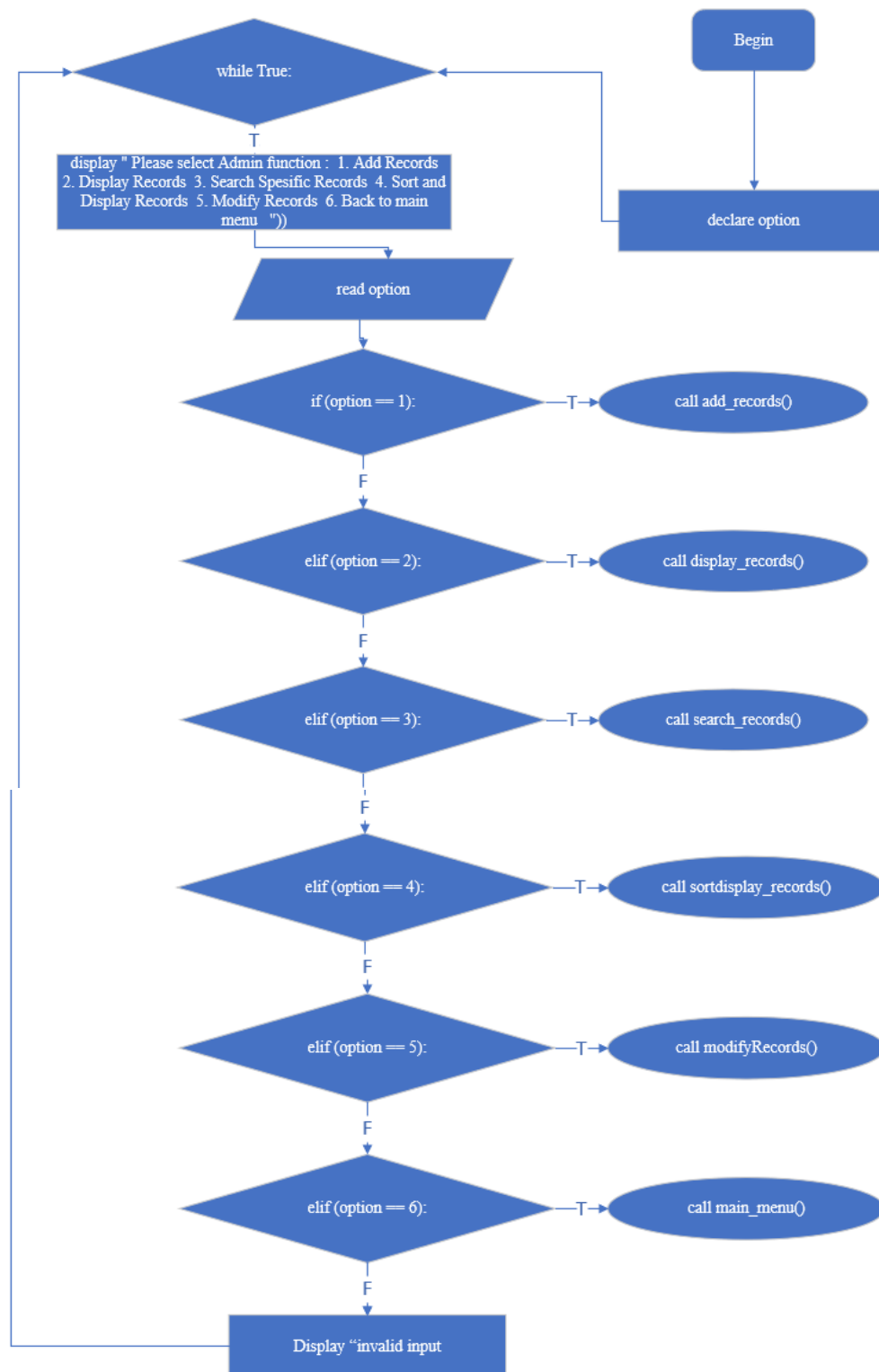
Student Function, View Details of : b. Self-Record (def self_record_detail())



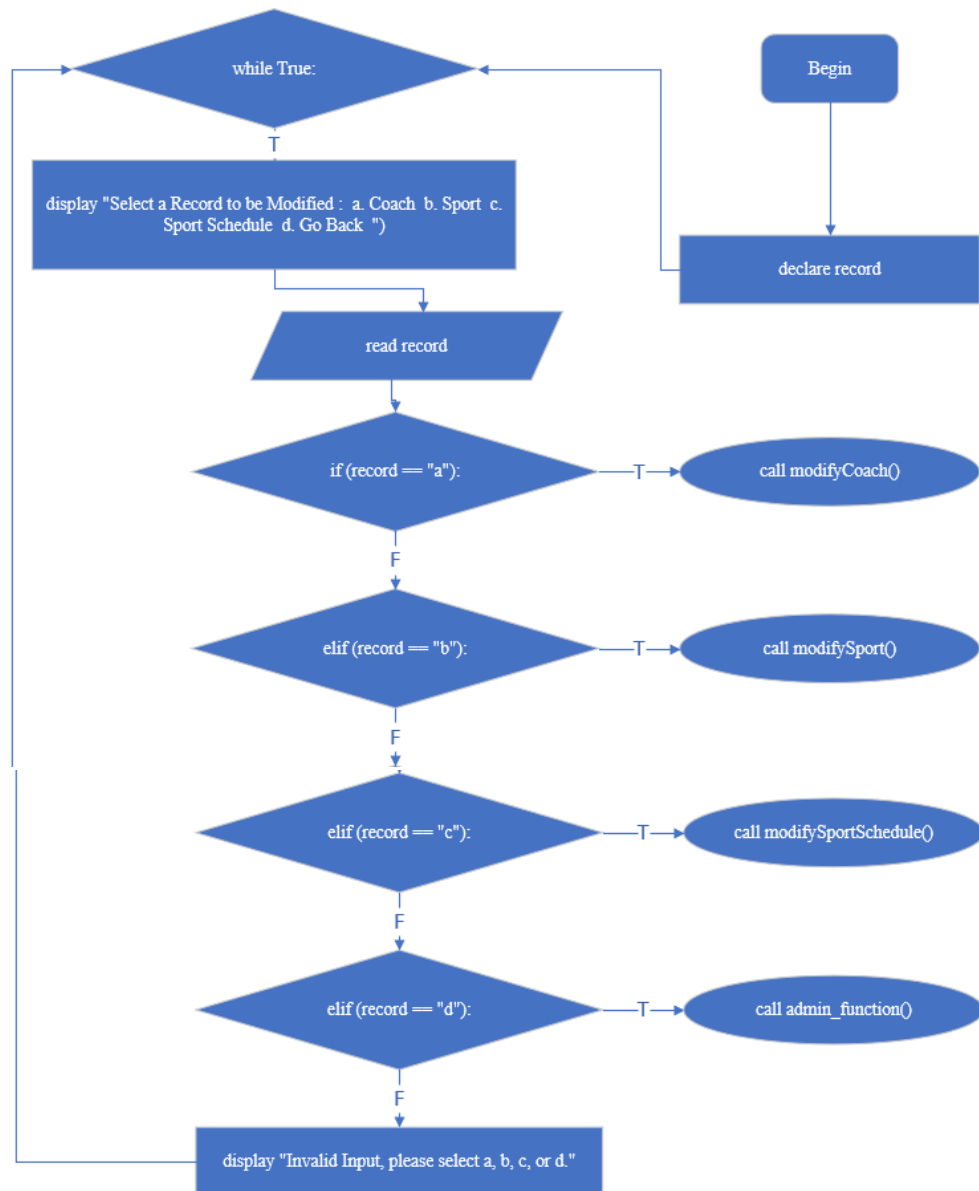
Student Function, View Details of : c. Registered Sport Schedule (def sport_schedule())



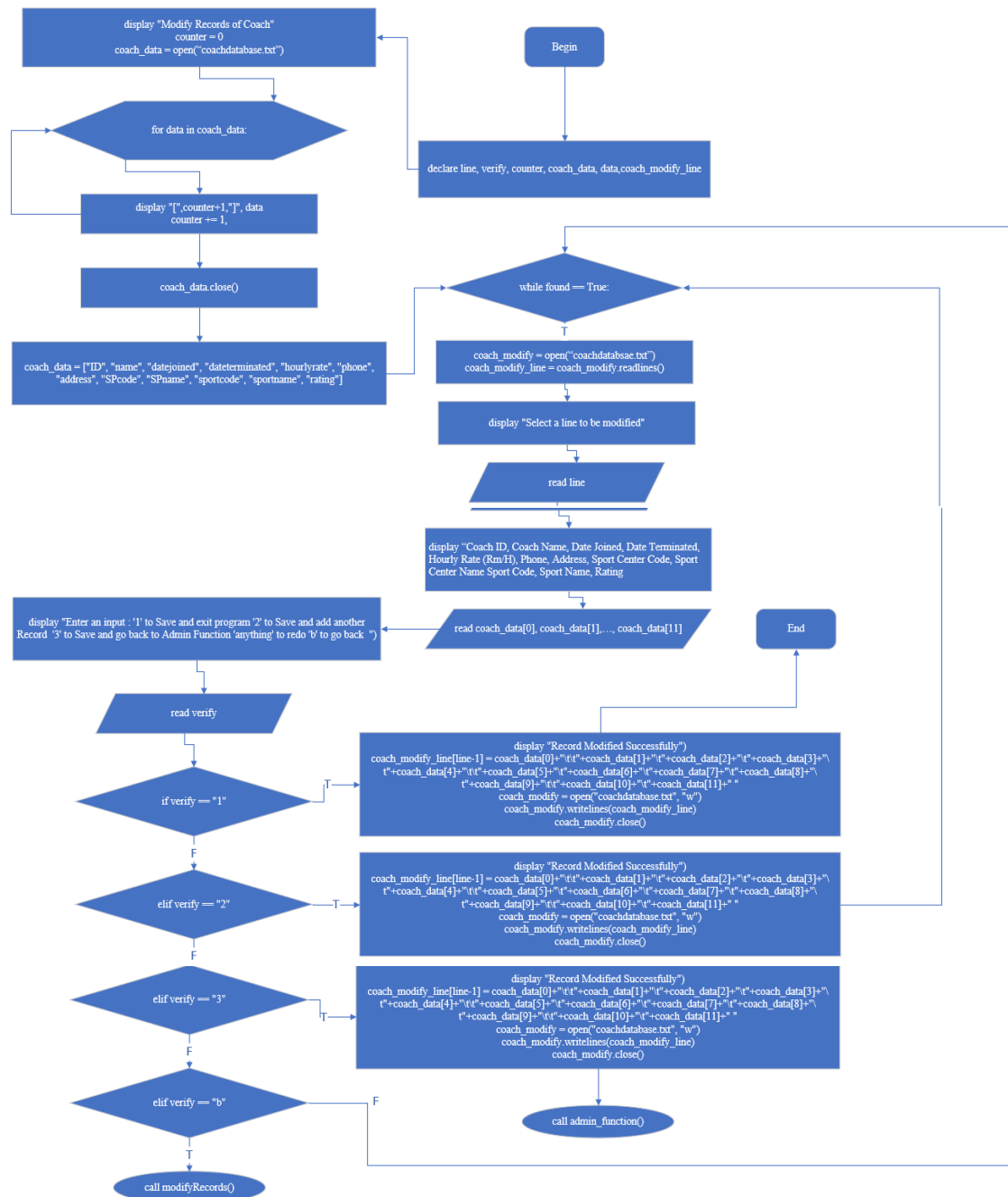
Admin Login Function (def admin_login())



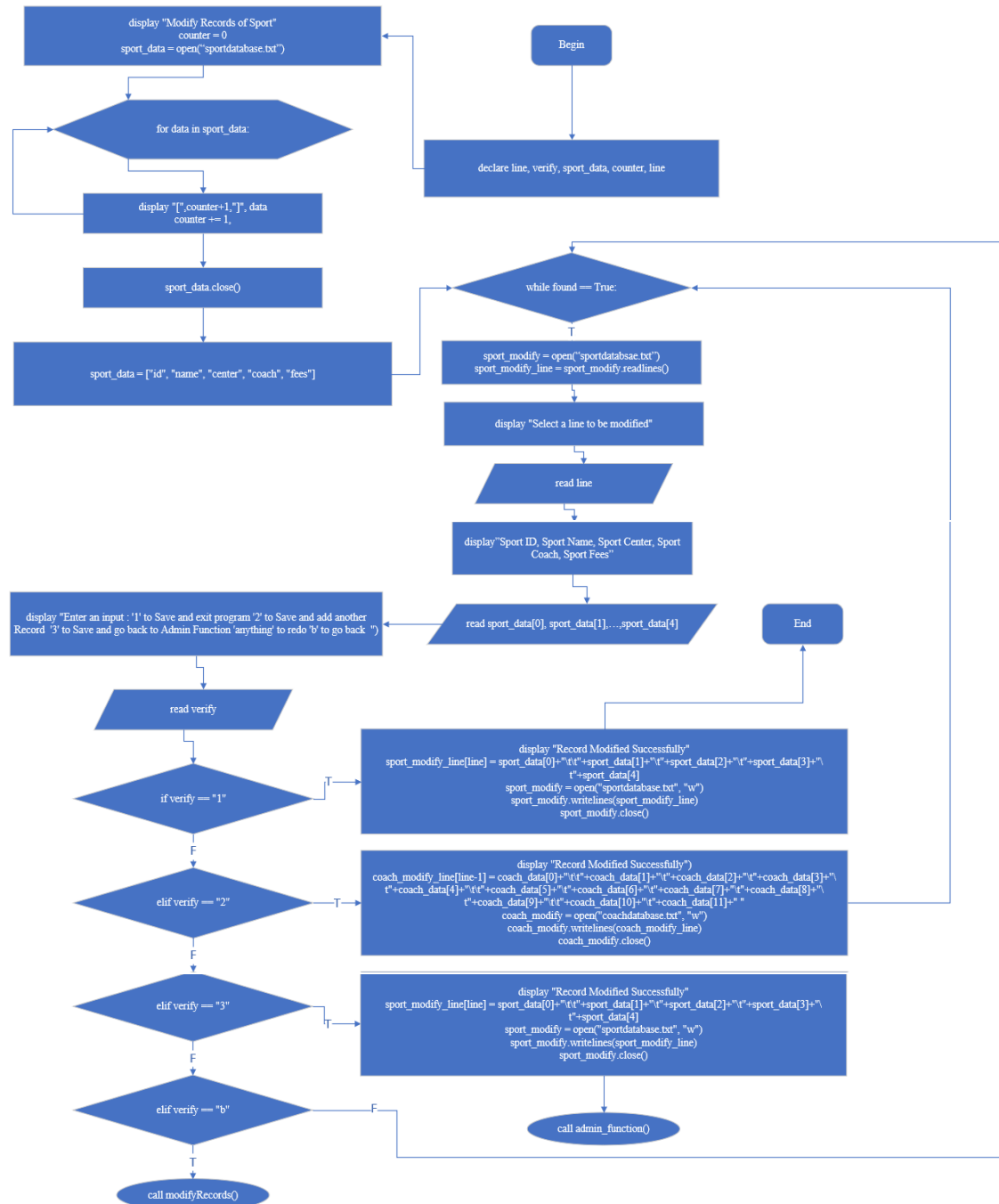
Admin Function : General (def admin_function())



Admin Functoin, Modify Records (def modifyRecords())



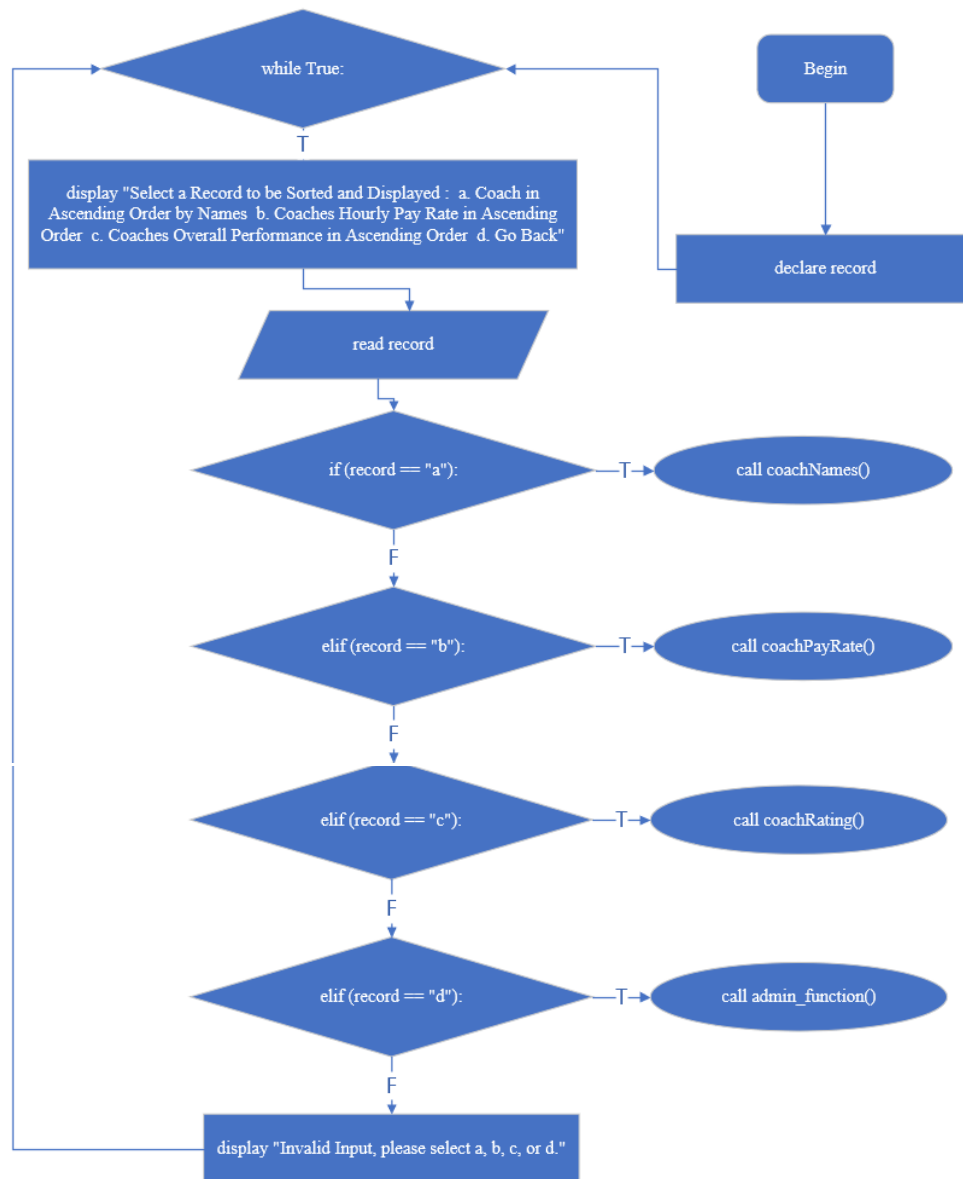
Admin Function, Modify Records of : a. Coach (def modifyCoach())



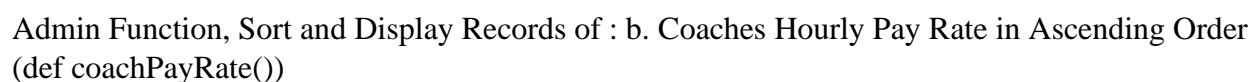
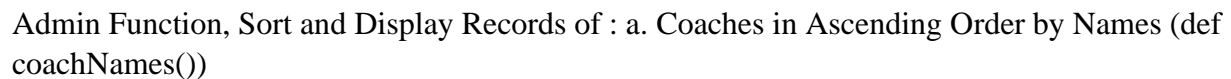
Admin Function, Modify Records of : b. Sport (def modifySport())



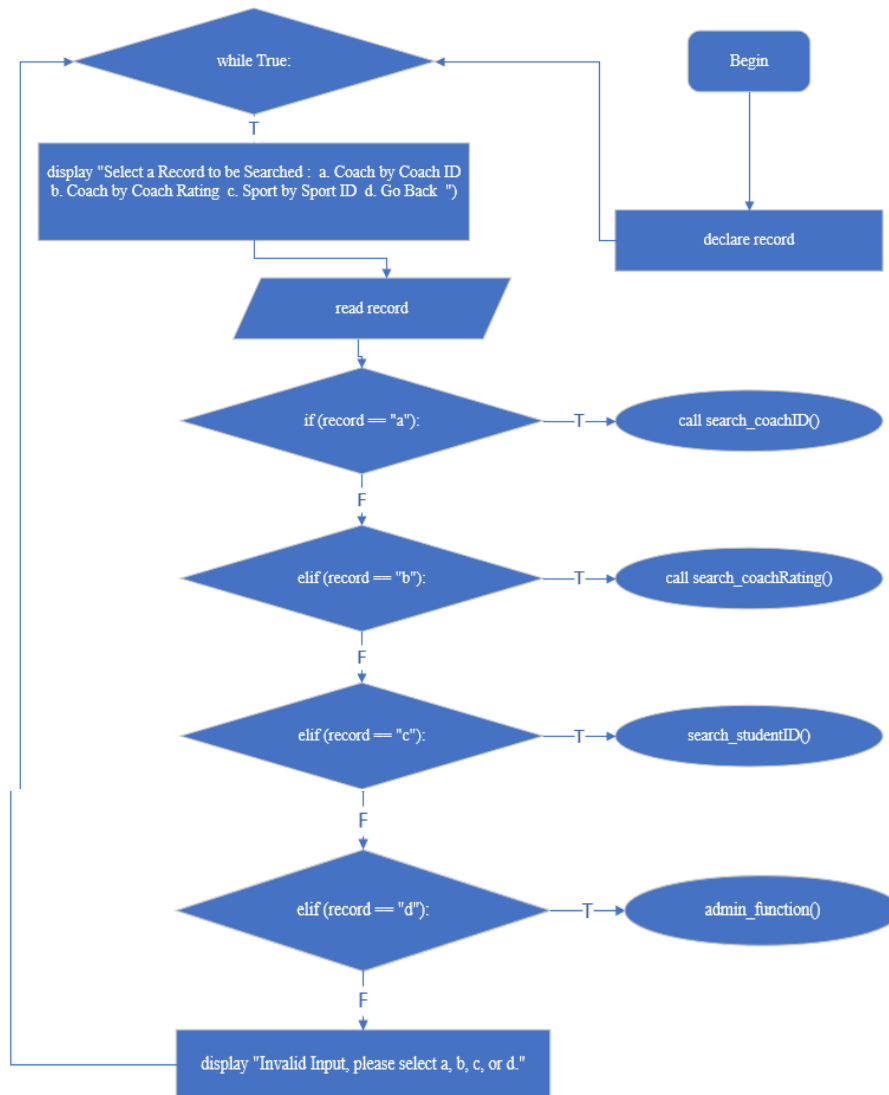
Admin Function, Modify Records of : c. Sport Schedule (def modifySportSchedule())



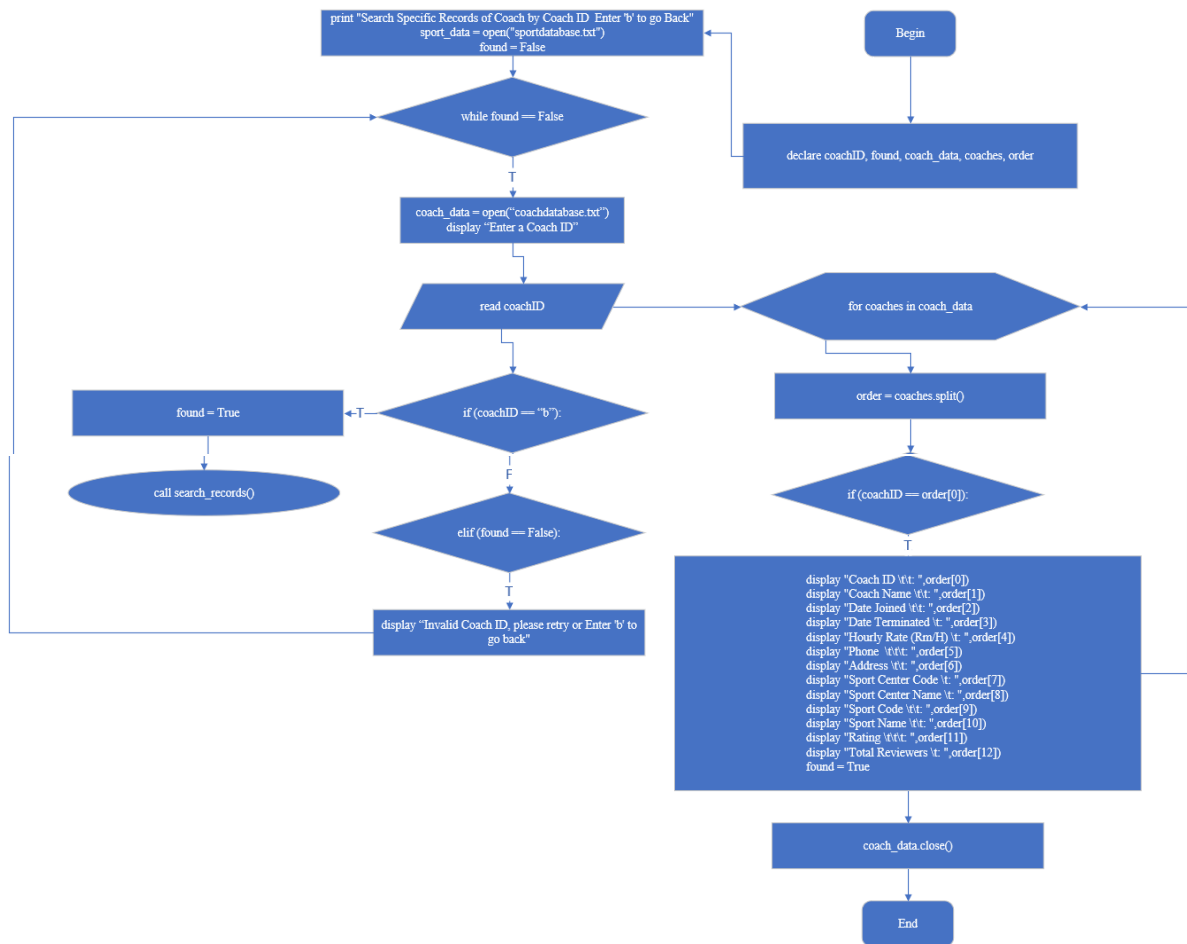
Admin Function, Sort and Display Records (def sortdisplay_records())



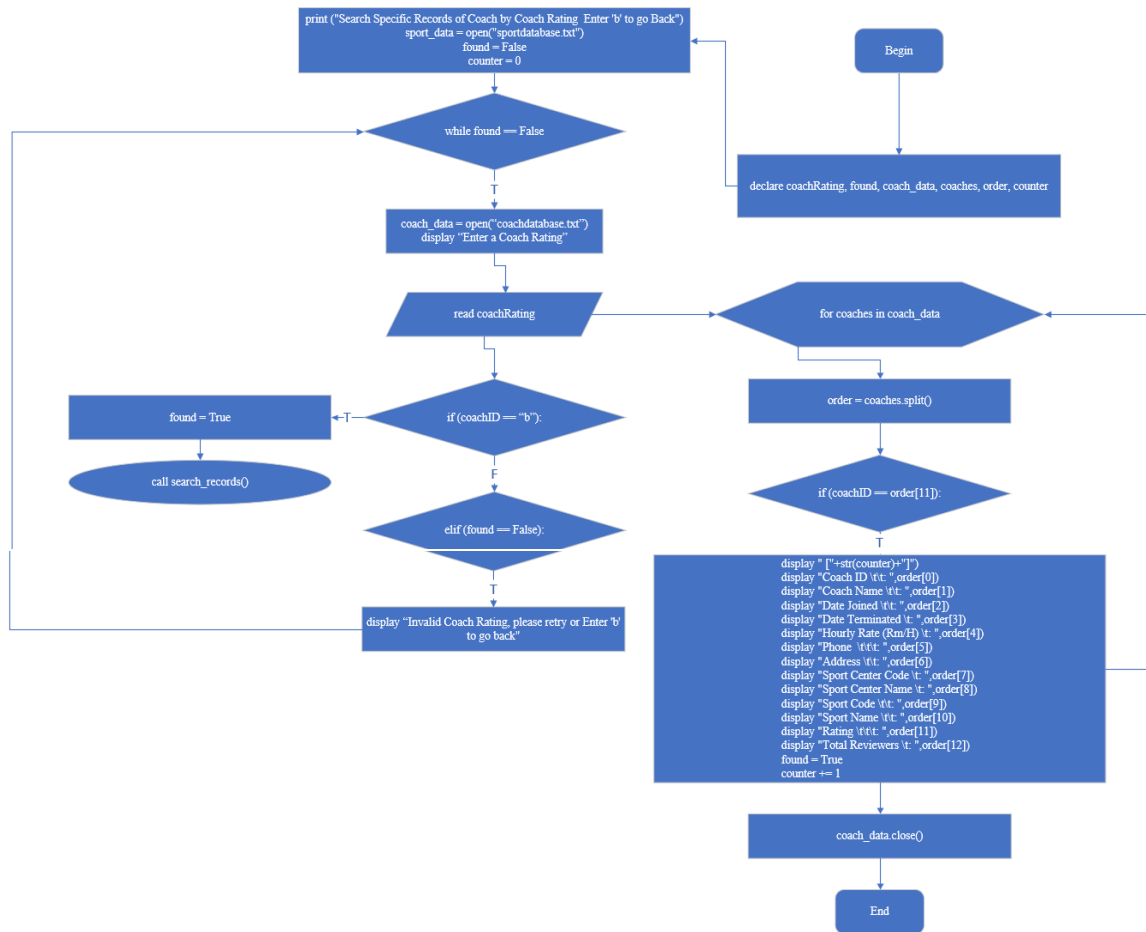




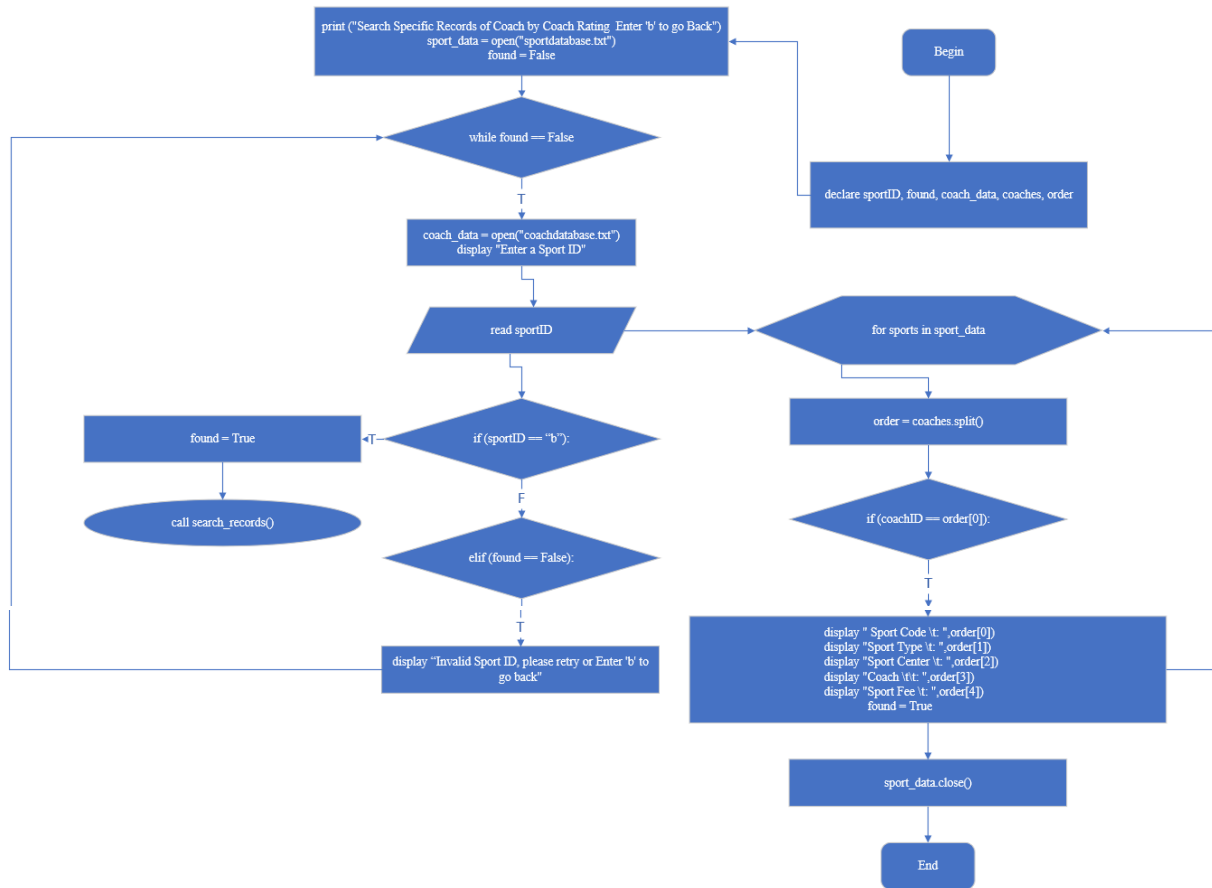
Admin Function, Search Specific Records (def search_records())



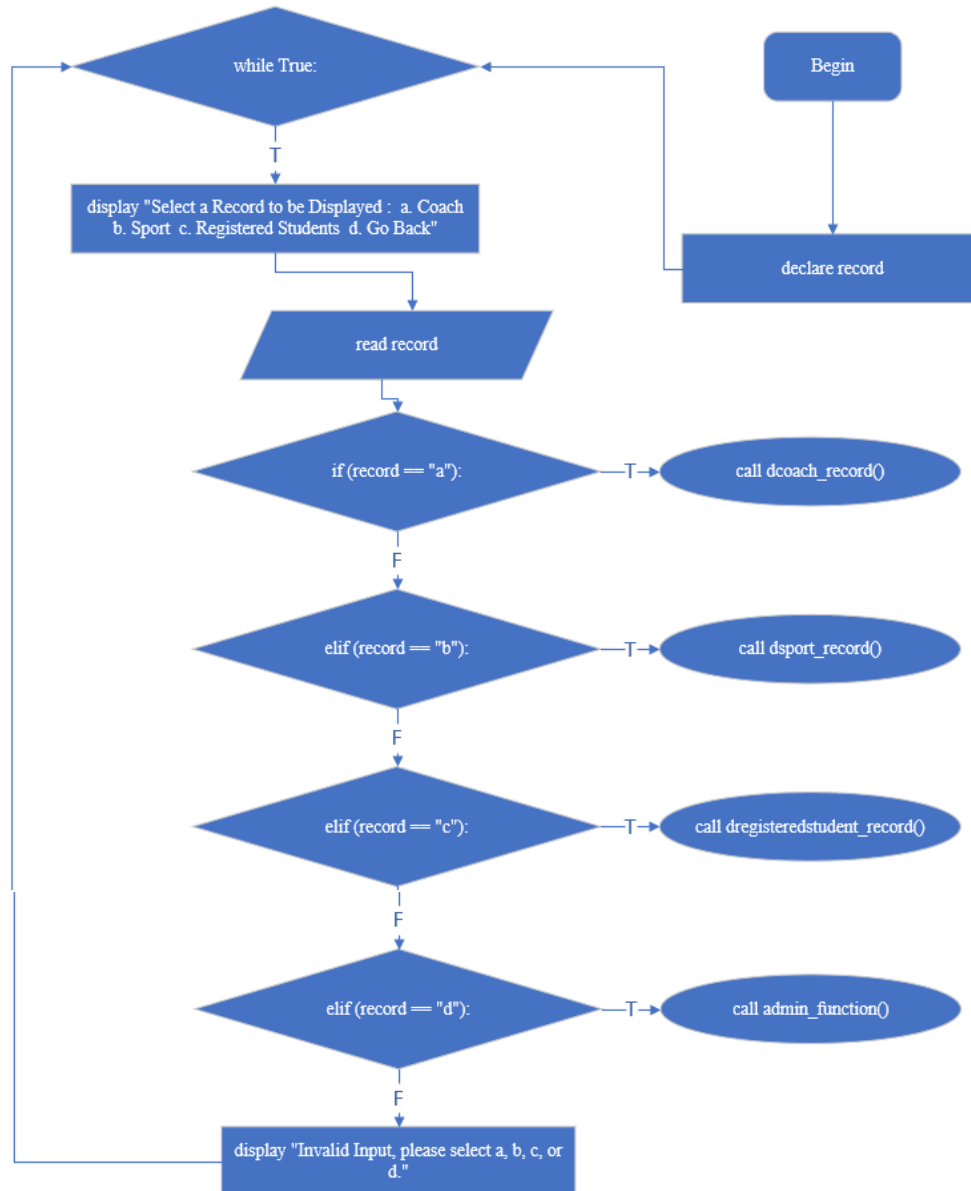
Admin Function, Search Specific Records of : a. Coach by Coach ID (def search_coachID())



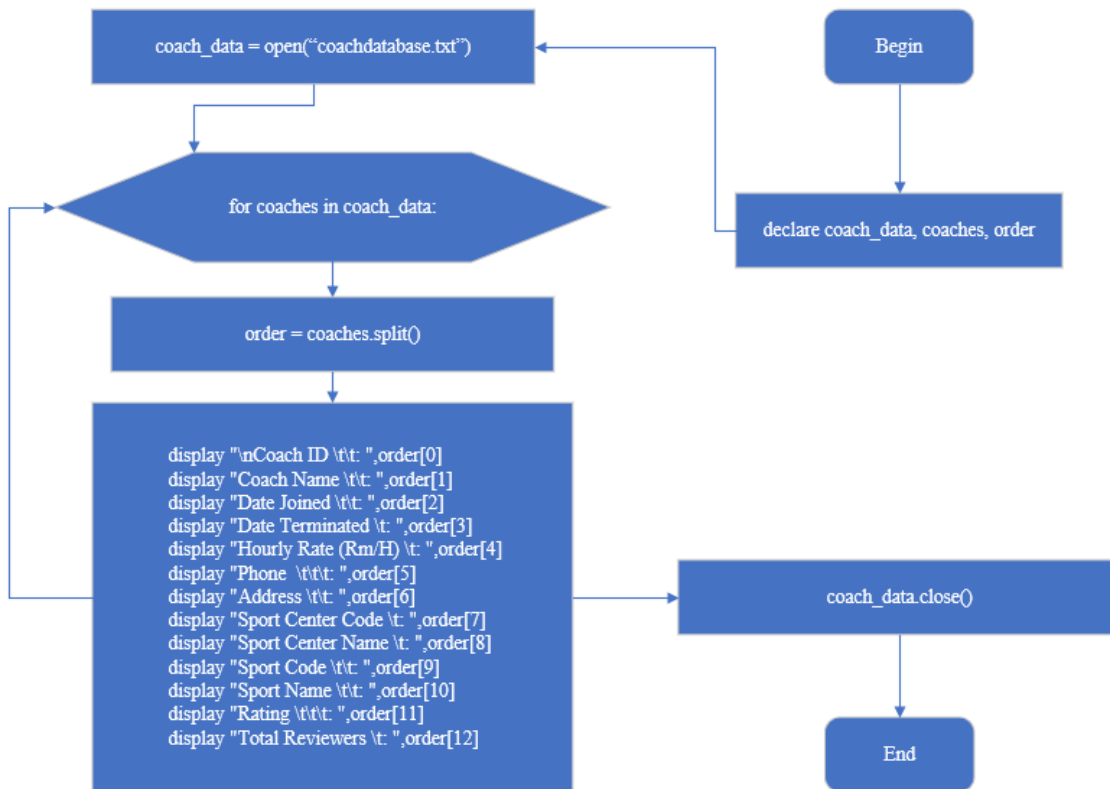
Admin Function, Search Specific Records of : b. Coach by Overall Performance (def search_coachRating())



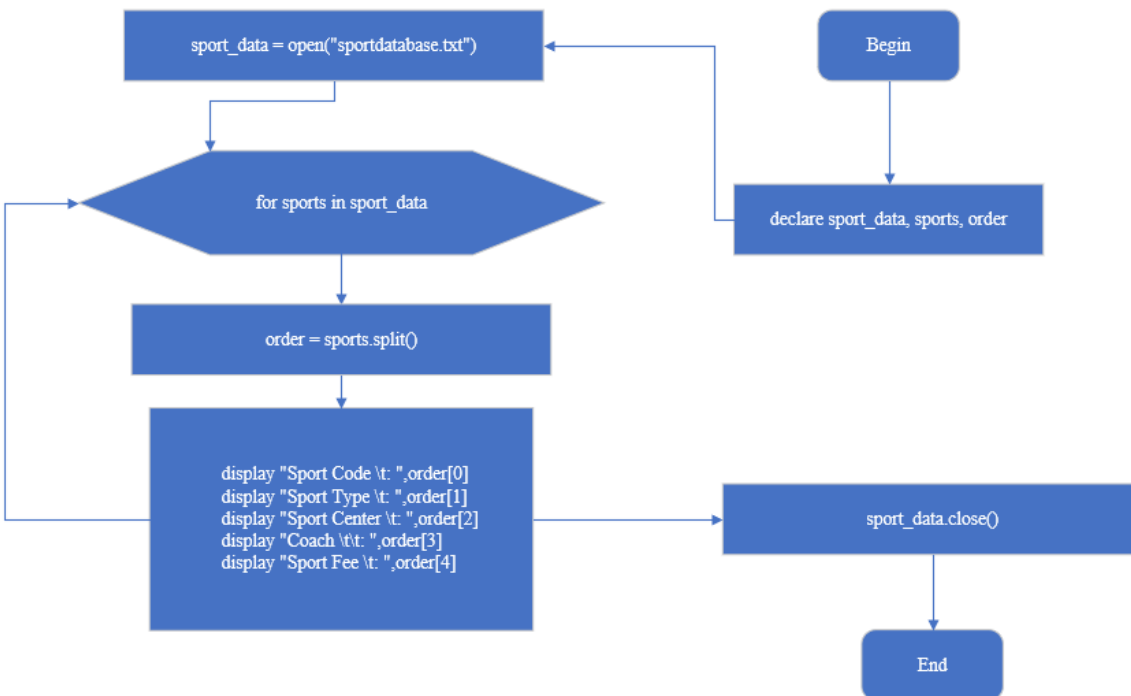
Admin Function, Search Specific Reords of : c. Sport by Sport ID (def search_sportID())



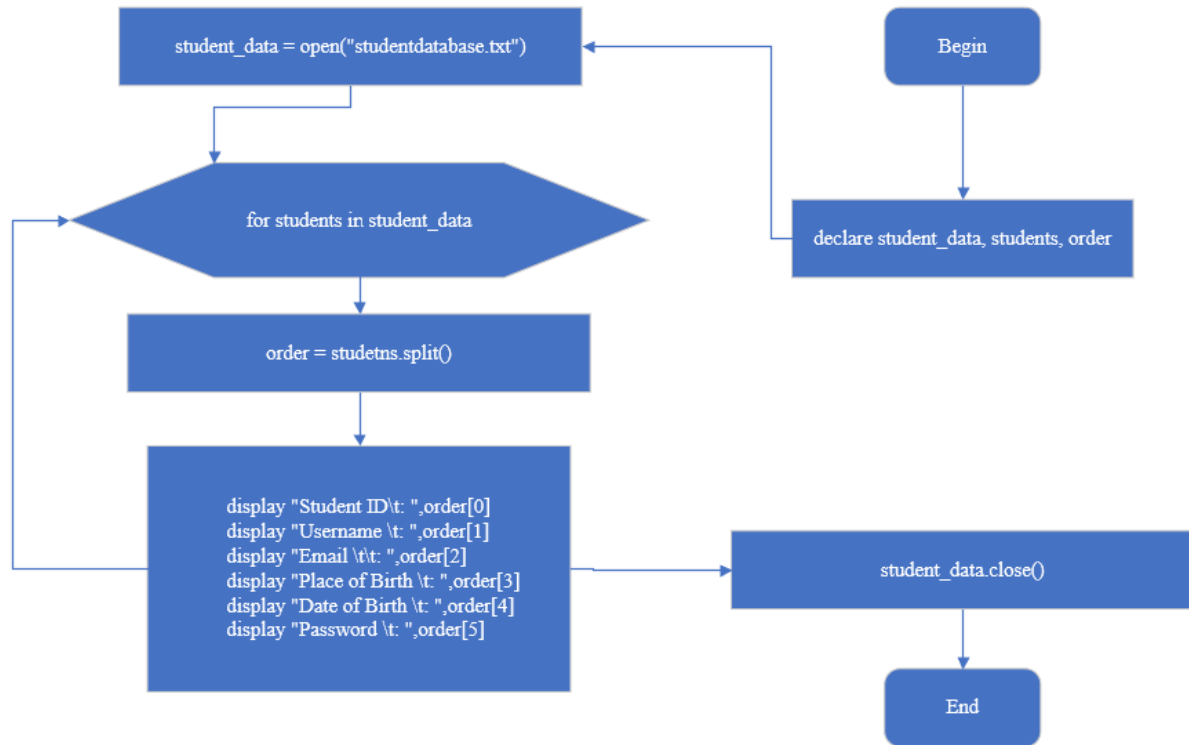
Admin Function, Display Records (def display_records())



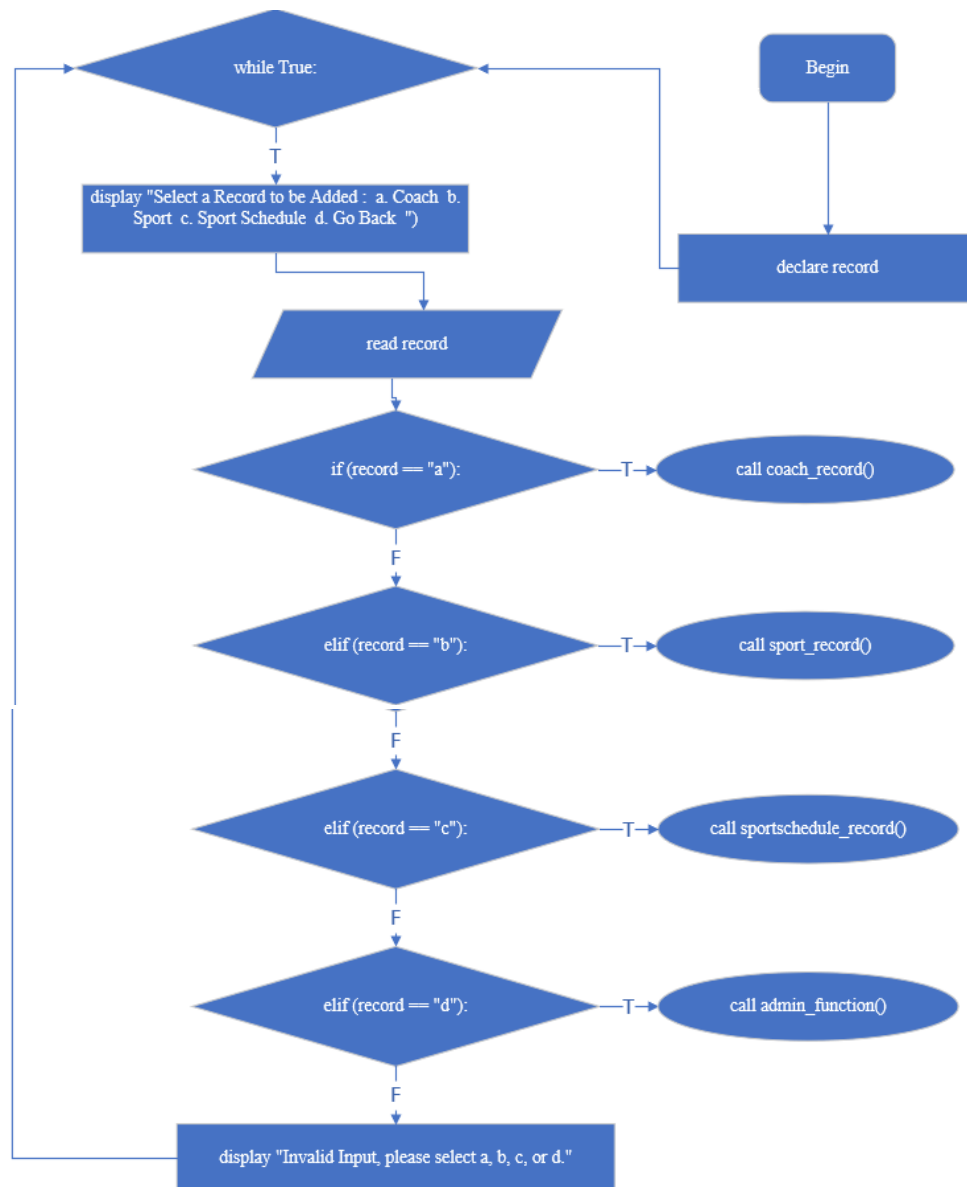
Admin Function, Display Records of : a. Coaches (def dcoach_record())



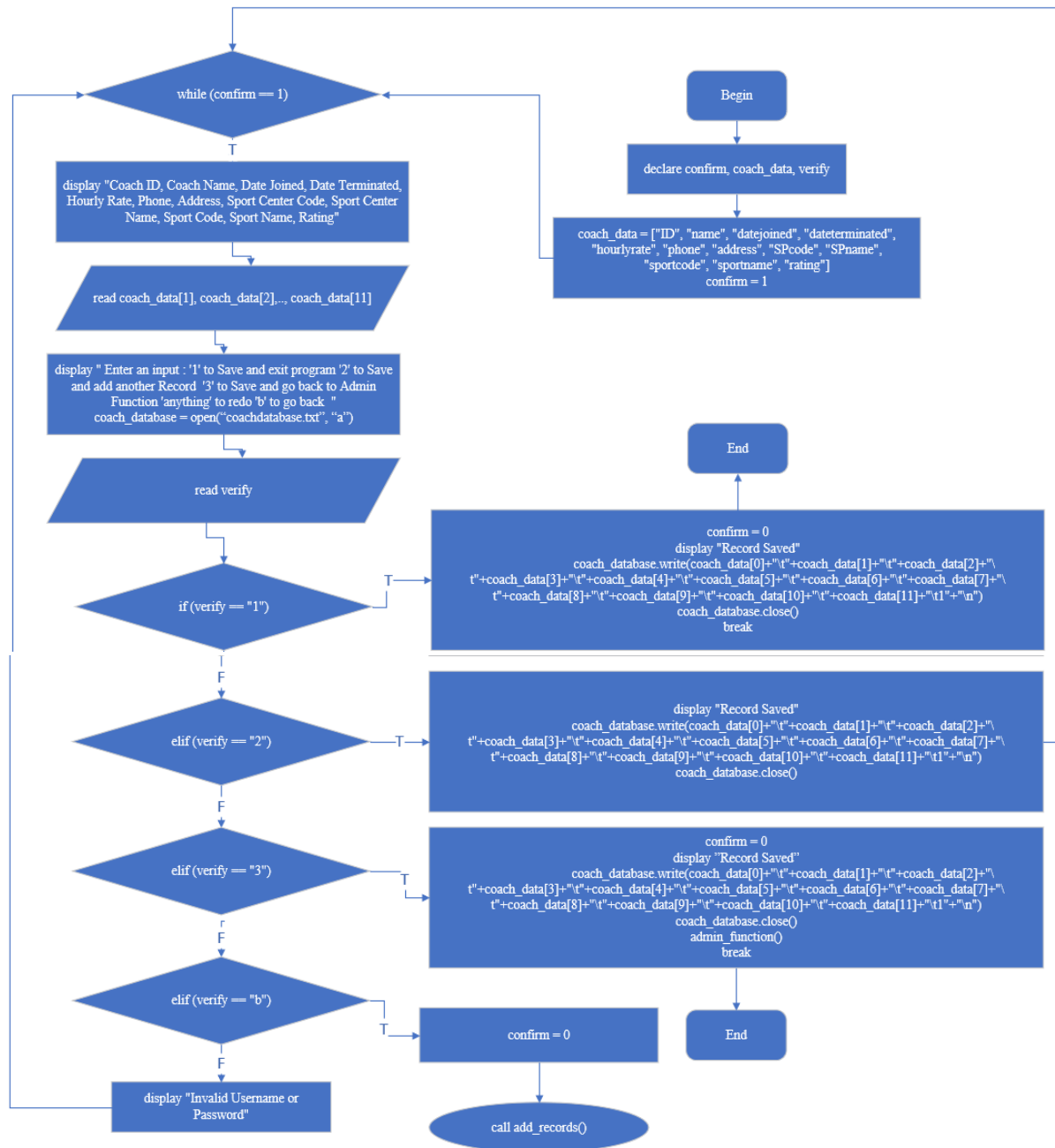
Admin Function, Display Records of : b. Sports (def dsport_record())



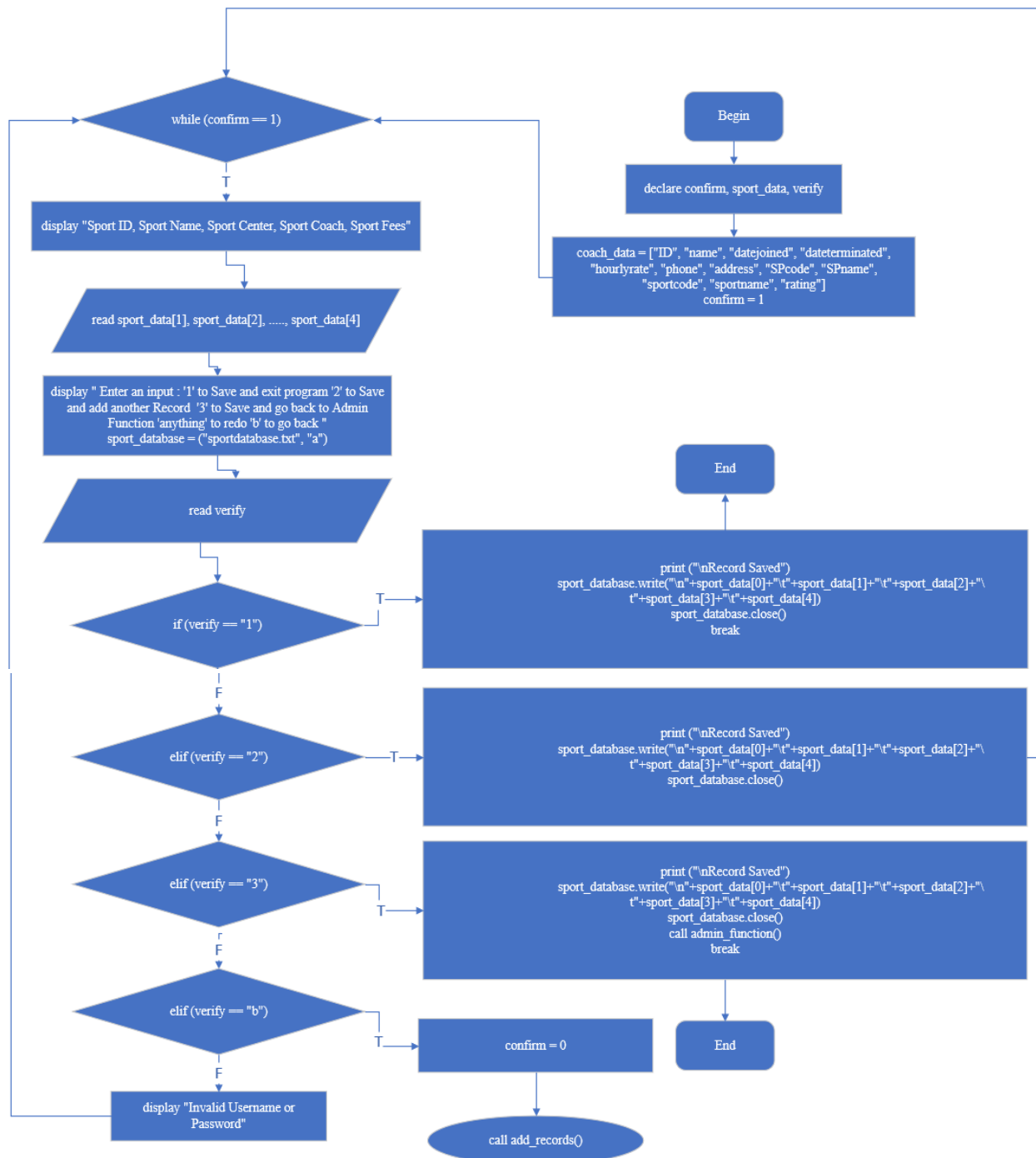
Admin Function, Display Records of : c. Registered Students (def dregisteredstudent_record())



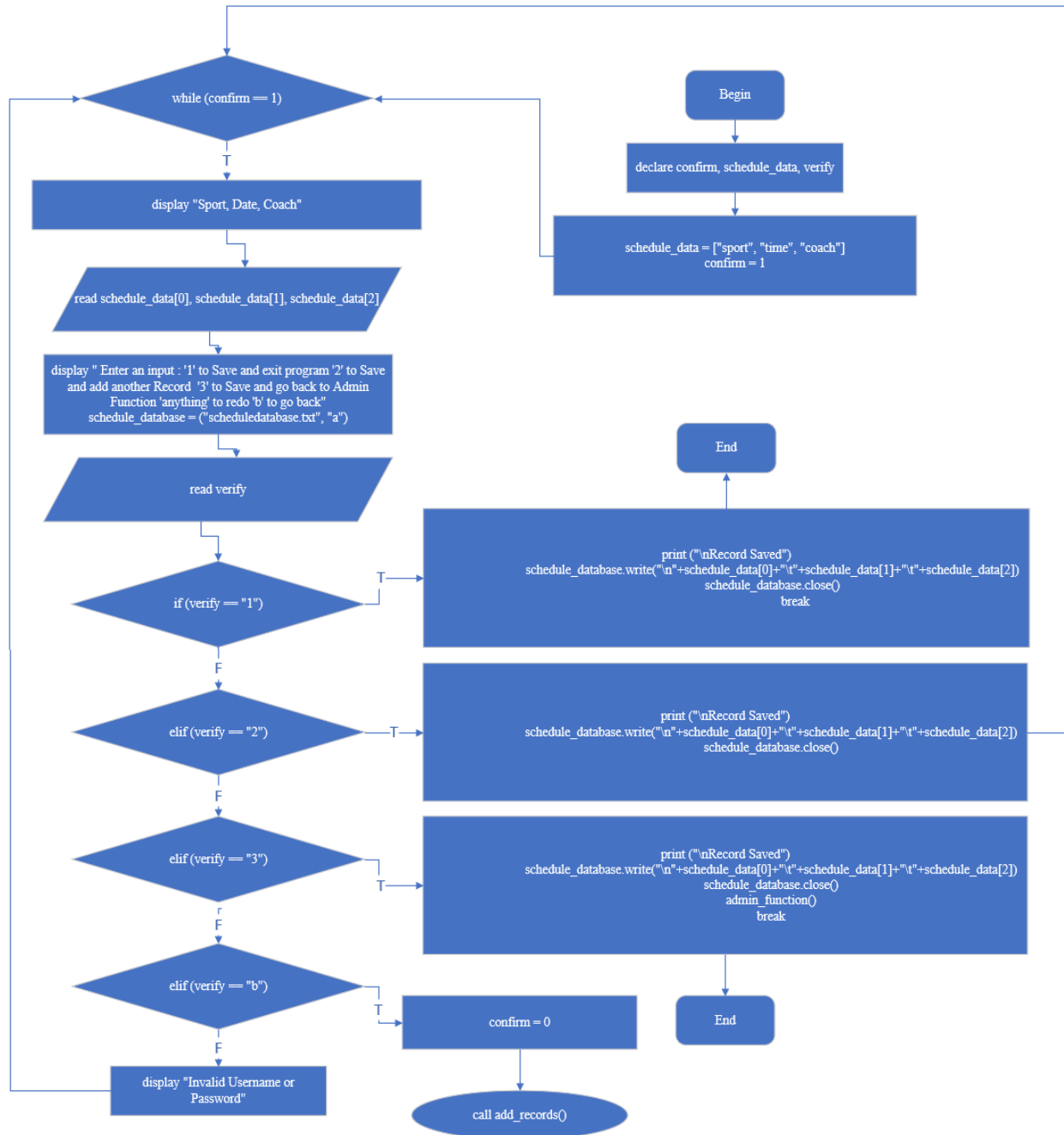
Admin Function, Add Records (def add_records())



Admin Function, Add Records of : a. Coaches (def coach_record())



Admin Function, Add Records of : b. Sports (def sport_record())



Admin Function, Add Records of : c. Sport Schedules (def sportschedule_record())

Sample Python Code/Source Code

a. Variable

```
student_data[0] = input("\nStudent ID \t\t: ")
student_data[1] = input("Username \t\t: ")
student_data[2] = input("Email \t\t\t: ")
student_data[3] = input("Place of Birth \t\t: ")
student_data[4] = input("Date of Birth \t\t: ")
student_data[5] = input("Password \t\t: ")
student_data[6] = input("Confirm your Password \t: ")
```

Variable is something that can store a value, in this example, student_data is a type of list, and all the inputs the user entered will be stored in the student_data list

b. Control Structure

1. If

```
if (studentLogin == False):
    print ("\nInvalid Username or Password\nEnter 'b' to go back")
```

'If' is a conditional statement which means if a condition were to be met, an action will be called, but if the condition weren't met, it will just do nothing. In this scenario, if the studentLogin is False, then the program will display said string

2. If-else

```
if (studentnm == order[1] and studentpw == order[5]):
    login = True
    break
elif(studentnm == "b" and studentpw == "b"):
    student_function()
    break
else:
    counter += 1
```

'If-else' is a type of conditional statement, that if the first 'If' weren't met, it will check for the next condition, and if all the conditions were not met, the 'else' will call the action. In this example, if the first statement were to be met, a Boolean value of True will be assigned to 'login', but if the second condition were to be met, it will call the student_function(), and if all those two conditions weren't met, the counter will be added by 1.

3. Nested If-Else

```

if coachname in lines:
    order = lines.split()
    rating = int(input("Enter Your Rating (1-5)\t: "))
    if (rating >= 1 and rating <= 5):
        totalrating = rating + float(order[11]) * float(order[12])
        order[12] = float(order[12]) + 1
        order[11] = totalrating/float(order[12])
        order[12] = str(order[12])
        order[11] = str(order[11])
    else:
        print("Invalid Input, Please enter 1-5")
        student_function()
        break

```

Nested If-else is a conditional statement, inside a conditional statement. In the example, If the variable 'coachname' is in 'lines', then it will ask another condition, which is if the rating inputted was more than or equal to 1 and less than or equal than 5, and if it is, it will calculate something, if that condition weren't met, it will print out a statement, and it will call student_function()

c. Looping Structure

1. While Loop

```

while True:
    adLoginnm = input("Enter admin username : ")
    adLoginpw = input("Enter admin password : ")
    if (adLoginnm == "b" or adLoginpw == "b"):
        main_menu()
        break
    elif (adLoginnm == ad_nm and adLoginpw == ad_pw):
        print("\nLogin Successful")
        admin_function()
        break
    else:
        print("\nInvalid Username or Password")

```

While Loop is a type loop that will execute as long as the statement were met, in this example, While True will always loop infinitely no matter what, so the only way to break the loop is if the user were to input 'b' on either admin username or password, which will bring the user to main_menu(), or if the user enters the admin username and password correctly, the admin_function() will be called which will lead the user to access admin functions, but if all the conditions weren't met, a statement will be printed and while loop will execute everything again.

2. For Loop

```
sportschedule = open("scheduledatabase.txt")
counter = 1
for schedule in sportschedule:
    order = schedule.split()
    print("\n["+str(counter)+"]")
    print("Sport Type \t: ", order[0])
    print("Date \t\t: ", order[1])
    print("Time \t\t: ", order[2])
    print("Sport Coach \t: ", order[3])
    counter +=1
sportschedule.close()
```

For Loop is a loop that executes a certain times depending on what its looping from, in this example, sportschedule is a variable that is used to store the value of “scheduledatabase.txt” text file, and thus the for loop loops for how many lines that are inside the text file.

d. List Variable

```
schedule_data = ["sport", "time", "coach"]
```

List Variable is a variable that is able to store more than 1 value, in this example, the list ‘schedule_data’ is storing three strings, which are “sport”, “time”, and “coach”.

e. Functions

```
#Main Menu Function
def main_menu():
    print ("\nWelcome to Sport Academy")
    while True:
        profile = input("Select a Profile : \n1. Admin \n2. Student \n3. Public (Un-Registered) \n4. Exit Program \n\n")
        if (profile == "1"):
            admin_login()
            break
        elif (profile == "2"):
            student_login()
            break
        elif (profile == "3"):
            public_function()
            break
        elif (profile == "4"):
            print("")
            break
        else:
            print("\nInvalid Input")
```

Function is a group of codes that will run whenever it’s called, in this example, if the user were to call the main_menu() function, the code inside the main_menu() will be called.

f. File Handling

1. Write Data in File

```
schedule_database = open("scheduledatabase.txt", "a")
schedule_database.write("\n"+schedule_data[0]+" \t"+schedule_data[1]+" \t"+schedule_data[2])
schedule_database.close()
```

To Write data in File, the program has to open the file first with the “a” or append mode, if the program is unable to find the text file, it will create a new text file based on the name of the text file called in the program. In this example, `schedule_database.write` will write whatever data that is programmed to be written.

2. Read Data from File

```
sport_data = open("sportdatabase.txt")
for sports in sport_data:
    order = sports.split()
    print("\nSport Code \t: ",order[0])
    print("Sport Type \t: ",order[1])
    print("Sport Center \t: ",order[2])
    print("Coach \t\t: ",order[3])
    print("Sport Fee \t: ",order[4])
sport_data.close()
```

To Read a data from a File, program has to first open the text file, with the read mode, it can be shown by the “r”, or if there are no modes, python will open it as read mode by default. In this example, “sportdatabase.txt” is opened and the code shows the data inside the text file.

3. Search/Modify Data of File

```
schedule_modify_line[line] = schedule_data[0]+"\\t"+schedule_data[1]+"\\t"+schedule_data[2]+"\\n"
schedule_modify = open("scheduledatabase.txt", "w")
schedule_modify.writelines(schedule_modify_line)
schedule_modify.close()
```

To Modify a data in File, first is to open the file with the “w” or write mode, this will allow the user to overwrite the data in the text file, In this example the program is programmed to overwrite a line, instead of the whole file.

Write mode is very dangerous when writing the code since a wrong step can overwrite the whole file, and it’s recommended to keep a copy of the original file for emergency.

Screenshot of Input/Output

```
Welcome to Sport Academy
Select a Profile :
1. Admin
2. Student
3. Public (Un-Registered)
4. Exit Program
```

A Welcome Message is the first thing that appears after the program is launched, then it asks the user 4 options, the first two options will lead into a login menu, while the third function just displays the option of the un-registered users, and the fourth option will exit the program

```
1

Admin Login
Type 'b' on username or password to go back
Enter admin username : |
```

If the user enters '1', an Admin Login menu will be executed and the user is required to enter the username of admin and the password of the admin, admin username and password are "Admin" and "Admin69".

```
Admin Login
Type 'b' on username or password to go back
Enter admin username : Admin69
Enter admin password : Admin

Invalid Username or Password
Enter admin username : |
```

If the user inputs the wrong username/password or the username/password are reversed, a message will pop up saying 'Invalid Username or Password'.

```
Invalid Username or Password
Enter admin username : Admin
Enter admin password : Admin69

Login Successful

Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu
```

After the user inputs the correct username and password, an Admin Function will pop up, asking the user of what admin function they want to execute.

```
1

Select something to be recorded :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back
```

Entering the value '1' will lead the user to "Add Record" menu, and asks the user which record they want to add

```
a

Add Records of Coach :
Coach ID :
```

If the user chooses the option 'a', the user will be asked to enter a bunch of coach informations such as, Coach ID, Coach Name, Date Joined, Date Terminated, Hourly Rate (Rm/H), Phone, Address, Sport Center Code, Sport Center Name, Sport Code, Sport Name, and Rating.

```

Add Records of Coach :
Coach ID           : 105
Coach Name         : Mingsen
Date Joined        : 12/4/2018
Date Terminated   : 12/4/2022
Hourly Rate (Rm/H) : 300
Phone              : 0231xxxxxxx
Address            : JlAyamGeprek
Sport Center Code   : 42
Sport Center Name   : MingsenSportCenter
Sport Code          : 4
Sport Name          : Archery
Rating             : 4

Enter an input :
'1' to Save and exit program
'2' to Save and add another Record
'3' to Save and go back to Admin Function
'anything' to redo
'b' to go back

```

After the user inputs every coach information, the user then will be asked if they want to save and exit the program, or save and add another program, or save and go back to admin function, the user can also enter 'any key' to redo if something were not in favor of the user, the user can also enter 'b' to go back into the admin function without saving the information. The data that has been "saved" will be appended to the "coachdatabase.txt" file

```

1

Record Saved

```

A message pops up saying "Record Saved" if the user inputs 1, which saves the information to the text file and end the program

```

2

Record Saved
Coach ID           : |

```

If the user inputs '2', a message pops up saying "Record Saved" and the program will ask the user again to enter a new record


```
3

Record Saved

Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu
```

When the user enters '3', a message also pops up saying "Record Saved", but the user will be directed to the admin function.

```
MacelGanteng
Coach ID      : |
```

If the user enters anything or any key they want, the program will loop again and ask all the coach information, this code is coded so if the user were to enter a wrong value, the user can retype it rather than the wrong value gets saved.

```
b

Select something to be recorded :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back
```

If the user enters 'b', the user will be directed to the "Add Record" menu.

```
b

Add Records of Sport :
Sport ID      : |
```

The code used for adding coach and sport is practically the same, but the users have to enter different information, in this code, the user can add a record of sport.

```

Add Records of Sport :
Sport ID      : 7
Sport Name    : Basketball
Sport Center  : AcienAcademy
Sport Coach   : Acien
Sport Fees    : 40

Enter an input :
'1' to Save and exit program
'2' to Save and add another Record
'3' to Save and go back to Admin Function
'anything' to redo
'b' to go back

```

After entering all the sport details, the user will be asked to enter an input, which will save and exit program, save and add another record, save and go back to admin function, the user can also type anything or any key to redo, and type 'b' to go back to "Add Record" menu. The data in this function will be appended in "sportdatabase.txt" file. All of the output for above code is the same as the output on the "Add Coach Record" code.

```

Select something to be recorded :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back

c

Add Records of Sport Schedule :
Sport :

```

The same code/logic is applied also for sport schedule, the user will be asked to input sport schedule information.

```
Add Records of Sport Schedule :  
Sport    : Cricket  
Date     : 21/1/2021  
Time     : 14:00  
Coach    : Leilei  
  
Enter an input :  
'1' to Save and exit program  
'2' to Save and add another Record  
'3' to Save and go back to Admin Function  
'anything' to redo  
'b' to go back
```

After entering the data for sport schedule record, user will be asked the same input for the “coach record”, and “sport record” code, the output is the same since the code is the same with the difference being the place to where these data are being saved. This particular data will be appended in “sportschedule.txt” file

```
Please select Admin function :  
1. Add Records  
2. Display Records  
3. Search Spesific Records  
4. Sort and Display Records  
5. Modify Records  
6. Back to main menu  
  
2  
  
Select a record to be displayed :  
a. Coach  
b. Sport  
c. Registered Students  
d. Go Back
```

When the user selects ‘2’ in the Admin Menu, user will be asked what records they want to see.

```

Select a record to be displayed :
a. Coach
b. Sport
c. Registered Students
d. Go Back

a

Coach ID           : 101
Coach Name         : MarcellAW
Date Joined        : 13/10/2001
Date Terminated   : 13/10/2021
Hourly Rate (Rm/H) : 499
Phone              : 0821xxxxxxxx
Address            : JlAyamPenyet
Sport Center Code   : 69
Sport Center Name   : MacelAcademy
Sport Code          : 2
Sport Name          : Badminton
Rating             : 4.8
Total Reviewers     : 5.0

Coach ID           : 102
Coach Name         : Sutong
Date Joined        : 21/11/2002
Date Terminated   : 21/11/2020
Hourly Rate (Rm/H) : 400
Phone              : 0823xxxxxxxx
Address            : JlAyamBakar
Sport Center Code   : 69
Sport Center Name   : SutongAcademy
Sport Code          : 1
Sport Name          : Swimming
Rating             : 4.5
Total Reviewers     : 4.0

```

Entering 'a' will show every coach data in the "coachdatabase.txt" file, if the user have 5 coaches recorded, all 5 will be shown.

```
b

Sport Code      : 1
Sport Type      : Swimming
Sport Center    : SutongSportCenter
Coach          : Nicholas
Sport Fee       : 20

Sport Code      : 7
Sport Type      : Basketball
Sport Center    : AcienAcademy
Coach          : Acien
Sport Fee       : 40
```

Entering 'b' will show data in "sportdatabase.txt" file

```
c

Student ID      : 69
Username        : macel69
Email           : marcellagungw@gmail.com
Place of Birth  : Surabaya
Date of Birth   : 13/10/2001
Password       : macel96

Student ID      : 70
Username        : gnotus123
Email           : vn2111@gmail.com
Place of Birth  : Samarinda
Date of Birth   : 21/11/2002
Password       : sutong123

Student ID      : 22
Username        : mingsen07
Email           : mingsengalauterus@yahoo.com
Place of Birth  : Samarinda
Date of Birth   : 12/3/2003
Password       : mingsen70
```

Entering 'c' will show registered student data, in the "studentdatabase.txt" file

```
d

Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu
```

And entering 'd' will bring the user back to Admin menu

```
Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu

3

Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back
```

On the admin menu, when the user inputs '3', the user will be asked what record they want to search.

```
a
Search Specific Records of Coach by Coach ID
Enter 'b' to go Back

Enter a Coach ID :
```

Entering 'a' will execute a code that can search a coach by his/her coach ID

```
Enter a Coach ID : 102
Coach ID          : 102
Coach Name        : Sutong
Date Joined       : 21/11/2002
Date Terminated  : 21/11/2020
Hourly Rate (Rm/H) : 400
Phone             : 0823xxxxxxxx
Address           : J1AyamBakar
Sport Center Code  : 69
Sport Center Name  : SutongAcademy
Sport Code         : 1
Sport Name         : Swimming
Rating            : 4.5
Total Reviewers    : 4.0
```

By entering a registered coach ID, the information of that coach will be displayed

```
Enter a Coach ID : 10
Invalid Coach ID, please retry or Enter 'b' to go back

Enter a Coach ID :
```

With a convenience measure, if the user enters the wrong coach ID, or enters an unregistered coach ID the program will say “Invalid Coach ID” and asks the user again to enter in the right ID

```
Enter a Coach ID : b

Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back
```

If the user enters ‘b’, the user will be directed to the search menu

```

Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back

b
Search Specific Records of Coach by Coach Rating
Enter 'b' to go Back

Enter Rating (Float) : |

```

On the search record menu, if the user inputs 'b', the user will be directed to a code that searches specific records of coach by coach rating.

```

Enter Rating (Float) : 4.8

[1]
Coach ID           : 101
Coach Name         : MarcellAW
Date Joined        : 13/10/2001
Date Terminated   : 13/10/2021
Hourly Rate (Rm/H) : 499
Phone              : 0821xxxxxxxxx
Address            : J1AyamPenyet
Sport Center Code   : 69
Sport Center Name   : MacelAcademy
Sport Code          : 2
Sport Name          : Badminton
Rating             : 4.8
Total Reviewers     : 5.0

```

If the user found a coach with the rating that the user inputted, the data of that coach will be shown.


```

Search Specific Records of Coach by Coach Rating
Enter 'b' to go Back

Enter Rating (Float) : 4

[1]
Coach ID           : 103
Coach Name         : Acien
Date Joined        : 2/3/2004
Date Terminated   : 2/3/2020
Hourly Rate (Rm/H) : 450
Phone              : 0832xxxxxxxx
Address            : JlAyamGoreng
Sport Center Code   : 69
Sport Center Name   : SutongAcademy
Sport Code          : 7
Sport Name          : Basketball
Rating             : 4
Total Reviewers     : 1

[2]
Coach ID           : 105
Coach Name         : Mingsen
Date Joined        : 12/4/2018
Date Terminated   : 12/4/2022
Hourly Rate (Rm/H) : 300
Phone              : 0231xxxxxxxx
Address            : JlAyamGeprek
Sport Center Code   : 42
Sport Center Name   : MingsenSportCenter
Sport Code          : 4
Sport Name          : Archery
Rating             : 4
Total Reviewers     : 1

```

If there are more than one coach with the rating, all coaches will be displayed.

```

Search Specific Records of Coach by Coach Rating
Enter 'b' to go Back

Enter Rating (Float) : b

Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back

b
Search Specific Records of Coach by Coach Rating
Enter 'b' to go Back

Enter Rating (Float) : test
Invalid Coach Rating, please retry or Enter 'b' to go back

Enter Rating (Float) : |

```

And as usual with any other functions, entering ‘b’ will lead the user back to the search menu, and entering the wrong value or unregistered value will loop the code and ask the user to enter the rating again.

```

Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back

c

Enter 'b' to go back
Enter a Sport ID :

```

In the search menu again, if the user inputs ‘c’, user can search the sport by entering the sport ID

```

Enter 'b' to go back
Enter a Sport ID : 14
Invalid Sport ID, please retry or Enter 'b' to go back

Enter 'b' to go back
Enter a Sport ID : 1

Sport Code      : 1
Sport Type     : Swimming
Sport Center    : SutongSportCenter
Coach          : Nicholas
Sport Fee      : 20

```

Entering the wrong value or unregistered sport will make the code to loop, and if the user enters a registered sport ID, the information about the sport will be shown.

```

d

Enter 'b' to go back
Enter a Student ID :

```

If the user inputs 'd' on the search record menu, the user can search a student by entering their student ID.

```

Enter 'b' to go back
Enter a Student ID : 69

Student ID      : 69
Username       : macel69
Email          : marcellagungw@gmail.com
Place of Birth : Surabaya
Date of Birth   : 13/10/2001
Password       : macel96

```

This is the data shown, if the student ID the user inputs is a registered student ID, if the ID wasn't registered the program will ask again the user, and if the user enters 'b' the user will be directed to the search menu.

```
Select a record to be searched :
a. Coach by Coach ID
b. Coach by Overall Performance (Rating)
c. Sport by Sport ID
d. Student by Student ID
e. Go Back

e

Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu
```

When user enters 'e' on search record menu, user will be directed to admin function.

```
Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu

4

Select a record to be Sorted and Displayed :
a. Coach in Ascending Order by Names
b. Coaches Hourly Pay Rate in Ascending Order
c. Coaches Overall Performance in Ascending Order
d. Go Back

a

Displaying Coaches by Names in Ascending Order, enter 1 to Continue or 0 to go back : |
```

on the admin function, if the user enters '4', Sort and Display menu will be executed, and user will be asked what they want to sort and be displayed, in this case, if the user selects 'a', the program will ask for confirmation.

```

Displaying Coaches by Names in Ascending Order, enter 1 to Continue or 0 to go back : 1

Coach ID      : 103
Coach Name    : Acien
Date Joined   : 2/3/2004
Date Terminated : 2/3/2020
Hourly Rate (Rm/H) : 450
Phone        : 0832xxxxxxx
Address      : JlAyamGoreng
Sport Center Code : 69
Sport Center Name : SutongAcademy
Sport Code    : 7
Sport Name    : Basketball
Rating       : 4
Total Reviewers : 1

Coach ID      : 104
Coach Name    : Jong
Date Joined   : 1/2/2013
Date Terminated : 1/2/2023
Hourly Rate (Rm/H) : 450
Phone        : 0919xxxxxxx
Address      : JlAyamHainan
Sport Center Code : 90
Sport Center Name : JongSportCenter
Sport Code    : 5
Sport Name    : Gymnastics
Rating       : 5
Total Reviewers : 1

Coach ID      : 107
Coach Name    : Leilei
Date Joined   : 6/7/2018
Date Terminated : 6/7/2023
Hourly Rate (Rm/H) : 70
Phone        : 0822xxxxxxx
Address      : JlAyamKecap
Sport Center Code : 44

```

When the user confirms, the program will display every coach sorted by names in ascending order.

```

Select a record to be Sorted and Displayed :
a. Coach in Ascending Order by Names
b. Coaches Hourly Pay Rate in Ascending Order
c. Coaches Overall Performance in Ascending Order
d. Go Back

b

Displaying Coaches by Hourly Pay Rate in Ascending Order, enter 1 to Continue or 0 to go back : |

```

If the user inputs 'd' in the sort and display function, the program will then again ask for confirmation if the user wants to continue or no, the confirmation code is applied to all three inputs, this is to prevent the user to mistakenly select the sort function, since the sort function will need a lot of memory and it will be sorting everything, if the file was huge, this process might take longer than expected.

```

Displaying Coaches by Hourly Pay Rate in Ascending Order, enter 1 to Continue or 0 to go back : 1

Coach ID      : 105
Hourly Rate (Rm/H) : 300
Coach Name    : Mingsen
Date Joined   : 12/4/2018
Date Terminated : 12/4/2022
Phone        : 0231xxxxxxx
Address      : JlAyamGeprek
Sport Center Code : 42
Sport Center Name : MingsenSportCenter
Sport Code    : 4
Sport Name    : Archery
Rating       : 4
Total Reviewers : 1

Coach ID      : 102
Hourly Rate (Rm/H) : 400
Coach Name    : Sutong
Date Joined   : 21/11/2002
Date Terminated : 21/11/2020
Phone        : 0823xxxxxxx
Address      : JlAyamBakar
Sport Center Code : 69
Sport Center Name : SutongAcademy
Sport Code    : 1
Sport Name    : Swimming
Rating       : 4.5
Total Reviewers : 4.0

Coach ID      : 103
Hourly Rate (Rm/H) : 450
Coach Name    : Acien
Date Joined   : 2/3/2004
Date Terminated : 2/3/2020
Phone        : 0832xxxxxxx
Address      : JlAyamGoreng
Sport Center Code : 69
Sport Center Name : SutongAcademy
Sport Code    : 7

```

After the user confirms, data will be sorted based on coaches hourly pay rate in ascending order.

```

Displaying Coaches by Overall Performance in Ascending Order, enter 1 to Continue or 0 to go back : 1

Coach ID      : 106
Rating        : 2
Total Reviewers : 1
Coach Name    : Leon
Date Joined   : 4/5/2016
Date Terminated : 4/5/2021
Hourly Rate (Rm/H) : 69
Phone        : 0696xxxxxxx
Address      : J1AyamRebus
Sport Center Code : 21
Sport Center Name : LeonSportCenter
Sport Code    : 6
Sport Name    : Volleyball

Coach ID      : 103
Rating        : 4
Total Reviewers : 1
Coach Name    : Acien
Date Joined   : 2/3/2004
Date Terminated : 2/3/2020
Hourly Rate (Rm/H) : 450
Phone        : 0832xxxxxxx
Address      : J1AyamGoreng
Sport Center Code : 69
Sport Center Name : SutongAcademy
Sport Code    : 7
Sport Name    : Basketball

Coach ID      : 105
Rating        : 4
Total Reviewers : 1
Coach Name    : Mingsen
Date Joined   : 12/4/2018
Date Terminated : 12/4/2022
Hourly Rate (Rm/H) : 300
Phone        : 0231xxxxxxx
Address      : J1AyamGeprek
Sport Center Code : 42

```

This is the output if the user chose 'c' which sorts the data and displays it according to coaches' overall performance (rating) in ascending order.

```

Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu

5

Select a Record to be Modified :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back

```

Now on to the 5th Admin function, which is the modify records, after the users inputs '5', the user will then be given three options, which are to modify coach, sport, or sport schedule.

```

Select a Record to be Modified :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back

a

Modify Records of Coach
[ 1 ] 101      MarcellAW      13/10/2001      13/10/2021      499      0821xxxxxxx      JIAyamPenyet
69      MacelAcademy      2      Badminton      4.8      5.0

[ 2 ] 102      Sutong      21/11/2002      21/11/2020      400      0823xxxxxxx      JIAyamBakar      69
SutongAcademy      1      Swimming      4.5      4.0

[ 3 ] 103      Acien      2/3/2004      2/3/2020      450      0832xxxxxxx      JIAyamGoreng      69
SutongAcademy      7      Basketball      4      1

[ 4 ] 105      Mingsen      12/4/2018      12/4/2022      300      0231xxxxxxx      JIAyamGeprek      42
MingsenSportCenter      4      Archery      4      1

[ 5 ] 104      Jong      1/2/2013      1/2/2023      450      0919xxxxxxx      JIAyamHainan      90
JongSportCenter      5      Gymnastics      5      1

[ 6 ] 106      Leon      4/5/2016      4/5/2021      69      0696xxxxxxx      JIAyamRebus      21
LeonSportCenter      6      Volleyball      2      1

[ 7 ] 107      Leilei      6/7/2018      6/7/2023      70      0822xxxxxxx      JIAyamKecap      44
LeileiSportCenter      8      Cricket      4      1

Select a line to be modified : |

```

If the user were to input 'a', all coach data will be shown with numbers according to their line, the user then will be asked which line they want to modify.


```

Select a line to be modified : 1
Coach ID                      : 101
Coach Name                    : Macel
Date Joined                   : 13/10/2001
Date Terminated             : 12/4/2021
Hourly Rate (RM/H)           : 500
Phone                        : 0821xxxxxxx
Address                       : JlAyamPenyet
Sport Center Code            : 69
Sport Center Name            : MacelAcademy
Sport Code                   : 2
Sport Name                   : Badminton
Rating                       : 5

Enter an input :
'1' to Save and exit program
'2' to Save and add another Record
'3' to Save and go back to Admin Function
'anything' to redo
'b' to go back

1
Record Modified Successfully

```

After the user selects a line, the user must input all data about the coach that they want to modify, and after that the user will be asked the same 5 inputs similar to the “Add Records” one, if the user modifies a coach rating, the total of reviewers will be reset to default which is one.

```

Select a Record to be Modified :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back

b

Modify Records of Sport
[ 1 ] 1 Swimming      SutongSportCenter      Nicholas      20
[ 2 ] 2 Badminton     MacelAcademy      Macel      69
[ 3 ] 3 Football      MacelAcademy      Mingsen 10
[ 4 ] 4 Archery JongSportCenter Jong      50
[ 5 ] 5 Gymnastics     JongSportCenter Jong      20
[ 6 ] 6 Volleyball     LeileiSportCenter      Leilei 25
[ 7 ] 7 Basketball     SutongAcademy      Acien      40
[ 8 ] 8 Cricket LeileiSportCenter      Leilei 30
[ 9 ] 9 Tennis MacelAcademy      Macel      69
[ 10 ] 10      TableTennis      LeonSportCenter Leon      1

Select a line to be modified :

```

When the user inputs 'b' in the modify record menu, the same output will be shown like the coach record modification, the user will be asked a line that then they can edit.

```

Select a line to be modified : 2
Sport ID      : 2
Sport Name    : Badminton
Sport Center  : MacelAcademy
Sport Coach   : Macel
Sport Fees (RM) : 69

Enter an input :
'1' to Save and exit program
'2' to Save and Modify another Record
'3' to Save and go back to Admin Function
'anything' to redo
'b' to go back

```

The user then has to input the sport data they want to modify, and they will be asked to save and exit program, or save and modify another record, save and go back to admin function, redo or go back to modify menu.

```

Select a Record to be Modified :
a. Coach
b. Sport
c. Sport Schedule
d. Go Back

c

Modify Records of Sport Schedule
[ 1 ] Swimming  24/1/2099      08:00   Sutong

[ 2 ] Badminton 22/1/2021      12:00   Marcell AW

[ 3 ] Volleyball 30/12/2020     14:30   Aaron

[ 4 ] Cricket   21/1/2021      14:00   Leilei

Select a line to be modified : 1
Sport   : Swimming
Date    : 12/4/2001
Time    : 00:00
Coach   : Macel

Enter an input :
'1' to Save and exit program
'2' to Save and Modify another Record
'3' to Save and go back to Admin Function
'anything' to redo
'b' to go back

```

Same logic

is applied to the third option in modify record menu, which is sport schedule. The user which line they want to modify, and after choosing the line, the program will ask

information on the sport schedule and it will overwrite the data on the line that the user previously entered.

```
Please select Admin function :
1. Add Records
2. Display Records
3. Search Spesific Records
4. Sort and Display Records
5. Modify Records
6. Back to main menu

6

Welcome to Sport Academy
Select a Profile :
1. Admin
2. Student
3. Public (Un-Registered)
4. Exit Program
```

The final admin function is “back to main menu” which brings the user back to the main menu and requires the user to select a profile.

```
Student Login
Type 'b' on Username or Password to go back
Enter student username : Wrong
Enter student password : w

Invalid Username or Password
Enter 'b' to go back
Enter student username : |
```

When the user selects the student profile, the user is required to enter in a registered student username and password, if the username/password is not correct or unregistered, the program will display a message saying “Invalid Username or Password”, and ask the user to enter in again, or the user can input ‘b’ to go back to main menu, incase the user selects the wrong profile.

```
Enter student username : macel69
Enter student password : macel96

Login Successful

Please select Student function :
1. View Details
2. Modify Self Record
3. Provide feedback and Star to Coach
4. Back to main menu
```

When the user has successfully login, four student function is displayed and the user can select any of these four function.

```
Please select Student function :
1. View Details
2. Modify Self Record
3. Provide feedback and Star to Coach
4. Back to main menu

1

Select a detail to be displayed :
a. Coach
b. Self-Record
c. Sport Schedule
d. Go Back
```

If the student selects the first function which is “View Details”, the user then again will be asked which detail that are going to be displayed

```

Select a detail to be displayed :
a. Coach
b. Self-Record
c. Sport Schedule
d. Go Back

a

Coach ID           : 101
Coach Name         : Macel
Date Joined        : 13/10/2001
Date Terminated   : 12/4/2021
Hourly Rate (Rm/H) : 500
Phone              : 0821xxxxxxxx
Address            : JlAyamPenyet
Sport Center Code   : 69
Sport Center Name   : MacelAcademy
Sport Code          : 2
Sport Name          : Badminton
Rating             : 5
Total Reviewers     : 1

Coach ID           : 102
Coach Name         : Sutong
Date Joined        : 21/11/2002
Date Terminated   : 21/11/2020
Hourly Rate (Rm/H) : 400
Phone              : 0823xxxxxxxx
Address            : JlAyamBakar
Sport Center Code   : 69
Sport Center Name   : SutongAcademy
Sport Code          : 1
Sport Name          : Swimming
Rating             : 4.5
Total Reviewers     : 4.0

```

Entering 'a' will show all coach data that are registered in the "coachdatabase.txt" file

```
Select a detail to be displayed :
a. Coach
b. Self-Record
c. Sport Schedule
d. Go Back

b
Details of Self, Enter 'b' on Username or Password to go back
Please Confirm your Username :
```

When the user inputs 'b', which is self-record, a confirmation is needed so privacy information of the student doesn't get leaked.

```
b
Details of Self, Enter 'b' on Username or Password to go back
Please Confirm your Username : macel69
Please Confirm Your Password : macel96

Student ID           : 69
Username             : macel69
Email                : marcellagungw@gmail.com
Place of Birth       : Surabaya
Date of Birth        : 13/10/2001
```

After the user's username and password is confirmed, record about the student will be showed, another option is to enter 'b' on the Username or Password and it will go back to the view details menu.

```
Please select Student function :
1. View Details
2. Modify Self Record
3. Provide feedback and Star to Coach
4. Back to main menu

2
Modify Self Record, Enter 'b' on Username or Password to go back

Please Confirm your Username :
```

If the user selects '2' (Modify Self Record), the user will be asked to confirm their username and password for safety precautions.

```
Please Confirm your Username : macel69
Please Confirm your Password : macele96
Invalid Username or Password, Please Retry or Enter 'b' on Username or Password to go back.
```

When the user inputs the wrong username/password, the program will display a message saying the username or password is invalid, the user is required to enter in the registered student username and password in order to proceed. The user can also enter 'b' and it will direct the user to the student function.

```
Please Confirm your Username : macel69
Please Confirm your Password : macel96

Student ID      : 69
Username       : macel69
Email          : macelhandsome@gmail.com
Place of Birth  : Surabaya
Date of Birth   : 13/10/2001
Password       : macel96

Enter an input :
'1' to Save and exit program
'2' to Save and go back to Student Function
'anything' to redo
'b' to go back and cancel modify
```

After confirming username and password, the user can then change all information about themselves, including their password. The user then gets asked if they want to save and exit program, or save and go back to student function, or redo, or go back and cancel modification.


```

Please select Student function :
1. View Details
2. Modify Self Record
3. Provide feedback and Star to Coach
4. Back to main menu

3

Provide Feedback and Star to Coach, Enter 'b' to go back
List of Coaches :
Macel
Sutong
Acien
Mingsen
Jong
Leon
Leilei

Enter a Coach name : |

```

The third student function is a function that will provide feedback and star to coach, when the user inputs '3', a list of coach names will appear, and the user will be asked who they want to rate

```

Provide Feedback and Star to Coach, Enter 'b' to go back
List of Coaches :
Macel
Sutong
Acien
Mingsen
Jong
Leon
Leilei

Enter a Coach name : test
Invalid Coach Name, Enter 'b' to go back

Enter a Coach name :

```

Entering an unregistered coach name will prompt a message saying “Invalid Coach Name, Enter ‘b’ to go back” and the user will be asked to enter a coach name again, if the user inputs ‘b’, user will be directed back to the student function.

```

Enter a Coach name : Macel

Coach ID           : 101
Coach Name         : Macel
Date Joined        : 13/10/2001
Date Terminated   : 12/4/2021
Hourly Rate (Rm/H) : 500
Phone              : 0821xxxxxxxxx
Address            : JlAyamPenyet
Sport Center Code   : 69
Sport Center Name   : MacelAcademy
Sport Code          : 2
Sport Name          : Badminton
Rating             : 5
Total Reviewers     : 1
Enter Your Rating (1-5) :

```

After entering the name of a coach they want to rate, the information about the coach is shown, and the program asks the user their rating.

```

Sport Name          : Badminton
Rating              : 5
Total Reviewers     : 1
Enter Your Rating (1-5) : 4
Rating Successfully Updated, Updated Rating : 4.5

```

The user rating will then be calculated using the present rating, the total reviewers and the newly added rating. The calculation is performed by multiplying the total reviewers to the present rating, and adding the newly added rating which gives the total rating, and the total rating will then be divided by the reviewers+1, because of the new reviewer, this will give the new rating.

```
Please select Student function :
1. View Details
2. Modify Self Record
3. Provide feedback and Star to Coach
4. Back to main menu

4

Welcome to Sport Academy
Select a Profile :
1. Admin
2. Student
3. Public (Un-Registered)
4. Exit Program
```

The fourth student function will lead the user back to main menu, where the user can exit program or select another profile.

```
Welcome to Sport Academy
Select a Profile :
1. Admin
2. Student
3. Public (Un-Registered)
4. Exit Program

3

Please select Public function :
1. View Details
2. Register as Student
3. Back to Main Menu
```

The last profile of this program is a public (un-registered) profile, when the user selects this profile, they have three options, to view details, to register as students and go back to main menu (incase the user selects the wrong profile)

```

Please select Public function :
1. View Details
2. Register as Student
3. Back to Main Menu

1

Select a detail to be displayed :
a. Sport
b. Sport Schedule
c. Back to main menu

```

When the user selects the first option, which is view details, the user can only see 2 records, which are sport and sport schedule.

```

Select a detail to be displayed :
a. Sport
b. Sport Schedule
c. Back to main menu

a

Sport Code      : 1
Sport Type      : Swimming
Sport Center    : SutongSportCenter
Coach           : Nicholas
Sport Fee       : 20

Sport Code      : 2
Sport Type      : Badminton
Sport Center    : MacelAcademy
Coach           : Macel
Sport Fee       : 69

Sport Code      : 3
Sport Type      : Football
Sport Center    : MacelAcademy
Coach           : Mingsen
Sport Fee       : 10

Sport Code      : 4
Sport Type      : Archery
Sport Center    : JongSportCenter
Coach           : Jong
Sport Fee       : 50

```

Selecting the sport detail will show all the registered sports in the “sportdatabase.txt” file

```
Select a detail to be displayed :
a. Sport
b. Sport Schedule
c. Back to main menu
```

b

```
[1]
Sport Type      : Swimming
Date            : 12/4/2001
Time            : 00:00
Sport Coach     : Macel

[2]
Sport Type      : Badminton
Date            : 22/1/2021
Time            : 12:00
Sport Coach     : Marcell

[3]
Sport Type      : Volleyball
Date            : 30/12/2020
Time            : 14:30
Sport Coach     : Aaron

[4]
Sport Type      : Cricket
Date            : 21/1/2021
Time            : 14:00
Sport Coach     : Leilei
```

If the user selects the 'b' input, the user will be able to see all details in the "sportschedule.txt" file

```
c

Please select Public function :
1. View Details
2. Register as Student
3. Back to Main Menu
```

Selecting the third option will bring back the user to the public function menu

```

Please select Public function :
1. View Details
2. Register as Student
3. Back to Main Menu

2

Student Registration
Enter 'b' on StudentID or Password to go back

Student ID      : |

```

If the user selects the “register as student” function, the user can register as a student.

```

Student Registration
Enter 'b' on StudentID or Password to go back

Student ID      : a
Username        : b
Email           : c
Place of Birth  : d
Date of Birth   : e
Password        : f
Confirm your Password : g

Password doesn't match, Please try again
or type 'b' on StudentID or Password to go back

Student ID      : |

```

The last input on the registration function is to confirm the password, if the password and the confirm password doesn’t match, the program will pop a message saying “password doesn’t match” and requires the user to enter again all the details, the user can also go back to the public function by entering ‘b’ on StudentID or Password.

```
Student Registration
Enter 'b' on StudentID or Password to go back

Student ID           : 992
Username             : edsel1
Email                : edseljagovalorant@yahoo.com
Place of Birth       : Samarinda
Date of Birth        : 2/3/2004
Password             : edsel2
Confirm your Password : edsel2
Registration Successful
Proceeding to Student Login Function

Student Login
Type 'b' on Username or Password to go back
Enter student username : |
```

When the registration function is successful, the user will be directed to the student login function for user convenience.

Additional Features

- a. Split, Split will split a string into a string, and makes each item in the list into an item, for example.

coachdatabase.txt - Notepad

101	Macel	13/10/2001	12/4/2021	500	0821xxxxxxx	JlAyamPenyet	69	MacelAcademy	2	Badminton	4.5	2.0
102	Sutong	21/11/2002	21/11/2020	400	0823xxxxxxx	JlAyamBakar	69	SutongAcademy	1	Swimming	4.5	4.0
103	Acien	2/3/2004	2/3/2020	450	0832xxxxxxx	JlAyamGoreng	69	SutongAcademy	7	Basketball	4	1
105	Mingsen	12/4/2018	12/4/2022	300	0231xxxxxxx	JlAyamGeprek	42	MingsenSportCenter		4 Archery	4	1
104	Jong	1/2/2013	1/2/2023	450	0919xxxxxxx	JlAyamHainan	90	JongSportCenter	5	Gymnastics	5	1
106	Leon	4/5/2016	4/5/2021	69	0696xxxxxxx	JlAyamRebus	21	LeonSportCenter	6	Volleyball	2	1
107	Leilei	6/7/2018	6/7/2023	70	0822xxxxxxx	JlAyamKecap	44	LeileiSportCenter	8	Cricket	4	1

This is a text file containing coach data base, by using split, we can split the lists into an item, and we can print out each of these items.

Code:

```
for coaches in coach_data:
    order = coaches.split()
    if(coachID == order[0]):
        print("Coach ID \t\t: ",order[0])
        print("Coach Name \t\t: ",order[1])
        print("Date Joined \t\t: ",order[2])
        print("Date Terminated \t: ",order[3])
        print("Hourly Rate (Rm/H) \t: ",order[4])
        print("Phone \t\t\t: ",order[5])
        print("Address \t\t: ",order[6])
        print("Sport Center Code \t: ",order[7])
        print("Sport Center Name \t: ",order[8])
        print("Sport Code \t\t: ",order[9])
        print("Sport Name \t\t: ",order[10])
        print("Rating \t\t\t: ",order[11])
        print("Total Reviewers \t: ",order[12])
```

Output:

```
Enter a Coach ID : 102
Coach ID          : 102
Coach Name        : Sutong
Date Joined       : 21/11/2002
Date Terminated : 21/11/2020
Hourly Rate (Rm/H) : 400
Phone             : 0823xxxxxxx
Address           : JlAyamBakar
Sport Center Code : 69
Sport Center Name : SutongAcademy
Sport Code        : 1
Sport Name        : Swimming
Rating            : 4.5
Total Reviewers   : 4.0
```


- b. Correction, each function in this program has a correction, which means if the user enters a wrong input, the user can always go back for user convenience.

```
if (verify == "1"):
    print ("\nRecord Saved")
    schedule_database.write("\n")
    schedule_database.close()
    break
if (verify == "2"):
    print ("\nRecord Saved")
    schedule_database.write("\n")
    schedule_database.close()
if (verify == "3"):
    print ("\nRecord Saved")
    schedule_database.write("\n")
    schedule_database.close()
    admin_function()
    break
elif (verify == "b"):
    add_records()
    break
```

In this example code, if the user were to accidentally execute this function, the user does not have to restart the program, but rather, there's an option for the user to go back, in this case, if the user enters 'b' for verify, the user will be directed back to a function before this function was executed.

Conclusion

1. It is concluded the system is able to ask the user to login to Student, Public, or Admin Account
2. It is concluded the admin can add records of :
 - a. Coaches
 - b. Sports
 - c. Sport Schedule
3. It is concluded the admin can display all records of:
 - a. Coaches
 - b. Sports
 - c. Registered Students
4. It is concluded the admin can search a specific records of:
 - a. Coaches by Coach ID
 - b. Coaches by overall Performance (Rating)
 - c. Sports by Sport ID
 - d. Students by Student ID
5. It is concluded the admin can sort and display records of:
 - a. Coaches in ascending order by names
 - b. Coaches hourly pay rate in ascending order
 - c. Coaches overall performance in ascending order
6. It is concluded the admin can modify records:
 - a. Coaches
 - b. Sports
 - c. Sport Schedule
7. It is concluded the public user can view details of:
 - a. Sports
 - b. Sport Schedules
8. It is concluded the public user can register as a new student
9. It is concluded the student user can view details of:
 - a. Coaches
 - b. Self-Record
 - c. Registered Sport Schedule
10. It is concluded the student user can modify self-record
11. It is concluded the student can provide feedback and star to coach

References

GeeksForGeeks, 2020. *Python Program for Bubble Sort*. [Online]

Available at: <https://www.geeksforgeeks.org/python-program-for-bubble-sort/>

[Accessed 20 January 2021].