# Section 1:

**Functional requirements:**

**There will be 4 categories of users in the system:**

* Admin
* Staff
* Residents
* Public

1. **Admin:**

As an admin I can:

* View all applications.
* Add staff or resident.
* Remove staff or resident.
* Can view all members.
* Can view resident fees in Descending order.
* View total expenses of a month.
* Generate fees.
* Make notice on the notice board.

1. **Staff:**

As a Staff I can:

* Mark Attendance.
* View all members.

1. **Resident:**

As a Resident I can:

* View my pending fees.
* View my mess Attendance.

1. **Public:**

As Public I can:

* Apply for allotment or employment.
* View notice board.
* View their Application Status.

**Project Vision:**

For the past few years the number of educational institutions is increasing rapidly. Thereby the number of hostels is also increasing, for the accommodation of the students studying in these institutions. And hence there is a lot of strain on the people that are running these hostels, and software solutions are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing systems.

# Section 2:

**Limitation Faced in Phase 1:**

Using simple input/output we can perform simple calculations, but we can’t do decision making which severely limits our range of applications.

Since my project depends a lot on decision making. I was only able to implement a simple mess fees calculator.

**Code:**

header = "                          \_\_  \_\_           \_\_       \_\_\n"\

"                         / / / /\_\_\_  \_\_\_\_\_/ /\_\_\_\_  / /\n"\

"                        / /\_/ / \_\_ \\/ \_\_\_/ \_\_/ \_ \\/ /\n"\

"                       / \_\_  / /\_/ (\_\_  ) /\_/  \_\_/ /\n"\

"                      /\_/ /\_/\\\_\_\_\_/\_\_\_\_/\\\_\_/\\\_\_\_/\_/\n"\

"           \_\_  \_\_\_                                                  \_\_\n"\

"          /  |/  /\_\_\_ \_\_\_\_\_  \_\_\_\_ \_\_\_\_\_ \_\_\_\_  \_\_\_\_ \_\_\_  \_\_\_  \_\_\_\_  / /\_\n"\

"         / /|\_/ / \_\_ `/ \_\_ \\/ \_\_ `/ \_\_ `/ \_ \\/ \_\_ `\_\_ \\/ \_ \\/ \_\_ \\/ \_\_/\n"\

"        / /  / / /\_/ / / / / /\_/ / /\_/ /  \_\_/ / / / / /  \_\_/ / / / /\_\n"\

"       /\_/  /\_/\\\_\_,\_/\_/ /\_/\\\_\_,\_/\\\_\_, /\\\_\_\_/\_/ /\_/ /\_/\\\_\_\_/\_/ /\_/\\\_\_/\n"\

"                                /\_\_\_\_/\n"\

"                      \_\_\_\_\_            \_\_\n"\

"                     / \_\_\_/\_\_  \_\_\_\_\_\_\_/ /\_\_\_\_  \_\_\_\_ \_\_\_\n"\

"                     \\\_\_ \\/ / / / \_\_\_/ \_\_/ \_ \\/ \_\_ `\_\_ \\  \n"\

"                    \_\_\_/ / /\_/ (\_\_  ) /\_/  \_\_/ / / / / /\n"\

"                   /\_\_\_\_/\\\_\_, /\_\_\_\_/\\\_\_/\\\_\_\_/\_/ /\_/ /\_/\n"\

"                        /\_\_\_\_/ \n"

print(header)

print("\nMess Fee Calculator:\n")

att = int(input("Enter the no. of times attended: "))

cost = int(input("Enter the price per meal: "))

total = att \* cost

print("The total bill is: ", total, "rupees.")

# Section 3:

**Improvement from previous phase:**

Using conditions and loops greatly increased the scope of the project. Many new functionalities were be added and the program grew to be much bigger.

I was able to make a program that runs continuously instead of terminating every time after completing its task, using loops and was able to implement decision making to perform different task depending on the user’s choice rather than performing a single operation.

**Limitation Faced in Phase 2:**

Lots of copy paste was needed and I wasn’t able store many users because a large number of variables was needed to do so. The code wasn’t reusable and was very in-efficient.

**Code:**

import os

staff1 = ''

staff2 = ''

staff3 = ''

staff1pass = ''

staff2pass = ''

staff3pass = ''

user1 = ''

user2 = ''

user3 = ''

user1pass = ''

user2pass = ''

user3pass = ''

user1att = 0

user2att = 0

user3att = 0

user1fees = ''

user2fees = ''

user3fees = ''

expenses = 0

admin\_name = "admin"

admin\_pass = "1234"

in\_menu = True

while in\_menu:

    os.system("cls")

    print("\n\t\tMain Menu > ",

          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------",

          "1. Login",

          "2. Exit", sep='\n')

    choice = input("\nEnter your choice: ")

    if choice == '1':

        os.system("cls")

        print("\n\t\tMain Menu > Login > ",

              "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tLogin as:\n",

              "1. Administrator",

              "2. Resident",

              "3. Staff",

              "4. Go back", sep='\n')

        choice = input("\nEnter your choice: ")

        if choice == '1':

            name\_in = input("\nEnter admin name: ")

            pass\_in = input("Enter admin password: ")

            if name\_in == admin\_name and pass\_in == admin\_pass:

                in\_admin = True

                while in\_admin:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Admin > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. Add a user",

                          "2. Remove a user",

                          "3. Add expenses",

                          "4. Calculate expenses",

                          "5. View all members",

                          "6. View resident fees",

                          "7. Logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("1. Add staff",

                              "2. Add resident",

                              sep='\n')

                        choice = input("\nYour choice: ")

                        if choice == '1':

                            if staff1 == '':

                                staff = input("Enter name for new staff: ")

                                staff1pass = input("Enter password for new staff: ")

                            elif staff2 == '':

                                staff2 = input("Enter name for new staff: ")

                                staff2pass = input("Enter password for new staff: ")

                            elif staff3 == '':

                                staff3 = input("Enter name for new staff: ")

                                staff3pass = input("Enter password for new staff: ")

                            else:

                                input(

                                    "Maximum User capacity has been reached\nPress ENTER to continue...")

                    if choice == '2':

                        name = input("Enter user name: ")

                        if name == user1:

                            user1 = ''

                            user1pass = ''

                            input(

                                name, "Removed successfully\nPress ENTER to continue...")

                        elif name == user2:

                            user2 = ''

                            user2pass = ''

                            input(

                                name, "Removed successfully\nPress ENTER to continue...")

                        elif name == user3:

                            user3 = ''

                            user3pass = ''

                            input(

                                name + " removed successfully\nPress ENTER to continue...")

                        else:

                            input("User not found\nPress ENTER to continue...")

                    if choice == '3':

                        expenses += int(input("Enter the expenses: "))

                        input("\nPress ENTER to continue...")

                    if choice == '4':

                        print("The total expenses are:", (user1att \* 50 + user2att \* 50 + user3att \* 50) + expenses)

                        input("\nPress ENTER to continue...")

                    if choice == '5':

                        print("Name\tPassword")

                        if user1 != '':

                            print(user1, user1pass, sep='\t')

                        if user2 != '':

                            print(user2, user2pass, sep='\t')

                        if user3 != '':

                            print(user3, user3pass, sep='\t')

                        input("\nPress ENTER to continue...")

                    if choice == '6':

                        print("Name\tFees")

                        if user1 != '':

                            print(user1, user1att \* 50, sep='\t')

                        if user2 != '':

                            print(user2, user2att \* 50, sep='\t')

                        if user3 != '':

                            print(user3, user3att \* 50, sep='\t')

                        input("\nPress ENTER to continue...")

                    if choice == '7':

                        in\_admin = False

                        input("Logging out\nPress ENTER to continue...")

                input("\nPress ENTER to continue...")

            else:

                input("Invalid name or password\nPress ENTER to continue...")

        elif choice == '2':

            name = input("Enter user name: ")

            password = input("Enter user password: ")

            if name == user1 and password == user1pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Resident > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View this month\'s bill",

                          "2. View attendance",

                          "3. logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("This month's total bill is: ", user1att \* 50)

                    elif choice == '2':

                        print("This month's total attendance is: ", user1att)

                    elif choice == '3':

                        in\_user = False

            elif name == user2 and password == user2pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Resident > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View this month\'s bill",

                          "2. View attendance",

                          "3. logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("This month's total bill is: ", user2att \* 50)

                        input("\nPress ENTER to continue...")

                    elif choice == '2':

                        print("This month's total attendance is: ", user2att)

                        input("\nPress ENTER to continue...")

                    elif choice == '3':

                        in\_user = False

            elif name == user3 and password == user3pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Resident > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View this month\'s bill",

                          "2. View attendance",

                          "3. logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("This month's total bill is: ", user3att \* 50)

                        input("\nPress ENTER to continue...")

                    elif choice == '2':

                        print("This month's total attendance is: ", user3att)

                        input("\nPress ENTER to continue...")

                    elif choice == '3':

                        in\_user = False

        elif choice == '3':

            name = input("Enter staff name: ")

            password = input("Enter staff password: ")

            if name == staff1 and password == staff1pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Staff > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View all Residents",

                          "2. Add Attendance",

                          "3. Logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("Name: ")

                        if user1 != '':

                            print(user1)

                        if user2 != '':

                            print(user2)

                        if user3 != '':

                            print(user3)

                            input("\nPress ENTER to continue...")

                    elif choice == '2':

                        name = input("Enter user name: ")

                        if name == user1:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user2:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user3:

                            user1att = int(input("Enter user attendence: "))

                        else:

                            input("Username not found!\nPress any key to continue...")

                        input("\nPress ENTER to continue...")

                    elif choice == '3':

                        in\_user = False

            elif name == staff2 and password == staff2pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Staff > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View all Residents",

                          "2. Add Attendance",

                          "3. Logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("Name: ")

                        if user1 != '':

                            print(user1)

                        if user2 != '':

                            print(user2)

                        if user3 != '':

                            print(user3)

                        input("\nPress ENTER to continue...")

                    elif choice == '2':

                        name = input("Enter user name: ")

                        if name == user1:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user2:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user3:

                            user1att = int(input("Enter user attendence: "))

                        else:

                            input("Username not found!\nPress any key to continue...")

                        input("\nPress ENTER to continue...")

                    elif choice == '3':

                        in\_user = False

            elif name == staff3 and password == staff3pass:

                in\_user = True

                while in\_user:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Staff > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. View all Residents",

                          "2. Add Attendance",

                          "3. Logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("Name: ")

                        if user1 != '':

                            print(user1)

                        if user2 != '':

                            print(user2)

                        if user3 != '':

                            print(user3)

                        input("\nPress ENTER to continue...")

                    elif choice == '2':

                        name = input("Enter user name: ")

                        if name == user1:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user2:

                            user1att = int(input("Enter user attendence: "))

                        elif name == user3:

                            user1att = int(input("Enter user attendence: "))

                        else:

                            input("Username not found!\nPress any key to continue...")

                        input("\nPress ENTER to continue...")

                    elif choice == '3':

                        in\_user = False

    if choice == '2':

        in\_menu = False

# Section 4:

**Improvement from previous phase:**

After adding lists as data storage instead of simple variables, the range of the program is expanded greatly. It can now be used for any number of users. Another big change is the removal of a lot clutter and stuff that was copy pasted due to limitations of simple variables.

**Limitation Faced in Phase 3:**

Even though the program is greatly improved, it is still hard to maintain and understand. There are some part of code that are repeated again and again, which causes redundancy.

**Code:**

import os

staff\_names = []

staff\_passwords = []

user\_names = []

user\_passwords = []

user\_atts = []

expenses = 0

admin\_name = "admin"

admin\_pass = "1234"

in\_menu = True

while in\_menu:

    os.system("cls")

    print("\n\t\tMain Menu > ",

          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------",

          "1. Login",

          "2. Exit", sep='\n')

    choice = input("\nEnter your choice: ")

    if choice == '1':

        os.system("cls")

        print("\n\t\tMain Menu > Login > ",

              "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tLogin as:\n",

              "1. Administrator",

              "2. Resident",

              "3. Staff",

              "4. Go back", sep='\n')

        choice = input("\nEnter your choice: ")

        if choice == '1':

            name\_in = input("\nEnter admin name: ")

            pass\_in = input("Enter admin password: ")

            if name\_in == admin\_name and pass\_in == admin\_pass:

                in\_admin = True

                while in\_admin:

                    os.system("cls")

                    print("\n\t\tMain Menu > Login > Admin > ",

                          "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                          "1. Add a user",

                          "2. Remove a user",

                          "3. Add expenses",

                          "4. Calculate expenses",

                          "5. View all members",

                          "6. View resident fees",

                          "7. Logout", sep='\n')

                    choice = input("\nEnter your choice: ")

                    if choice == '1':

                        print("1. Add staff",

                              "2. Add resident",

                              sep='\n')

                        choice = input("\nYour choice: ")

                        if choice == '1':

                            staff = input("Enter name for new staff: ")

                            staff\_pass = input("Enter password for new staff: ")

                            staff\_names.append(staff)

                            staff\_passwords.append(staff\_pass)

                        elif choice == '2':

                            user = input("Enter name for new user: ")

                            user\_pass = input("Enter password for new user: ")

                            user\_names.append(user)

                            user\_passwords.append(user\_pass)

                            user\_atts.append(0)

                    elif choice == '2':

                        name = input("Enter user name: ")

                        if name in user\_names:

                            idx = user\_names.index(name)

                            user\_names.remove(idx)

                            user\_passwords.remove(idx)

                            user\_atts.remove(idx)

                        else:

                            input("Username not found!\nPress any key to continue...")

                    elif choice == '3':

                        expenses += int(input("Enter the expenses: "))

                        input("\nPress ENTER to continue...")

                    elif choice == '4':

                        total = 0

                        for attendence in user\_atts:

                            total += attendence \* 50

                        print("The total expenses are:", total)

                        input("\nPress ENTER to continue...")

                    elif choice == '5':

                        print("Name\tPassword")

                        for user, password in zip(user\_names, user\_passwords):

                            print(user, password, sep='\t')

                        input("\nPress ENTER to continue...")

                    elif choice == '6':

                        print("Name\tFees")

                        for user, attendance in zip(user\_names, user\_atts):

                            print(user, attendance, sep='\t')

                        input("\nPress ENTER to continue...")

                    elif choice == '7':

                        in\_admin = False

                        input("Logging out\nPress ENTER to continue...")

                input("\nPress ENTER to continue...")

            else:

                input("Invalid name or password\nPress ENTER to continue...")

        elif choice == '2':

            name = input("Enter user name: ")

            password = input("Enter user password: ")

            if name in user\_names:

                idx = user\_names.index(name)

                if password == user\_passwords[idx]:

                    in\_user = True

                    while in\_user:

                        os.system("cls")

                        print("\n\t\tMain Menu > Login > Resident > ",

                            "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                            "1. View this month\'s bill",

                            "2. View attendance",

                            "3. logout", sep='\n')

                        choice = input("\nEnter your choice: ")

                        if choice == '1':

                            print("This month's total bill is: ", user\_atts[idx] \* 50)

                        elif choice == '2':

                            print("This month's total attendance is: ", user\_atts[idx])

                        elif choice == '3':

                            in\_user = False

                        input("\nPress any key to continue...")

        elif choice == '3':

            name = input("Enter staff name: ")

            password = input("Enter staff password: ")

            if name in staff\_names:

                idx = staff\_names.index(name)

                if password == staff\_passwords[idx]:

                    in\_user = True

                    while in\_user:

                        os.system("cls")

                        print("\n\t\tMain Menu > Login > Staff > ",

                            "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

                            "1. View all Residents",

                            "2. Add Attendance",

                            "3. Logout", sep='\n')

                        choice = input("\nEnter your choice: ")

                        if choice == '1':

                            print("Name: ")

                            for user\_name in user\_names:

                                print(user\_name)

                        elif choice == '2':

                            name = input("Enter user name: ")

                            if name in user\_names:

                                user\_atts[user\_names.index(name)] = int(input("Enter user attendence: "))

                            else:

                                input("Username not found!\nPress any key to continue...")

                            input("\nPress ENTER to continue...")

                        elif choice == '3':

                            in\_user = False

                        input("\nPress ENTER to continue...")

    elif choice == '2':

        in\_menu = False

# Section 5:

**Improvement from previous phase:**

The redundancy is almost completely removed. Functions made it easier to re-use code without copy pasting. We can also make functions independent of the user interface and write code that can be used for any kind of interface.

**Limitation Faced in Phase 4:**

We still can’t do anything about it if the user enters invalid information. We don’t have a way to validate user input and save the data. The user may enter invalid data and crash the program. We need to take measures to avoid that.

**Code:**

import os

staff\_names = []

staff\_passwords = []

user\_names = []

user\_passwords = []

user\_atts = []

notices = []

applications = []

fees = []

expenses = 0

sorted\_indexes = []

sorted\_fees = []

sorted\_names = []

admin\_pass = "1234"

def main():

    load\_demo\_data()

    while True:

        os.system("cls")

        display\_header()

        option = display\_menu(get\_menu\_items("main"))

        if option == '1':

            login\_option = '-1'

            while login\_option != '4':

                os.system("cls")

                display\_header()

                login\_option = display\_menu(get\_menu\_items("login"))

                if login\_option == '1':  # Admin

                    pass\_check = input("\n\nEnter Password: ")

                    if pass\_check == admin\_pass:

                        admin\_option = '-1'

                        while admin\_option != '12':  # Admin menu loop

                            os.system("cls")

                            display\_header()

                            admin\_option = display\_menu(

                                get\_menu\_items("admin"))

                            if admin\_option == '1':  # Adding User

                                add\_option = display\_menu(

                                    get\_menu\_items("add"))

                                if add\_option == '1':

                                    add\_staff\_ui()

                                    wait("")

                                elif add\_option == '2':

                                    add\_resident\_ui()

                                    wait("")

                                elif add\_option == '3':

                                    wait("")

                            elif admin\_option == '2':  # Removing User

                                removing\_option = display\_menu(

                                    get\_menu\_items("remove"))

                                if removing\_option == '1':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("staff", name)

                                    wait("")

                                elif removing\_option == '2':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("resident", name)

                                    wait("")

                                elif removing\_option == '3':

                                    wait("")

                            elif admin\_option == '3':  # View all apllications

                                view\_applications()

                                wait("")

                            elif admin\_option == '4':  # Remove Application

                                remove\_application()

                                wait("")

                            elif admin\_option == '5':  # Add expenses

                                expenses = int(input("\n\t\tEnter expenses: "))

                                print("This month's expenses are: ", expenses)

                                wait("")

                            elif admin\_option == '6':  # Calculate expenses

                                calculate\_fees()

                                wait("")

                            elif admin\_option == '7':  # View expenses

                                print("This month's expenses are: ", expenses)

                                wait("")

                            elif admin\_option == '8':  # Make a notice on the notice board

                                add\_notice\_ui()

                                wait("")

                            elif admin\_option == '9':  # Display notices

                                show\_notices()

                                wait("")

                            elif admin\_option == '10':  # View all members

                                display\_members(

                                    "Staff", staff\_names, staff\_passwords)

                                display\_members(

                                    "Residents", user\_names, user\_passwords)

                                wait("")

                            elif admin\_option == '11':  # View resident fees

                                calculate\_fees()

                                sort()

                                display\_ordered\_residents()

                                wait("")

                            elif admin\_option == '12':  # Logout

                                wait("")

                    else:

                        print("\nInvalid Password!!")

                        wait("")

                elif login\_option == '2':  # Resident

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(user\_names, user\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            resident\_option = '-1'

                            user\_idx = user\_names.index(name)

                            resident\_option = display\_menu(

                                get\_menu\_items("resident"))

                            if resident\_option == '1':  # Bill for current month

                                calculate\_fees()

                                print(

                                    "\nYour Bill for the Current Month is: ", fees[user\_idx])

                                wait("")

                            elif resident\_option == 2:  # Attendance for the Current Month

                                print(

                                    "\nYour Attendance for the Current Month is: ", user\_atts[user\_idx])

                                wait("")

                            elif resident\_option == 3:  # Logout

                                logi = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == 3:  # Staff

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(staff\_names, staff\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            staff\_option = '-1'

                            user\_idx = staff\_names.index(name)

                            staff\_option = display\_menu(

                                get\_menu\_items("staff"))

                            if staff\_option == '1':  # Display members

                                display\_members(

                                    "residents", user\_names, user\_passwords)

                                wait("")

                            elif staff\_option == '2':  # Add attendance

                                add\_attendance\_ui()

                                wait("")

                            elif staff\_option == '3':  # logout

                                login = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == '4':  # go back

                    wait("")

        elif option == '2':  # View Notice Board

            show\_notices()

            wait("")

        elif option == '3':  # Apply for Allotment

            make\_application()

            wait("")

        elif option == 4:  # View application status

            check\_status()

            wait("")

        elif option == 5:  # Close program

            print("\n\t\tThank you for using the software,")

            wait("")

def display\_header():

    header = ["                          \_\_  \_\_           \_\_       \_\_",

              "                         / / / /\_\_\_  \_\_\_\_\_/ /\_\_\_\_  / /",

              "                        / /\_/ / \_\_ \\/ \_\_\_/ \_\_/ \_ \\/ /",

              "                       / \_\_  / /\_/ (\_\_  ) /\_/  \_\_/ /",

              "                      /\_/ /\_/\\\_\_\_\_/\_\_\_\_/\\\_\_/\\\_\_\_/\_/",

              "           \_\_  \_\_\_                                                  \_\_",

              "          /  |/  /\_\_\_ \_\_\_\_\_  \_\_\_\_ \_\_\_\_\_ \_\_\_\_  \_\_\_\_ \_\_\_  \_\_\_  \_\_\_\_  / /\_",

              "         / /|\_/ / \_\_ `/ \_\_ \\/ \_\_ `/ \_\_ `/ \_ \\/ \_\_ `\_\_ \\/ \_ \\/ \_\_ \\/ \_\_/",

              "        / /  / / /\_/ / / / / /\_/ / /\_/ /  \_\_/ / / / / /  \_\_/ / / / /\_",

              "       /\_/  /\_/\\\_\_,\_/\_/ /\_/\\\_\_,\_/\\\_\_, /\\\_\_\_/\_/ /\_/ /\_/\\\_\_\_/\_/ /\_/\\\_\_/",

              "                                /\_\_\_\_/",

              "                      \_\_\_\_\_            \_\_",

              "                     / \_\_\_/\_\_  \_\_\_\_\_\_\_/ /\_\_\_\_  \_\_\_\_ \_\_\_",

              "                     \\\_\_ \\/ / / / \_\_\_/ \_\_/ \_ \\/ \_\_ `\_\_ \\  ",

              "                    \_\_\_/ / /\_/ (\_\_  ) /\_/  \_\_/ / / / / /",

              "                   /\_\_\_\_/\\\_\_, /\_\_\_\_/\\\_\_/\\\_\_\_/\_/ /\_/ /\_/",

              "                        /\_\_\_\_/ "]

    for line in header:

        print(line)

def display\_menu(menu):

    for item in menu:

        print(item)

    selection = input("\nYour Choice: ")

    return selection

def get\_menu\_items(menu\_name):

    menu\_names = ["main", "login", "admin",

                  "resident", "staff", "add", "remove"]

    menus = [[

        "\n\t\tMain Menu > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------",

        "1. Login",

        "2. View Notice Board",

        "3. Apply for Allotment/Employment",

        "4. View status",

        "5. Exit"

    ], [

        "\n\t\tMain Menu > Login > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tLogin as:\n",

        "1. Administrator",

        "2. Resident",

        "3. Staff",

        "4. Go back"

    ], [

        "\n\t\tMain Menu > Login > Admin > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add a user",

        "2. Remove a user",

        "3. View applications",

        "4. Remove applications",

        "5. Add expenses",

        "6. Calculate expenses",

        "7. View this month\'s expenses",

        "8. Make a notice on the notice board",

        "9. View Notice board",

        "10. View all members",

        "11. View resident fees",

        "12. Logout"

    ], [

        "\n\t\tMain Menu > Login > Resident > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View this month\'s bill",

        "2. View attendance",

        "3. logout"

    ], [

        "\n\t\tMain Menu > Login > Staff > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View all Residents",

        "2. Add Attendance",

        "3. Logout"

    ], [

        "\n\t\tMain Menu > Login > Admin > Add User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add Staff",

        "2. Add Resident",

        "3. Go Back"

    ], [

        "\n\t\tMain Menu > Login > Staff > Remove User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Remove Staff",

        "2. Remove Resident",

        "3. Go Back"

    ]]

    if menu\_name in menu\_names:

        return menus[menu\_names.index(menu\_name)]

def store\_user(category, user\_name, password):

    if category == "staff":

        staff\_names.append(user\_name)

        staff\_passwords.append(password)

    elif category == "resident":

        user\_names.append(user\_name)

        user\_passwords.append(password)

        user\_atts.append(0)

def add\_resident\_ui():

    name = input("Enter resident name: ")

    password = input("Enter resident password: ")

    store\_user("resident", name, password)

def add\_staff\_ui():

    name = input("Enter staff name: ")

    password = input("Enter staff password: ")

    store\_user("staff", name, password)

def remove\_member(category, name):

    found = False

    if category == "resident":

        if name in user\_names:

            idx = user\_names.index(name)

            user\_names.pop(idx)

            user\_passwords.pop(idx)

            found = True

    elif category == "staff":

        if name in staff\_names:

            idx = staff\_names.index(name)

            staff\_names.pop(idx)

            staff\_passwords.pop(idx)

            found = True

    if found:

        print(name, "has been removed from", category)

    else:

        print("Couldn't find", name, "in", category)

def view\_applications():

    for application in applications:

        print(application)

        print("---------------------------------------------------------------------------")

def remove\_application():

    app\_no = int(

        input("Enter application number you want to be removed: ")) - 1

    if app\_no < len(applications):

        applications.pop(app\_no)

    else:

        print("Cannot find the application")

def display\_members(category, users, passwords):

    print(category + ":")

    print("------- -------- -------- -------- -------- -------- -------- ---------")

    print("Username\t|\tPassword", end='')

    if category == "Residents":

        print("\t|\tAttendance", end='')

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

    for i in range(len(users)):

        print(users[i] , "\t\t|\t", passwords[i], end='')

        if category == "Residents":

            print("\t\t|\t", user\_atts[i], end='')

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

def add\_notice\_ui():

    notice = input("\nEnter Notice: ")

    store\_notice(notice)

def store\_notice(notice):

    notices.append(notice)

def show\_notices():

    for notice in notices:

        print(notice)

def login(names, passwords, name, password):

    if name in names:  # find name

        idx = names.index(name)

        if passwords[idx] == password:  # check password

            return True

    return False

def add\_attendance\_ui():

    name = input("Enter user name: ")

    if name in user\_names:

        idx = user\_names.index(name)

        att = int(input("Enter attendance: "))

        store\_attendance(att, idx)

def store\_attendance(attendance, index):

    user\_atts[index] = attendance

def make\_application():

    applications.append(input(

        "Please enter your name\n and the password you want to use\nand the category you want to apply for\n(seperate them with \'\_\'):\n\nInput: "))

def wait(optional\_string):

    input(optional\_string + "\nPress ENTER key to continue...")

def calculate\_fees():

    global fees

    fees = []

    for att in user\_atts:

        fees.append(att \* 80)

def sort():

    global sorted\_indexes

    global sorted\_fees

    global sorted\_names

    sorted\_indexes = []

    copy\_fees = fees.copy()

    sorted\_fees = copy\_fees.copy()

    sorted\_fees.sort(reverse=True)

    for i in range(len(copy\_fees)):

        idx = copy\_fees.index(sorted\_fees[i])

        sorted\_names.append(user\_names[idx])

        copy\_fees.pop(idx)

def display\_ordered\_residents():

    calculate\_fees()

    sort()

    print("\nResidents ordered w.r.t fees: ")

    print("--------- -------- -------- -------- -------- -------- ")

    print("Username\t|\tFees")

    print("--------- -------- -------- -------- -------- -------- ")

    for name, fee in zip(sorted\_names, sorted\_fees):

        print("\t\t", name, "\t\t|\t", fee)

    print("--------- -------- -------- -------- -------- -------- ")

    wait("")

def check\_status():

    name = input("\nEnter your name: ")

    if name in user\_names or name in staff\_names:

        print("Your application has been accepted!")

    else:

        print("Your application has not been accepted yet...")

def load\_demo\_data():

    store\_user("resident", "safee", "s123")

    store\_user("resident", "adeel", "a123")

    store\_user("resident", "hassan", "h123")

    store\_user("resident", "javad", "j123")

    store\_user("resident", "moeez", "m123")

    store\_user("resident", "umair", "u123")

    store\_user("resident", "mubshir", "m123")

    store\_attendance(2, 0)

    store\_attendance(34, 1)

    store\_attendance(12, 2)

    store\_attendance(4, 3)

    store\_attendance(5, 4)

    store\_attendance(20, 5)

    store\_attendance(14, 6)

    store\_user("staff", "adeel", "a123")

    store\_user("staff", "bilal", "b123")

    store\_user("staff", "nawaz", "n123")

    store\_user("staff", "zardari", "z123")

    store\_notice(

        "Welcome to the hostel. please read the notice carefully. It contains inportant information, which might be useful to you.")

    store\_notice(

        "please read the notice carefully. Welcome to the hostel. It contains inportant information, which might be useful to you.")

    store\_notice(

        "Welcome to the hostel. It contains inportant information, which might be useful to you. please read the notice carefully.")

main()

# Section 6:

**Improvement from previous phase:**

The problem of invalid input is solved. If the user enters invalid input the program won’t proceed any further and ask the user to re-enter the input instead of crashing or giving error.

**Limitation Faced in Phase 5:**

The limitation we now is that we cannot store data. Instead the program needs to run continuously to keep the data alive. Since the data is stored in RAM and not Hard Drive, It resets every time the program is terminated. We need to store permanently instead of temporarily storing it in RAM.

**Code:**

import os

staff\_names = []

staff\_passwords = []

user\_names = []

user\_passwords = []

user\_atts = []

notices = []

applications = []

fees = []

expenses = 0

sorted\_indexes = []

sorted\_fees = []

sorted\_names = []

admin\_pass = "1234"

def main():

    load\_demo\_data()

    while True:

        os.system("cls")

        display\_header()

        option = display\_menu(get\_menu\_items("main"))

        if option == '1':

            login\_option = '-1'

            while login\_option != '4':

                os.system("cls")

                display\_header()

                login\_option = display\_menu(get\_menu\_items("login"))

                if login\_option == '1':  # Admin

                    pass\_check = input("\n\nEnter Password: ")

                    if pass\_check == admin\_pass:

                        admin\_option = '-1'

                        while admin\_option != '12':  # Admin menu loop

                            os.system("cls")

                            display\_header()

                            admin\_option = display\_menu(

                                get\_menu\_items("admin"))

                            if admin\_option == '1':  # Adding User

                                add\_option = display\_menu(

                                    get\_menu\_items("add"))

                                if add\_option == '1':

                                    add\_staff\_ui()

                                    wait("")

                                elif add\_option == '2':

                                    add\_resident\_ui()

                                    wait("")

                                elif add\_option == '3':

                                    wait("")

                            elif admin\_option == '2':  # Removing User

                                removing\_option = display\_menu(

                                    get\_menu\_items("remove"))

                                if removing\_option == '1':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("staff", name)

                                    wait("")

                                elif removing\_option == '2':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("resident", name)

                                    wait("")

                                elif removing\_option == '3':

                                    wait("")

                            elif admin\_option == '3':  # View all apllications

                                view\_applications()

                                wait("")

                            elif admin\_option == '4':  # Remove Application

                                remove\_application()

                                wait("")

                            elif admin\_option == '5':  # Add expenses

                                expenses = int(input("\n\t\tEnter expenses: "))

                                if expenses >= 0 and expenses <= 100000:

                                    print("This month's expenses are: ", expenses)

                                else:

                                    print("Invalid amount entered. Please retry")

                                wait("")

                            elif admin\_option == '6':  # Calculate expenses

                                calculate\_fees()

                                wait("")

                            elif admin\_option == '7':  # View expenses

                                print("This month's expenses are: ", expenses)

                                wait("")

                            elif admin\_option == '8':  # Make a notice on the notice board

                                add\_notice\_ui()

                                wait("")

                            elif admin\_option == '9':  # Display notices

                                show\_notices()

                                wait("")

                            elif admin\_option == '10':  # View all members

                                display\_members(

                                    "Staff", staff\_names, staff\_passwords)

                                display\_members(

                                    "Residents", user\_names, user\_passwords)

                                wait("")

                            elif admin\_option == '11':  # View resident fees

                                calculate\_fees()

                                sort()

                                display\_ordered\_residents()

                                wait("")

                            elif admin\_option == '12':  # Logout

                                wait("")

                    else:

                        print("\nInvalid Password!!")

                        wait("")

                elif login\_option == '2':  # Resident

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(user\_names, user\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            resident\_option = '-1'

                            user\_idx = user\_names.index(name)

                            resident\_option = display\_menu(

                                get\_menu\_items("resident"))

                            if resident\_option == '1':  # Bill for current month

                                calculate\_fees()

                                print(

                                    "\nYour Bill for the Current Month is: ", fees[user\_idx])

                                wait("")

                            elif resident\_option == 2:  # Attendance for the Current Month

                                print(

                                    "\nYour Attendance for the Current Month is: ", user\_atts[user\_idx])

                                wait("")

                            elif resident\_option == 3:  # Logout

                                logi = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == 3:  # Staff

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(staff\_names, staff\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            staff\_option = '-1'

                            user\_idx = staff\_names.index(name)

                            staff\_option = display\_menu(

                                get\_menu\_items("staff"))

                            if staff\_option == '1':  # Display members

                                display\_members(

                                    "residents", user\_names, user\_passwords)

                                wait("")

                            elif staff\_option == '2':  # Add attendance

                                add\_attendance\_ui()

                                wait("")

                            elif staff\_option == '3':  # logout

                                login = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == '4':  # go back

                    wait("")

        elif option == '2':  # View Notice Board

            show\_notices()

            wait("")

        elif option == '3':  # Apply for Allotment

            make\_application()

            wait("")

        elif option == 4:  # View application status

            check\_status()

            wait("")

        elif option == 5:  # Close program

            print("\n\t\tThank you for using the software,")

            wait("")

def display\_header():

    header = ["                          \_\_  \_\_           \_\_       \_\_",

              "                         / / / /\_\_\_  \_\_\_\_\_/ /\_\_\_\_  / /",

              "                        / /\_/ / \_\_ \\/ \_\_\_/ \_\_/ \_ \\/ /",

              "                       / \_\_  / /\_/ (\_\_  ) /\_/  \_\_/ /",

              "                      /\_/ /\_/\\\_\_\_\_/\_\_\_\_/\\\_\_/\\\_\_\_/\_/",

              "           \_\_  \_\_\_                                                  \_\_",

              "          /  |/  /\_\_\_ \_\_\_\_\_  \_\_\_\_ \_\_\_\_\_ \_\_\_\_  \_\_\_\_ \_\_\_  \_\_\_  \_\_\_\_  / /\_",

              "         / /|\_/ / \_\_ `/ \_\_ \\/ \_\_ `/ \_\_ `/ \_ \\/ \_\_ `\_\_ \\/ \_ \\/ \_\_ \\/ \_\_/",

              "        / /  / / /\_/ / / / / /\_/ / /\_/ /  \_\_/ / / / / /  \_\_/ / / / /\_",

              "       /\_/  /\_/\\\_\_,\_/\_/ /\_/\\\_\_,\_/\\\_\_, /\\\_\_\_/\_/ /\_/ /\_/\\\_\_\_/\_/ /\_/\\\_\_/",

              "                                /\_\_\_\_/",

              "                      \_\_\_\_\_            \_\_",

              "                     / \_\_\_/\_\_  \_\_\_\_\_\_\_/ /\_\_\_\_  \_\_\_\_ \_\_\_",

              "                     \\\_\_ \\/ / / / \_\_\_/ \_\_/ \_ \\/ \_\_ `\_\_ \\  ",

              "                    \_\_\_/ / /\_/ (\_\_  ) /\_/  \_\_/ / / / / /",

              "                   /\_\_\_\_/\\\_\_, /\_\_\_\_/\\\_\_/\\\_\_\_/\_/ /\_/ /\_/",

              "                        /\_\_\_\_/ "]

    for line in header:

        print(line)

def display\_menu(menu):

    for item in menu:

        print(item)

    selection = input("\nYour Choice: ")

    return selection

def get\_menu\_items(menu\_name):

    menu\_names = ["main", "login", "admin",

                  "resident", "staff", "add", "remove"]

    menus = [[

        "\n\t\tMain Menu > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------",

        "1. Login",

        "2. View Notice Board",

        "3. Apply for Allotment/Employment",

        "4. View status",

        "5. Exit"

    ], [

        "\n\t\tMain Menu > Login > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tLogin as:\n",

        "1. Administrator",

        "2. Resident",

        "3. Staff",

        "4. Go back"

    ], [

        "\n\t\tMain Menu > Login > Admin > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add a user",

        "2. Remove a user",

        "3. View applications",

        "4. Remove applications",

        "5. Add expenses",

        "6. Calculate expenses",

        "7. View this month\'s expenses",

        "8. Make a notice on the notice board",

        "9. View Notice board",

        "10. View all members",

        "11. View resident fees",

        "12. Logout"

    ], [

        "\n\t\tMain Menu > Login > Resident > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View this month\'s bill",

        "2. View attendance",

        "3. logout"

    ], [

        "\n\t\tMain Menu > Login > Staff > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View all Residents",

        "2. Add Attendance",

        "3. Logout"

    ], [

        "\n\t\tMain Menu > Login > Admin > Add User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add Staff",

        "2. Add Resident",

        "3. Go Back"

    ], [

        "\n\t\tMain Menu > Login > Staff > Remove User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Remove Staff",

        "2. Remove Resident",

        "3. Go Back"

    ]]

    if menu\_name in menu\_names:

        return menus[menu\_names.index(menu\_name)]

def store\_user(category, user\_name, password):

    if category == "staff":

        staff\_names.append(user\_name)

        staff\_passwords.append(password)

    elif category == "resident":

        user\_names.append(user\_name)

        user\_passwords.append(password)

        user\_atts.append(0)

def add\_resident\_ui():

    validating = True

    while validating:

        name = input("Enter resident name: ")

        if not name.isalnum() and not name.isdigit():

            validating = False

        else:

            print("Invalid name entered.")

        password = input("Enter resident password: ")

    store\_user("resident", name, password)

def add\_staff\_ui():

    validating = True

    while validating:

        name = input("Enter staff name: ")

        if not name.isalnum() and not name.isdigit():

            validating = False

        else:

            print("Invalid name entered.")

        password = input("Enter staff password: ")

    store\_user("staff", name, password)

def remove\_member(category, name):

    found = False

    if category == "resident":

        if name in user\_names:

            idx = user\_names.index(name)

            user\_names.pop(idx)

            user\_passwords.pop(idx)

            found = True

    elif category == "staff":

        if name in staff\_names:

            idx = staff\_names.index(name)

            staff\_names.pop(idx)

            staff\_passwords.pop(idx)

            found = True

    if found:

        print(name, "has been removed from", category)

    else:

        print("Couldn't find", name, "in", category)

def view\_applications():

    for application in applications:

        print(application)

        print("---------------------------------------------------------------------------")

def remove\_application():

    app\_no = int(

        input("Enter application number you want to be removed: ")) - 1

    if app\_no < len(applications):

        applications.pop(app\_no)

    else:

        print("Cannot find the application")

def display\_members(category, users, passwords):

    print(category + ":")

    print("------- -------- -------- -------- -------- -------- -------- ---------")

    print("Username\t|\tPassword", end='')

    if category == "Residents":

        print("\t|\tAttendance", end='')

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

    for i in range(len(users)):

        print(users[i] , "\t\t|\t", passwords[i], end='')

        if category == "Residents":

            print("\t\t|\t", user\_atts[i], end='')

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

def add\_notice\_ui():

    notice = input("\nEnter Notice: ")

    store\_notice(notice)

def store\_notice(notice):

    notices.append(notice)

def show\_notices():

    for notice in notices:

        print(notice)

def login(names, passwords, name, password):

    if name in names:  # find name

        idx = names.index(name)

        if passwords[idx] == password:  # check password

            return True

    return False

def add\_attendance\_ui():

    validating = False

    while validating:

        name = input("Enter user name: ")

        if name in user\_names:

            idx = user\_names.index(name)

            att = int(input("Enter attendance: "))

            if att > 0 and att < 90:

                validating = False

                store\_attendance(att, idx)

def store\_attendance(attendance, index):

    user\_atts[index] = attendance

def make\_application():

    applications.append(input(

        "Please enter your name\n and the password you want to use\nand the category you want to apply for\n(seperate them with \'\_\'):\n\nInput: "))

def wait(optional\_string):

    input(optional\_string + "\nPress ENTER key to continue...")

def calculate\_fees():

    global fees

    fees = []

    for att in user\_atts:

        fees.append(att \* 80)

def sort():

    global sorted\_indexes

    global sorted\_fees

    global sorted\_names

    sorted\_indexes = []

    copy\_fees = fees.copy()

    sorted\_fees = copy\_fees.copy()

    sorted\_fees.sort(reverse=True)

    for i in range(len(copy\_fees)):

        idx = copy\_fees.index(sorted\_fees[i])

        sorted\_names.append(user\_names[idx])

        copy\_fees.pop(idx)

def display\_ordered\_residents():

    calculate\_fees()

    sort()

    print("\nResidents ordered w.r.t fees: ")

    print("--------- -------- -------- -------- -------- -------- ")

    print("Username\t|\tFees")

    print("--------- -------- -------- -------- -------- -------- ")

    for name, fee in zip(sorted\_names, sorted\_fees):

        print("\t\t", name, "\t\t|\t", fee)

    print("--------- -------- -------- -------- -------- -------- ")

    wait("")

def check\_status():

    name = input("\nEnter your name: ")

    if name in user\_names or name in staff\_names:

        print("Your application has been accepted!")

    else:

        print("Your application has not been accepted yet...")

def load\_demo\_data():

    store\_user("resident", "safee", "s123")

    store\_user("resident", "adeel", "a123")

    store\_user("resident", "hassan", "h123")

    store\_user("resident", "javad", "j123")

    store\_user("resident", "moeez", "m123")

    store\_user("resident", "umair", "u123")

    store\_user("resident", "mubshir", "m123")

    store\_attendance(2, 0)

    store\_attendance(34, 1)

    store\_attendance(12, 2)

    store\_attendance(4, 3)

    store\_attendance(5, 4)

    store\_attendance(20, 5)

    store\_attendance(14, 6)

    store\_user("staff", "adeel", "a123")

    store\_user("staff", "bilal", "b123")

    store\_user("staff", "nawaz", "n123")

    store\_user("staff", "zardari", "z123")

    store\_notice(

        "Welcome to the hostel. please read the notice carefully. It contains inportant information, which might be useful to you.")

    store\_notice(

        "please read the notice carefully. Welcome to the hostel. It contains inportant information, which might be useful to you.")

    store\_notice(

        "Welcome to the hostel. It contains inportant information, which might be useful to you. please read the notice carefully.")

main()

# Section 7:

**Improvement from previous phase:**

After adding file system our problems are pretty much solved at this point. We don’t have to face the problem of data being reset every time and now we can store data permanently.

**Limitation Faced in Phase 6:**

The data is related to each other but stored separately. Even though we can take advantage of it to categorize and group it together into one entity, we cannot do that in Procedural programming.

We know that in real life this data belongs to a single entity but we have to deal with it like completely unrelated data. This is one of the main drawbacks of procedural programming.

If we want to use a function that takes many inputs that are related to the same entity. We have to take them all separately. It causes redundancy, makes the program more prone to errors and makes it less maintainable. If we want to make a single change in the function we have to modify each and every instance where we used it. This problem is manageable in small projects but it becomes impossibly difficult in big projects with thousands or hundreds of thousands of lines of codes.

It is hard to extend our program. We cannot divide our program into different parts that each deal with a separate functionality of the program. We have to write everything in a single file and it becomes very complex for bigger projects.

**Code:**

import os

staff\_names = []

staff\_passwords = []

user\_names = []

user\_passwords = []

user\_atts = []

notices = []

applications = []

fees = []

expenses = 0

sorted\_indexes = []

sorted\_fees = []

sorted\_names = []

admin\_pass = "1234"

def main():

    read\_notices()

    read\_staff()

    read\_users()

    while True:

        os.system("cls")

        display\_header()

        option = display\_menu(get\_menu\_items("main"))

        if option == '1':

            login\_option = '-1'

            while login\_option != '4':

                os.system("cls")

                display\_header()

                login\_option = display\_menu(get\_menu\_items("login"))

                if login\_option == '1':  # Admin

                    pass\_check = input("\n\nEnter Password: ")

                    if pass\_check == admin\_pass:

                        admin\_option = '-1'

                        while admin\_option != '12':  # Admin menu loop

                            os.system("cls")

                            display\_header()

                            admin\_option = display\_menu(

                                get\_menu\_items("admin"))

                            if admin\_option == '1':  # Adding User

                                add\_option = display\_menu(

                                    get\_menu\_items("add"))

                                if add\_option == '1':

                                    add\_staff\_ui()

                                    wait("")

                                elif add\_option == '2':

                                    add\_resident\_ui()

                                    wait("")

                                elif add\_option == '3':

                                    wait("")

                            elif admin\_option == '2':  # Removing User

                                removing\_option = display\_menu(

                                    get\_menu\_items("remove"))

                                if removing\_option == '1':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("staff", name)

                                    wait("")

                                elif removing\_option == '2':

                                    name = input("\n\nEnter User Name: ")

                                    remove\_member("resident", name)

                                    wait("")

                                elif removing\_option == '3':

                                    wait("")

                            elif admin\_option == '3':  # View all apllications

                                view\_applications()

                                wait("")

                            elif admin\_option == '4':  # Remove Application

                                remove\_application()

                                wait("")

                            elif admin\_option == '5':  # Add expenses

                                expenses = int(input("\n\t\tEnter expenses: "))

                                if expenses >= 0 and expenses <= 100000:

                                    print("This month's expenses are: ", expenses)

                                else:

                                    print("Invalid amount entered. Please retry")

                                wait("")

                            elif admin\_option == '6':  # Calculate expenses

                                calculate\_fees()

                                wait("")

                            elif admin\_option == '7':  # View expenses

                                print("This month's expenses are: ", expenses)

                                wait("")

                            elif admin\_option == '8':  # Make a notice on the notice board

                                add\_notice\_ui()

                                wait("")

                            elif admin\_option == '9':  # Display notices

                                show\_notices()

                                wait("")

                            elif admin\_option == '10':  # View all members

                                display\_members(

                                    "Staff", staff\_names, staff\_passwords)

                                display\_members(

                                    "Residents", user\_names, user\_passwords)

                                wait("")

                            elif admin\_option == '11':  # View resident fees

                                calculate\_fees()

                                sort()

                                display\_ordered\_residents()

                                wait("")

                            elif admin\_option == '12':  # Logout

                                wait("")

                    else:

                        print("\nInvalid Password!!")

                        wait("")

                elif login\_option == '2':  # Resident

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(user\_names, user\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            resident\_option = '-1'

                            user\_idx = user\_names.index(name)

                            resident\_option = display\_menu(

                                get\_menu\_items("resident"))

                            if resident\_option == '1':  # Bill for current month

                                calculate\_fees()

                                print(

                                    "\nYour Bill for the Current Month is: ", fees[user\_idx])

                                wait("")

                            elif resident\_option == 2:  # Attendance for the Current Month

                                print(

                                    "\nYour Attendance for the Current Month is: ", user\_atts[user\_idx])

                                wait("")

                            elif resident\_option == 3:  # Logout

                                logi = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == 3:  # Staff

                    name = input("Enter username: ")

                    password = input("Enter password: ")

                    logi = login(staff\_names, staff\_passwords, name, password)

                    if logi:

                        while logi:

                            os.system("cls")

                            display\_header()

                            staff\_option = '-1'

                            user\_idx = staff\_names.index(name)

                            staff\_option = display\_menu(

                                get\_menu\_items("staff"))

                            if staff\_option == '1':  # Display members

                                display\_members(

                                    "residents", user\_names, user\_passwords)

                                wait("")

                            elif staff\_option == '2':  # Add attendance

                                add\_attendance\_ui()

                                wait("")

                            elif staff\_option == '3':  # logout

                                login = False

                                wait("")

                    else:

                        wait("")

                elif login\_option == '4':  # go back

                    wait("")

        elif option == '2':  # View Notice Board

            show\_notices()

            wait("")

        elif option == '3':  # Apply for Allotment

            make\_application()

            wait("")

        elif option == 4:  # View application status

            check\_status()

            wait("")

        elif option == 5:  # Close program

            print("\n\t\tThank you for using the software,")

            wait("")

def display\_header():

    header = ["                          \_\_  \_\_           \_\_       \_\_",

              "                         / / / /\_\_\_  \_\_\_\_\_/ /\_\_\_\_  / /",

              "                        / /\_/ / \_\_ \\/ \_\_\_/ \_\_/ \_ \\/ /",

              "                       / \_\_  / /\_/ (\_\_  ) /\_/  \_\_/ /",

              "                      /\_/ /\_/\\\_\_\_\_/\_\_\_\_/\\\_\_/\\\_\_\_/\_/",

              "           \_\_  \_\_\_                                                  \_\_",

              "          /  |/  /\_\_\_ \_\_\_\_\_  \_\_\_\_ \_\_\_\_\_ \_\_\_\_  \_\_\_\_ \_\_\_  \_\_\_  \_\_\_\_  / /\_",

              "         / /|\_/ / \_\_ `/ \_\_ \\/ \_\_ `/ \_\_ `/ \_ \\/ \_\_ `\_\_ \\/ \_ \\/ \_\_ \\/ \_\_/",

              "        / /  / / /\_/ / / / / /\_/ / /\_/ /  \_\_/ / / / / /  \_\_/ / / / /\_",

              "       /\_/  /\_/\\\_\_,\_/\_/ /\_/\\\_\_,\_/\\\_\_, /\\\_\_\_/\_/ /\_/ /\_/\\\_\_\_/\_/ /\_/\\\_\_/",

              "                                /\_\_\_\_/",

              "                      \_\_\_\_\_            \_\_",

              "                     / \_\_\_/\_\_  \_\_\_\_\_\_\_/ /\_\_\_\_  \_\_\_\_ \_\_\_",

              "                     \\\_\_ \\/ / / / \_\_\_/ \_\_/ \_ \\/ \_\_ `\_\_ \\  ",

              "                    \_\_\_/ / /\_/ (\_\_  ) /\_/  \_\_/ / / / / /",

              "                   /\_\_\_\_/\\\_\_, /\_\_\_\_/\\\_\_/\\\_\_\_/\_/ /\_/ /\_/",

              "                        /\_\_\_\_/ "]

    for line in header:

        print(line)

def display\_menu(menu):

    for item in menu:

        print(item)

    selection = input("\nYour Choice: ")

    return selection

def get\_menu\_items(menu\_name):

    menu\_names = ["main", "login", "admin",

                  "resident", "staff", "add", "remove"]

    menus = [[

        "\n\t\tMain Menu > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------",

        "1. Login",

        "2. View Notice Board",

        "3. Apply for Allotment/Employment",

        "4. View status",

        "5. Exit"

    ], [

        "\n\t\tMain Menu > Login > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tLogin as:\n",

        "1. Administrator",

        "2. Resident",

        "3. Staff",

        "4. Go back"

    ], [

        "\n\t\tMain Menu > Login > Admin > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add a user",

        "2. Remove a user",

        "3. View applications",

        "4. Remove applications",

        "5. Add expenses",

        "6. Calculate expenses",

        "7. View this month\'s expenses",

        "8. Make a notice on the notice board",

        "9. View Notice board",

        "10. View all members",

        "11. View resident fees",

        "12. Logout"

    ], [

        "\n\t\tMain Menu > Login > Resident > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View this month\'s bill",

        "2. View attendance",

        "3. logout"

    ], [

        "\n\t\tMain Menu > Login > Staff > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. View all Residents",

        "2. Add Attendance",

        "3. Logout"

    ], [

        "\n\t\tMain Menu > Login > Admin > Add User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Add Staff",

        "2. Add Resident",

        "3. Go Back"

    ], [

        "\n\t\tMain Menu > Login > Staff > Remove User > ",

        "\n\t--------- --------- -------- -------- -------- -------- -------- -------- --------- --------\n\t\tChoose option:\n",

        "1. Remove Staff",

        "2. Remove Resident",

        "3. Go Back"

    ]]

    if menu\_name in menu\_names:

        return menus[menu\_names.index(menu\_name)]

def store\_user(category, user\_name, password):

    read\_staff()

    read\_users()

    if category == "staff":

        staff\_names.append(user\_name)

        staff\_passwords.append(password)

    elif category == "resident":

        user\_names.append(user\_name)

        user\_passwords.append(password)

        user\_atts.append(0)

    save\_users(user\_names, user\_passwords, user\_atts)

    save\_staff(user\_names, user\_passwords)

def add\_resident\_ui():

    validating = True

    while validating:

        name = input("Enter resident name: ")

        if not name.isalnum() and not name.isdigit():

            validating = False

        else:

            print("Invalid name entered.")

        password = input("Enter resident password: ")

    store\_user("resident", name, password)

def add\_staff\_ui():

    validating = True

    while validating:

        name = input("Enter staff name: ")

        if not name.isalnum() and not name.isdigit():

            validating = False

        else:

            print("Invalid name entered.")

        password = input("Enter staff password: ")

    store\_user("staff", name, password)

def remove\_member(category, name):

    read\_users()

    read\_staff()

    found = False

    if category == "resident":

        if name in user\_names:

            idx = user\_names.index(name)

            user\_names.pop(idx)

            user\_passwords.pop(idx)

            found = True

    elif category == "staff":

        if name in staff\_names:

            idx = staff\_names.index(name)

            staff\_names.pop(idx)

            staff\_passwords.pop(idx)

            found = True

    if found:

        print(name, "has been removed from", category)

    else:

        print("Couldn't find", name, "in", category)

    save\_users(user\_names, user\_passwords, user\_atts)

    save\_staff(user\_names, user\_passwords)

def view\_applications():

    for application in applications:

        print(application)

        print("---------------------------------------------------------------------------")

def remove\_application():

    app\_no = int(

        input("Enter application number you want to be removed: ")) - 1

    if app\_no < len(applications):

        applications.pop(app\_no)

    else:

        print("Cannot find the application")

def display\_members(category, users, passwords):

    read\_users()

    print(category + ":")

    print("------- -------- -------- -------- -------- -------- -------- ---------")

    print("Username\t|\tPassword", end='')

    if category == "Residents":

        print("\t|\tAttendance", end='')

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

    for i in range(len(users)):

        print(users[i] , "\t\t|\t", passwords[i], end='')

        if category == "Residents":

            print("\t\t|\t", user\_atts[i])

        else:

            print(" ")

    print("\n--------- -------- -------- -------- -------- -------- -------- ---------")

def add\_notice\_ui():

    notice = input("\nEnter Notice: ")

    store\_notice(notice)

    save\_notices()

def store\_notice(notice):

    notices.append(notice)

def show\_notices():

    for notice in notices:

        print(notice)

def login(names, passwords, name, password):

    read\_users()

    if name in names:  # find name

        idx = names.index(name)

        if passwords[idx] == password:  # check password

            return True

    return False

def add\_attendance\_ui():

    read\_users()

    validating = False

    while validating:

        name = input("Enter user name: ")

        if name in user\_names:

            idx = user\_names.index(name)

            att = int(input("Enter attendance: "))

            if att > 0 and att < 90:

                validating = False

                store\_attendance(att, idx)

    save\_users(user\_names, user\_passwords, user\_atts)

def store\_attendance(attendance, index):

    user\_atts[index] = attendance

def make\_application():

    applications.append(input(

        "Please enter your name\n and the password you want to use\nand the category you want to apply for\n(seperate them with \'\_\'):\n\nInput: "))

def wait(optional\_string):

    input(optional\_string + "\nPress ENTER key to continue...")

def calculate\_fees():

    read\_users()

    global fees

    fees = []

    for att in user\_atts:

        fees.append(int(att) \* 80)

def sort():

    read\_users()

    global sorted\_indexes

    global sorted\_fees

    global sorted\_names

    sorted\_indexes = []

    copy\_fees = fees.copy()

    sorted\_fees = copy\_fees.copy()

    sorted\_fees.sort(reverse=True)

    for i in range(len(copy\_fees)):

        idx = copy\_fees.index(sorted\_fees[i])

        sorted\_names.append(user\_names[idx])

        copy\_fees.pop(idx)

def display\_ordered\_residents():

    read\_users()

    calculate\_fees()

    sort()

    print("\nResidents ordered w.r.t fees: ")

    print("--------- -------- -------- -------- -------- -------- ")

    print("Username\t|\tFees")

    print("--------- -------- -------- -------- -------- -------- ")

    for name, fee in zip(sorted\_names, sorted\_fees):

        print("\t\t", name, "\t\t|\t", fee, end='\n')

    print("--------- -------- -------- -------- -------- -------- ")

    wait("")

def check\_status():

    read\_users()

    name = input("\nEnter your name: ")

    if name in user\_names or name in staff\_names:

        print("Your application has been accepted!")

    else:

        print("Your application has not been accepted yet...")

def load\_demo\_data():

    store\_user("resident", "safee", "s123")

    store\_user("resident", "adeel", "a123")

    store\_user("resident", "hassan", "h123")

    store\_user("resident", "javad", "j123")

    store\_user("resident", "moeez", "m123")

    store\_user("resident", "umair", "u123")

    store\_user("resident", "mubshir", "m123")

    store\_attendance(2, 0)

    store\_attendance(34, 1)

    store\_attendance(12, 2)

    store\_attendance(4, 3)

    store\_attendance(5, 4)

    store\_attendance(20, 5)

    store\_attendance(14, 6)

    store\_user("staff", "adeel", "a123")

    store\_user("staff", "bilal", "b123")

    store\_user("staff", "nawaz", "n123")

    store\_user("staff", "zardari", "z123")

    store\_notice(

        "Welcome to the hostel. please read the notice carefully. It contains inportant information, which might be useful to you.")

    store\_notice(

        "please read the notice carefully. Welcome to the hostel. It contains inportant information, which might be useful to you.")

    store\_notice(

        "Welcome to the hostel. It contains inportant information, which might be useful to you. please read the notice carefully.")

def save\_users(names, passwords, attendances):

    with open("residents.txt", 'w') as file:

        for name, password, attendance in zip(names, passwords, attendances):

            print(name, password, attendance, sep=',', file=file)

def save\_staff(names, passwords):

    with open("staff.txt", 'w') as file:

        for name, password in zip(names, passwords):

            print(name, password, sep=',', file=file)

def save\_notices(notices):

    with open("notices.txt", 'w') as file:

        for notice in notices:

            print(notice, file=file)

def read\_users():

    global user\_names

    global user\_passwords

    global user\_atts

    user\_names = []

    user\_passwords = []

    user\_atts = []

    with open("residents.txt", 'r') as file:

        for line in   file.readlines():

            if len(line.split(',')) ==  3:

                name, password, attendance = line.split(',')

                user\_names.append(name)

                user\_passwords.append(password)

                user\_atts.append(attendance.split('\n')[0])

def read\_staff():

    global staff\_names

    global staff\_passwords

    staff\_names = []

    staff\_passwords = []

    with open("staff.txt", 'r') as file:

        for line in   file.readlines():

            if len(line.split(',')) ==  2:

                name, password = line.split(',')

                staff\_names.append(name)

                staff\_passwords.append(password.split('\n')[0])

def read\_notices():

    global notices

    notices = []

    with open("notices.txt", 'r') as file:

        for line in file.readlines():

            if line != '':

                notices.append(line)

main()