MEDICARE

HOSPITAL DATABASE MANAGEMENT SYSTEM

BY:

NAME: SHELDON ASHISH STEPHEN

ROLL NUMBER: 20632056

CLASS AND SECTION: XII A

DON BOSCO SCHOOL OF EXCELLENCE
SENIOR SECONDARY SCHOOL

(AFFILIATED TO CBSE, DELHI, AFFILIATION CODE N0 – 1930554)
No. 53, Pantheon Road, Egmore, Chennai – 600 008.

BONAFIDE CERTIFICATE

Certified to be the Bonafide Record of Project work done by _____ of Class XII ____ in the Computer Laboratory of Don Bosco School of Excellence, Chennai, during the academic year _____

Register No.

Date : Teacher-in-charge

Submitted for All India Senior Secondary Practical Examination
in _____ held on _____
at _____.

Principal Internal Examiner External Examiner

CONTENTS

ACKNOWLEDGEMENT

We are elated in presenting our Computer Science Project. We express our sincere gratitude to our beloved **Principal Rev. Fr. Philip Louie** and the institution for providing us with the excellent laboratory facilities for the successful completion of this project. We are extremely grateful to our computer science teachers for their guidance and valuable suggestions. We also thank the lab assistant for their timely help. We thank our fellow classmates for the support and suggestions they gave during the course of the project.

AIM

Design a project that helps patients to easily find hospital beds in any branch of the hospital, helps to place orders for medicines and other services provided by the hospital. Also helps patients to place appointments easily. Also allows users to make donations to the hospital.

INTRODUCTION

With the increase in covid cases people are finding it difficult to go to a hospital and find beds to get admitted in hospitals. But we are going to make it easy for them to get better. By making an online hospital management system where patients can make appointments with doctors online and providing services and amenities for them to be delivered to them just with a click of a button from home.

FUNCTIONS INCLUDED

validateLogin(username, password)

mysqlpythonsignup()

mysqlpythonlogin()

adminhomepage()

admindonations()

adminappointments()

adminavailability()

adminpackages()

adminusers()

adminpackagescode()

addrecord()

updaterecord()

selectrecord()

deleterecord()

close()

adminusersscode()

admindonationcode()

adminappointmentscode()

adminavailabilitycode()

homepage()

donations()

appointments()

availability()

packages()

```
packagescode()
packagesinput()
donationcode()
mysqlpythondon()
appointmentscode()
mysqlpythonapp()
availabilitycode()
availability()
pack()
grid()
title()
geometry()
configure()
Label()
Button()
destroy()
connect()
is_connected()
cursor()
column()
heading()
insert()
delete()
focus()
format()
fetchall()
execute()
mainloop()
```

PROJECT CODE

```
#*************************************************
*************************************************
*************************************************
*********

# IMPORTS
#*************************************************
*************************************************
*************************************************
*********
from tkinter import *
from tkinter import ttk
import mysql.connector as sql
from functools import partial
from tkinter import messagebox
from PIL import ImageTk,Image
def validateLogin(username, password):
    username= username.get()
    password= password.get()
    print(username,password)
#*************************************************
*************************************************
*************************************************
*********
#*************************************************
*************************************************
*************************************************
*********



#================================================
================================================
```

```
=============================================
=========

# LOGIN CODE
#=============================================
=============================================
=============================================
=========

def mysqlpythonsignup():



mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from login"
    cursor.execute(st)
    data=list(cursor.fetchall())
    uname=username.get()
    passwd=password.get()
    if (uname,passwd) in data:
         messagebox.showinfo("showinfo",
"Username already exists!")


    else:
        #mysql python connector
```

```
mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into login
values('{}','{}')".format(username.get(),password
.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from login"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
        mycon.close()
        messagebox.showinfo("showinfo", "User
Created")



def mysqlpythonlogin():
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
```

```
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from login"
    cursor.execute(st)
    data=list(cursor.fetchall())
    uname=username.get()
    passwd=password.get()
    if (uname,passwd) in data:
        tkWindow.destroy()
        homepage()
    elif uname=="admin" and passwd=="123":
        tkWindow.destroy()
        adminhomepage()
    else:
        messagebox.showerror("showerror", "Error!
username and password does not exist!")
#============================================
============================================
============================================
=========
#============================================
============================================
============================================
=========
```

```
#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-------
#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------



# ADMIN CODE



#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
-------



def adminhomepage():

    global window
    window=Tk()
    #window
    window.title('Medicare Admin')
    #setting tkinter window size
    window.geometry("600x300")
```

```
    window.configure(bg='teal')


    #heading label
    headingLabel = Label(window, text="ADMIN",
fg='white',
bg='#3A3B3C',height='3',width='7',font=("Arial",
15)).place(x=270,y=5)
    #buttons
    btn1=Button(window,
text="USERS",command=adminusers, fg='white',
bg='#3A3B3C', height='2')
    btn1.place(x=95, y=100)


    btn2=Button(window, text="AVAILABILITY",
command= adminavailability,fg='white',
bg='#3A3B3C',height='2')
    btn2.place(x=145, y=100)


    btn3=Button(window,
text="PACKAGES",command=adminpackages
,fg='white', bg='#3A3B3C',height='2')
    btn3.place(x=233, y=100)


    btn4=Button(window,
text="APPOINMENTS",command= adminappointments,
fg='white', bg='#3A3B3C',height='2')
    btn4.place(x=307, y=100)


    btn5=Button(window, text="DONATIONS",command=
admindonations,fg='white',
bg='#3A3B3C',height='2')
    btn5.place(x=405, y=100)
```

```
    window.mainloop()


def admindonations():
    window.destroy()
    admindonationcode()

def adminappointments():
    window.destroy()
    adminappointmentscode()

def adminavailability():
    window.destroy()
    adminavailabilitycode()

def adminpackages():
    window.destroy()
    adminpackagescode()

def adminusers():
    window.destroy()
    adminusersscode()
#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
def adminpackagescode():
```

```
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from packages"
    cursor.execute(st)
    data=list(cursor.fetchall())
    for row in data:
        print(row)




    # window
    root= Tk()
    root.title("Medicare Admin")
    root.geometry("700x700")


    tree=ttk.Treeview(root)


    # Defining columns

tree['column']=("S.NO","Username","Name","Package
s","Quantity")
```

```
    # format our columns
    tree.column("#0",width=0,stretch=NO)

tree.column("S.NO",anchor=W,width=0,stretch=NO)
    tree.column("Username",anchor=W, width=120)
    tree.column("Name", anchor=W,width=140)
    tree.column("Packages",anchor=W,width=100)
    tree.column("Quantity",anchor=W,width=100)



    # create headings
    tree.heading("S.NO",text="S.NO",anchor=W)
    tree.heading("Username",text="Username",
anchor=W)

tree.heading("Name",text="Name",anchor=CENTER)
    tree.heading("Packages",text="Package
No",anchor=W)

tree.heading("Quantity",text="Quantity",anchor=W)


    # add data
    global count
    count=1
    for record in data:

tree.insert(parent='',index='end',text="",
values=(count,record[0],record[1],record[2],recor
d[3]))
```

```
    count= count+1
# pack to screen
tree.pack(pady=20)


addframe=Frame(root)
addframe.pack(pady=20)


n1=Label(addframe,text="Username")
n1.grid(row=0,column=0)

il=Label(addframe,text="Name")
il.grid(row=0,column=1)

tl=Label(addframe,text="Packages")
tl.grid(row=0,column=2)

tl=Label(addframe,text="Quantity")
tl.grid(row=0,column=3)

Usernamebox=Entry(addframe)
Usernamebox.grid(row=1,column=0)

Namebox=Entry(addframe)
Namebox.grid(row=1,column=1)

Packagesbox=Entry(addframe)
```

```
    Packagesbox.grid(row=1,column=2)


    Quantitybox=Entry(addframe)
    Quantitybox.grid(row=1,column=3)







    #add record
    def addrecord():
        global count

tree.insert(parent='',index='end',text="",iid=cou
nt,
values=(count,Usernamebox.get(),Namebox.get(),Pac
kagesbox.get(),Quantitybox.get()))


        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into packages
values('{}','{}','{}','{}')".format(Usernamebox.g
et(),Namebox.get(),Packagesbox.get(),Quantitybox.
get())
```

```
            cursor.execute(st)
            mycon.commit()
            st="select * from packages"
            cursor.execute(st)
            data=cursor.fetchall()
            for row in data:
                print(row)
            messagebox.showinfo("showinfo", "Booked
Successful")
            #clear the boxes
            Usernamebox.delete(0,END)
            Namebox.delete(0,END)
            Packagesbox.delete(0,END)
            Quantitybox.delete(0,END)

        # update record
        def updaterecord():
            #grab record number
            selected=tree.focus()

            #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
            if mycon.is_connected():
                print("Successfully Conected to
database")
            else:
                print("Access denied")
```

```
        cursor=mycon.cursor()
        st="update packages set name='{}',
packages='{}', quantity='{}' where
username='{}'".format(Namebox.get(),Packagesbox.g
et(),Quantitybox.get(),Usernamebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  packages"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)

        #save new data
        tree.item(selected,text="",values
=(count-
1,Usernamebox.get(),Namebox.get(),Packagesbox.get
(),Quantitybox.get()))

        #delete text in text boxes
        Usernamebox.delete(0,END)
        Namebox.delete(0,END)
        Packagesbox.delete(0,END)
        Quantitybox.delete(0,END)


    # select record
    def selectrecord():
        #clear entery boxes
        Usernamebox.delete(0,END)
```

```
            Namebox.delete(0,END)
            Packagesbox.delete(0,END)
            Quantitybox.delete(0,END)

            #grab record number
            selected=tree.focus()

            #grab record values
            values = tree.item(selected,'values')

            #temp.config(text=values[1])

            #output to entery boxes
            Usernamebox.insert(0,values[1])
            Namebox.insert(0,values[2])
            Packagesbox.insert(0,values[3])
            Quantitybox.insert(0,values[4])



        # delete record
        def deleterecord():
            #grab record number
            selected=tree.focus()

            #mysql part

    mycon=sql.connect(host="localhost",user="root",pa
    sswd="password", database="project")
```

```
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="delete from packages where
username='{}'".format(Usernamebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  packages"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)

        #selection of record to delete
        x=tree.selection()[0]
        tree.delete(x)

        #delete text in text boxes
        Usernamebox.delete(0,END)
        Namebox.delete(0,END)
        Packagesbox.delete(0,END)
        Quantitybox.delete(0,END)
```

```
    # buttons
    # add record
    addrecord= Button(root, text="Add
record",command=addrecord)
    addrecord.pack(pady=10)


    delrecord= Button(root, text="Delete
record",command=deleterecord)
    delrecord.pack(pady=10)


    selectrecord= Button(root, text="Select
record",command=selectrecord)
    selectrecord.pack(pady=10)


    updaterecord= Button(root, text="Update
record",command=updaterecord)
    updaterecord.pack(pady=10)


    #close
    def close():
        root.destroy()
        adminhomepage()


    close= Button(root,
text="close",command=close)
    close.pack(pady=10)
```

```
    temp=Label(root,text="")
    temp.pack (pady=10)



    root.mainloop()




#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
def adminusersscode():
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from login"
    cursor.execute(st)
    data=list(cursor.fetchall())
    for row in data:
        print(row)
```

```
    # window
    root= Tk()
    root.title("Medicare Admin")
    root.geometry("700x700")

    tree=ttk.Treeview(root)

    # Defining columns
    tree['column']=("S.NO","Username","Password")

    # format our columns
    tree.column("#0",width=0,stretch=NO)

tree.column("S.NO",anchor=W,width=0,stretch=NO)
    tree.column("Username",anchor=W, width=120)
    tree.column("Password", anchor=W,width=140)



    # create headings
    tree.heading("S.NO",text="S.NO",anchor=W)
    tree.heading("Username",text="Username",
anchor=W)

tree.heading("Password",text="Password",anchor=CE
NTER)
    # add data
    global count
    count=1
```

```
    for record in data:

tree.insert(parent='',index='end',text="",
values=(count,record[0],record[1]))
        count= count+1
    # pack to screen
    tree.pack(pady=20)



    addframe=Frame(root)
    addframe.pack(pady=20)



    n1=Label(addframe,text="Username")
    n1.grid(row=0,column=0)

    il=Label(addframe,text="Password")
    il.grid(row=0,column=1)



    Usernamebox=Entry(addframe)
    Usernamebox.grid(row=1,column=0)

    Passwordbox=Entry(addframe)
    Passwordbox.grid(row=1,column=1)
```

```
    #add record
    def addrecord():
        global count

tree.insert(parent='',index='end',text="",iid=cou
nt,
values=(count,Usernamebox.get(),Passwordbox.get()
))


        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into login
values('{}','{}')".format(Usernamebox.get(),Passw
ordbox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from login"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
```

```
        messagebox.showinfo("showinfo", "SignUp
Successful")
        #clear the boxes
        Usernamebox.delete(0,END)
        Passwordbox.delete(0,END)

    # update record
    def updaterecord():
        #grab record number
        selected=tree.focus()

        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="update login set password='{}' where
username='{}'".format(Passwordbox.get(),Usernameb
ox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  login"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
```

```
            print(row)

        #save new data
        tree.item(selected,text="",values
=(count-1,Usernamebox.get(),Passwordbox.get()))

        #delete text in text boxes
        Usernamebox.delete(0,END)
        Passwordbox.delete(0,END)



    # select record
    def selectrecord():
        #clear entery boxes
        Usernamebox.delete(0,END)
        Passwordbox.delete(0,END)

        #grab record number
        selected=tree.focus()

        #grab record values
        values = tree.item(selected,'values')

        #temp.config(text=values[1])

        #output to entery boxes
        Usernamebox.insert(0,values[1])
        Passwordbox.insert(0,values[2])
```

```
    # delete record
    def deleterecord():
        #grab record number
        selected=tree.focus()

        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="delete from login where
username='{}'".format(Usernamebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  login"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)

        #selection of record to delete
        x=tree.selection()[0]
```

```
        tree.delete(x)


        #delete text in text boxes
        Usernamebox.delete(0,END)
        Passwordbox.delete(0,END)




    # buttons
    # add record
    addrecord= Button(root, text="Add
record",command=addrecord)
    addrecord.pack(pady=10)


    delrecord= Button(root, text="Delete
record",command=deleterecord)
    delrecord.pack(pady=10)


    selectrecord= Button(root, text="Select
record",command=selectrecord)
    selectrecord.pack(pady=10)


    updaterecord= Button(root, text="Update
record",command=updaterecord)
    updaterecord.pack(pady=10)


    #close
    def close():
```

```
        root.destroy()
        adminhomepage()


    close= Button(root,
text="close",command=close)
    close.pack(pady=10)


    temp=Label(root,text="")
    temp.pack (pady=10)



    root.mainloop()




#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
def admindonationcode():
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
```

```
    st="select * from donations"
    cursor.execute(st)
    data=list(cursor.fetchall())
    for row in data:
        print(row)



    # window
    root= Tk()
    root.title("Medicare Admin")
    root.geometry("700x700")

    tree=ttk.Treeview(root)

    # Defining columns

tree['column']=("S.NO","Name","Address","Account
No","Amount")

    # format our columns
    tree.column("#0",width=0,stretch=NO)

tree.column("S.NO",anchor=W,width=0,stretch=NO)
    tree.column("Name",anchor=W, width=120)
    tree.column("Address",
anchor=CENTER,width=100)
    tree.column("Account No",anchor=W,width=140)
    tree.column("Amount",anchor=W,width=140)
```

```
    # create headings
    tree.heading("S.NO",text="S.NO",anchor=W)
    tree.heading("Name",text="Name", anchor=W)
tree.heading("Address",text="Address",anchor=CENT
ER)
    tree.heading("Account No",text="Account
No",anchor=W)
    tree.heading("Amount",text="Amount",anchor=W)

    # add data

    count=1
    for record in data:
tree.insert(parent='',index='end',text="",
values=(count,record[0],record[1],record[2],recor
d[3]))
        count= count+1
    # pack to screen
    tree.pack(pady=20)


    addframe=Frame(root)
    addframe.pack(pady=20)


    n1=Label(addframe,text="Name")
```

```
n1.grid(row=0,column=0)

il=Label(addframe,text="Address")
il.grid(row=0,column=1)

tl=Label(addframe,text="Account No")
tl.grid(row=0,column=2)

tl=Label(addframe,text="Amount")
tl.grid(row=0,column=3)

namebox=Entry(addframe)
namebox.grid(row=1,column=0)

addressbox=Entry(addframe)
addressbox.grid(row=1,column=1)

accnobox=Entry(addframe)
accnobox.grid(row=1,column=2)

amountbox=Entry(addframe)
amountbox.grid(row=1,column=3)




#add record
def addrecord():
```

```
        global count

tree.insert(parent='',index='end',text="",iid=cou
nt,
values=(count,namebox.get(),addressbox.get(),accn
obox.get(),amountbox.get()))


        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into donations
values('{}','{}','{}','{}')".format(namebox.get()
,addressbox.get(),accnobox.get(),amountbox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from donations"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
        messagebox.showinfo("showinfo", "Donation
Successful")
        #clear the boxes
        namebox.delete(0,END)
```

```
        addressbox.delete(0,END)
        accnobox.delete(0,END)
        amountbox.delete(0,END)



    # buttons
    # add record
    addrecord= Button(root, text="Add
record",command=addrecord)
    addrecord.pack(pady=20)


    #close
    def close():
        root.destroy()
        adminhomepage()


    close= Button(root,
text="close",command=close)
    close.pack(pady=10)



    root.mainloop()




#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
```

```
def adminappointmentscode():
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from appointments"
    cursor.execute(st)
    data=cursor.fetchall()
    for row in data:
        print(row)



    # window
    root= Tk()
    root.title("Medicare Admin")
    root.geometry("700x700")


    tree=ttk.Treeview(root)


    # Defining columns

tree['column']=("S.NO","UserName","Name","Age","G
ender","Symptoms")
```

```
    # format our columns
    tree.column("#0",width=0,stretch=NO)

tree.column("S.NO",anchor=W,width=0,stretch=NO)
    tree.column("UserName",anchor=W, width=120)
    tree.column("Name", anchor=CENTER,width=100)
    tree.column("Age",anchor=W,width=80)
    tree.column("Gender",anchor=W,width=140)
    tree.column("Symptoms",anchor=W,width=140)



    # create headings
    tree.heading("S.NO",text="S.NO",anchor=W)
    tree.heading("UserName",text="UserName",
anchor=W)

tree.heading("Name",text="Name",anchor=CENTER)
    tree.heading("Age",text="Age",anchor=W)
    tree.heading("Gender",text="Gender",anchor=W)

tree.heading("Symptoms",text="Symptoms",anchor=W)

    # add data
    global count
    count=1
    for record in data:

tree.insert(parent='',index='end',text="",
```

```
values=(count,record[0],record[1],record[2],recor
d[3],record[4]))
        count= count+1
    # pack to screen
    tree.pack(pady=20)


    addframe=Frame(root)
    addframe.pack(pady=20)


    n1=Label(addframe,text="UserName")
    n1.grid(row=0,column=0)

    il=Label(addframe,text="Name")
    il.grid(row=0,column=1)

    tl=Label(addframe,text="Age")
    tl.grid(row=0,column=2)

    ql=Label(addframe,text="Gender")
    ql.grid(row=0,column=3)

    wl=Label(addframe,text="Symptoms")
    wl.grid(row=0,column=4)

    UserNamebox=Entry(addframe)
    UserNamebox.grid(row=1,column=0)
```

```
    Namebox=Entry(addframe)
    Namebox.grid(row=1,column=1)

    Agebox=Entry(addframe)
    Agebox.grid(row=1,column=2)

    Genderbox=Entry(addframe)
    Genderbox.grid(row=1,column=3)

    Symptomsbox=Entry(addframe)
    Symptomsbox.grid(row=1,column=4)




    #add record
    def addrecord():
        global count

tree.insert(parent='',index='end',text="",iid=cou
nt, values=(count,UserNamebox.get(),
Namebox.get(),Agebox.get(),Genderbox.get(),Sympto
msbox.get()))


mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
```

```
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into appointments
values('{}','{}','{}','{}','{}')".format(UserName
box.get(),
Namebox.get(),Agebox.get(),Genderbox.get(),Sympto
msbox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  appointments"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
        #clear the boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Agebox.delete(0,END)
        Genderbox.delete(0,END)
        Symptomsbox.delete(0,END)

    # select record
    def selectrecord():
        #clear entery boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
```

```
        Agebox.delete(0,END)
        Genderbox.delete(0,END)
        Symptomsbox.delete(0,END)

        #grab record number
        selected=tree.focus()

        #grab record values
        values = tree.item(selected,'values')

        #temp.config(text=values[1])

        #output to entery boxes
        UserNamebox.insert(0,values[1])
        Namebox.insert(0,values[2])
        Agebox.insert(0,values[3])
        Genderbox.insert(0,values[4])
        Symptomsbox.insert(0,values[5])


    # update record
    def updaterecord():
        #grab record number
        selected=tree.focus()

        #mysql part
```

```
mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="update appointments set name='{}',
age='{}', gender='{}', problems='{}' where
username='{}'".format(Namebox.get(),Agebox.get(),
Genderbox.get(),Symptomsbox.get(),UserNamebox.get
())
        cursor.execute(st)
        mycon.commit()
        st="select * from  appointments"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)


        #save new data
        tree.item(selected,text="",values
=(count-1,UserNamebox.get(),
Namebox.get(),Agebox.get(),Genderbox.get(),Sympto
msbox.get()))


        #delete text in text boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
```

```
        Agebox.delete(0,END)
        Genderbox.delete(0,END)
        Symptomsbox.delete(0,END)


    # delete record
    def deleterecord():
        #grab record number
        selected=tree.focus()

        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="delete from appointments where
username='{}'".format(UserNamebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  appointments"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
```

```
        #selection of record to delete
        x=tree.selection()[0]
        tree.delete(x)

        #delete text in text boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Agebox.delete(0,END)
        Genderbox.delete(0,END)
        Symptomsbox.delete(0,END)




    # buttons
    # add record
    addrecord= Button(root, text="Add
record",command=addrecord)
    addrecord.pack(pady=10)


    delrecord= Button(root, text="Delete
record",command=deleterecord)
    delrecord.pack(pady=10)


    selectrecord= Button(root, text="Select
record",command=selectrecord)
```

```
    selectrecord.pack(pady=10)


    updaterecord= Button(root, text="Update
record",command=updaterecord)
    updaterecord.pack(pady=10)


    #close
    def close():
        root.destroy()
        adminhomepage()


    close= Button(root,
text="close",command=close)
    close.pack(pady=10)


    temp=Label(root,text="")
    temp.pack (pady=10)



    root.mainloop()









#-----------------------------------------------
-----------------------------------------------
```

```
-----------------------------------------------
--------


def adminavailabilitycode():
    #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
    if mycon.is_connected():
        print("Successfully Conected to
database")
    else:
        print("Access denied")
    cursor=mycon.cursor()
    st="select * from  availability"
    cursor.execute(st)
    data=cursor.fetchall()
    for row in data:
        print(row)



    # window
    root= Tk()
    root.title("Medicare Admin")
    root.geometry("700x700")

    tree=ttk.Treeview(root)

    # Defining columns
```

```
tree['column']=("S.NO","UserName","Name","Require
d_beds")

    # format our columns
    tree.column("#0",width=0,stretch=NO)

tree.column("S.NO",anchor=W,width=0,stretch=NO)
    tree.column("UserName",anchor=W, width=120)
    tree.column("Name", anchor=CENTER,width=100)

tree.column("Required_beds",anchor=W,width=140)




    # create headings
    tree.heading("S.NO",text="S.NO",anchor=W)
    tree.heading("UserName",text="UserName",
anchor=W)

tree.heading("Name",text="Name",anchor=CENTER)

tree.heading("Required_beds",text="Required_beds"
,anchor=W)



    # add data
    global count
    count=1
    for record in data:
```

```
tree.insert(parent='',index='end',text="",
values=(count,record[0],record[1],record[2]))
        count= count+1
    # pack to screen
    tree.pack(pady=20)


    addframe=Frame(root)
    addframe.pack(pady=20)


    n1=Label(addframe,text="UserName")
    n1.grid(row=0,column=0)

    il=Label(addframe,text="Name")
    il.grid(row=0,column=1)

    tl=Label(addframe,text="Required_beds")
    tl.grid(row=0,column=2)

    UserNamebox=Entry(addframe)
    UserNamebox.grid(row=1,column=0)

    Namebox=Entry(addframe)
    Namebox.grid(row=1,column=1)

    Required_bedsbox=Entry(addframe)
```

```
    Required_bedsbox.grid(row=1,column=2)




    #add record
    def addrecord():
        global count

tree.insert(parent='',index='end',text="",iid=cou
nt,
values=(count,UserNamebox.get(),Namebox.get(),Req
uired_bedsbox.get()))



mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into availability
values('{}','{}','{}')".format(UserNamebox.get(),
Namebox.get(),Required_bedsbox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  availability"
        cursor.execute(st)
        data=cursor.fetchall()
```

```
        for row in data:
            print(row)
        #clear the boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Required_bedsbox.delete(0,END)

    # select record
    def selectrecord():
        #clear entery boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Required_bedsbox.delete(0,END)

        #grab record number
        selected=tree.focus()
        #grab record values
        values = tree.item(selected,'values')

        #temp.config(text=values[1])

        #output to entery boxes
        UserNamebox.insert(0,values[1])
        Namebox.insert(0,values[2])
        Required_bedsbox.insert(0,values[3])



    # update record
```

```
    def updaterecord():
        #grab record number
        selected=tree.focus()

        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="update availability set name='{}',
beds='{}' where
username='{}'".format(Namebox.get(),Required_beds
box.get(),UserNamebox.get())

print(Namebox.get(),Required_bedsbox.get(),UserNa
mebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  availability"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)

        #save new data
```

```
        tree.item(selected,text="",values=(count-
1,UserNamebox.get(),Namebox.get(),Required_bedsbo
x.get()))


        #delete text in text boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Required_bedsbox.delete(0,END)




    # delete record
    def deleterecord():
        #grab record number
        selected=tree.focus()


        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="delete from availability where
username='{}'".format(UserNamebox.get())
```

```
print(Namebox.get(),Required_bedsbox.get(),UserNa
mebox.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  availability"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)

        #selection of record to delete
        x=tree.selection()[0]
        tree.delete(x)

        #delete text in text boxes
        UserNamebox.delete(0,END)
        Namebox.delete(0,END)
        Required_bedsbox.delete(0,END)









    # buttons
```

```
    # add record
    addrecord= Button(root, text="Add
record",command=addrecord)
    addrecord.pack(pady=10)


    delrecord= Button(root, text="Delete
record",command=deleterecord)
    delrecord.pack(pady=10)


    selectrecord= Button(root, text="Select
record",command=selectrecord)
    selectrecord.pack(pady=10)


    updaterecord= Button(root, text="Update
record",command=updaterecord)
    updaterecord.pack(pady=10)


    #close
    def close():
        root.destroy()
        adminhomepage()


    close= Button(root,
text="close",command=close)
    close.pack(pady=10)


    temp=Label(root,text="")
    temp.pack (pady=10)
```

```
    root.mainloop()


#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------

#-----------------------------------------------
-----------------------------------------------
-----------------------------------------------
--------
```

```
################################################
################################################
################################################
########
```

```
#PATIENT (CUSTOMER) PAGE  CODE
```

```
################################################
################################################
################################################
########
```

```
def homepage():
    global window
    window=Tk()
    #window
    window.title('Medicare')
    #getting screen width and height of display
    width= window.winfo_screenwidth()
    height= window.winfo_screenheight()
    #setting tkinter window size
    window.geometry("%dx%d" % (width, height))
    window.configure(bg='#008080')

    #logo
    global img1
    img1=
ImageTk.PhotoImage(Image.open('logo1.png'))
    panel = Label(window, image = img1)
    panel.place(x=5, y=5)

    #number
    img3=
ImageTk.PhotoImage(Image.open('num1.png'))
    panel = Label(window, image = img3)
    panel.place(x=1000, y=5)

    #background picture
```

```
    img2 =
ImageTk.PhotoImage(Image.open('backgroundpic1.png
'))

    panel = Label(window, image = img2)

    panel.place(x=0, y=150)


    #buttons

    btn1=Button(window, text="HOME", fg='white',
bg='#3A3B3C', height='2')

    btn1.place(x=95, y=100)


    btn2=Button(window,
text="AVAILABILITY",command=availability,
fg='white', bg='#3A3B3C',height='2')

    btn2.place(x=145, y=100)


    btn3=Button(window,
text="PACKAGES",command=packages,fg='white',
bg='#3A3B3C',height='2')

    btn3.place(x=233, y=100)


    btn4=Button(window,
text="APPOINMENTS",command=appointments,
fg='white', bg='#3A3B3C',height='2')

    btn4.place(x=307, y=100)


    btn5=Button(window, text="DONATIONS",command=
donations,fg='white', bg='#3A3B3C',height='2')

    btn5.place(x=405, y=100)
```

```
    window.mainloop()

def donations():

    window.destroy()
    donationcode()

def appointments():
    window.destroy()
    appointmentscode()

def availability():
    window.destroy()
    availabilitycode()

def packages():
    window.destroy()
    packagescode()



##################################################
##################################################
##################################################
########
def packagescode():
    #add record
    def packagesinput():

        #mysql part
```

```
mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into packages
values('{}','{}','{}','{}')".format(username.get(
),name.get(),entry.get(),quantity.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  packages"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
        messagebox.showinfo("showinfo", "Booking
Successful")
        tkWindow.destroy()


        mycon.close()


        homepage()




    #window
```

```
    tkWindow = Tk()
    tkWindow.geometry('500x400')
    tkWindow.title('Medicare')
    tkWindow.configure(bg='#008080')

    #username label and text entry box
    usernameLabel = Label(tkWindow, text="User
Name:").place(x=10,y=50)
    username = StringVar()
    usernameEntry = Entry(tkWindow,
textvariable=username).place(x=80,y=50)

    #name label and text entry box
    nameLabel = Label(tkWindow,
text="Name:").place(x=10,y=90)
    name = StringVar()
    nameEntry = Entry(tkWindow,
textvariable=name).place(x=80,y=90)

    #labels for packages
    headinglabel=Label(tkWindow, text="Please
choose from the menu below:").place(x=10,y=130)
    oneLabel = Label(tkWindow, text="1- Oxygen
Cylinders, Covid-19 kit, N95 Masks, Steam
Inhaler, Sanitizers").place(x=10,y=160)
    twoLabel = Label(tkWindow, text="2- Covid-19
kit, N95 Masks, Steam Inhaler").place(x=10,y=190)
    threeLabel = Label(tkWindow, text="3- N95
Masks, Sanitizers ").place(x=10,y=220)
```

```
    #entry label and text entry box
    entryLabsel = Label(tkWindow, text="Package
number: ").place(x=10,y=270)
    entry = StringVar()
    entryEntry = Entry(tkWindow,
textvariable=entry).place(x=120,y=270)


    #quantity label and text entry box
    quantityLabsel = Label(tkWindow,
text="Quantity: ").place(x=10,y=300)
    quantity  = StringVar()
    quantityEntry = Entry(tkWindow,
textvariable=quantity ).place(x=120,y=300)



    #Enter the command

button_submit=Button(tkWindow,text="Book",command
= packagesinput).place(x=200,y=330)


    #close
    def close():
        tkWindow.destroy()
        homepage()


    close= Button(tkWindow,
text="close",command=close)
    close.place(x=10,y=330)
```

```
    tkWindow.mainloop()


##################################################
##################################################
##################################################
########
def donationcode():


        def mysqlpythondon():
            #mysql python connector


mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
            if mycon.is_connected():
                print("Successfully Conected to
database")
            else:
                print("Access denied")
            cursor=mycon.cursor()
            st="insert into donations
values('{}','{}','{}','{}')".format(name.get(),ad
dress.get(),accountno.get(),amount.get())
            cursor.execute(st)
            mycon.commit()
            st="select * from donations"
            cursor.execute(st)
            data=cursor.fetchall()
            for row in data:
                print(row)
            messagebox.showinfo("showinfo",
"Donation Successful")
```

```
            tkWindow.destroy()


            mycon.close()


            homepage()


        #window
        tkWindow = Tk()
        tkWindow.geometry('500x350')
        tkWindow.title('Medicare')
        tkWindow.configure(bg='#008080')
        #getting screen width and height of
display
        #width= tkWindow.winfo_screenwidth()
        #height= tkWindow.winfo_screenheight()
        #setting tkinter window size
        #tkWindow.geometry("%dx%d" % (width,
height))


        # heading label
        nameLabel = Label(tkWindow, text=" Make
Donation").place(x=10,y=10)
        #name label and entry box
        nameLabel = Label(tkWindow,
text="Name").place(x=10,y=50)
        name = StringVar()
        nameEntry = Entry(tkWindow,
textvariable=name).place(x=80,y=50)
```

```
        #address label and entry box
        addressLabel =
Label(tkWindow,text="Address").place(x=10,y=100)
        address = StringVar()
        addressEntry = Entry(tkWindow,
textvariable=address).place(x=80,y=100)


        #accountno label and entry box
        accountnoLabel =
Label(tkWindow,text="Account
Number").place(x=10,y=150)
        accountno = StringVar()
        accountnoEntry = Entry(tkWindow,
textvariable=accountno).place(x=120,y=150)


        #amount label and entry box
        amountLabel = Label(tkWindow,text="Amount
").place(x=10,y=200)
        amount  = StringVar()
        amountEntry = Entry(tkWindow,
textvariable=amount ).place(x=80,y=200)


        #pay button
        payButton = Button(tkWindow,
text="Donate",command=mysqlpythondon).place(x=150
,y=250)


        #close
        def close():
            tkWindow.destroy()
            homepage()
```

```
        close= Button(tkWindow,
text="close",command=close)
        close.place(x=10,y=250)


        #main loop
        tkWindow.mainloop()


##################################################
##################################################
##################################################
########


def appointmentscode():
        def mysqlpythonapp():
            #mysql python connector

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
            if mycon.is_connected():
                print("Successfully Conected to
database")
            else:
                print("Access denied")
            cursor=mycon.cursor()
            st="insert into appointments
values('{}','{}','{}','{}','{}')".format(username
.get(),name.get(),age.get(),gender.get(),problems
.get())
            cursor.execute(st)
            mycon.commit()
```

```
            st="select * from appointments"
            cursor.execute(st)
            data=cursor.fetchall()
            for row in data:
                print(row)
            messagebox.showinfo("showinfo",
"Appointment Booked")

            mycon.close()

            tkWindow.destroy()

            homepage()

        #window
        tkWindow = Tk()
        tkWindow.geometry('500x350')
        tkWindow.title('Medicare')
        tkWindow.configure(bg='#008080')
        #getting screen width and height of
display
        #width= tkWindow.winfo_screenwidth()
        #height= tkWindow.winfo_screenheight()
        #setting tkinter window size
        #tkWindow.geometry("%dx%d" % (width,
height))

        #heading label
```

```
        headingLabel = Label(tkWindow, text="Book
Appointment").place(x=10,y=5)

        #username label and text entry box

        usernameLabel = Label(tkWindow,
text="User Name").place(x=10,y=50)

        username = StringVar()

        usernameEntry = Entry(tkWindow,
textvariable=username).place(x=80,y=50)


        #name label and text entry box

        nameLabel = Label(tkWindow,
text="Name").place(x=10,y=100)

        name = StringVar()

        nameEntry = Entry(tkWindow,
textvariable=name).place(x=80,y=100)


        #age label and entry box

        ageLabel =
Label(tkWindow,text="Age").place(x=10,y=150)

        age = StringVar()

        addressEntry = Entry(tkWindow,
textvariable=age).place(x=80,y=150)


        #gender label and entry box

        genderLabel =
Label(tkWindow,text="Gender").place(x=10,y=200)

        gender = StringVar()

        genderEntry = Entry(tkWindow,
textvariable=gender).place(x=80,y=200)


        #problems label and entry box
```

```
        problemsLabel = Label(tkWindow,text="What
are your symptoms? Describe the
problem:").place(x=10,y=250)

        problems  = StringVar()

        problemsEntry = Entry(tkWindow,
textvariable=problems ).place(x=300,y=250)


        #book appoinment button

        appointmentButton = Button(tkWindow,
text="Book
Appointment",command=mysqlpythonapp).place(x=150,
y=300)



        #close

        def close():

            tkWindow.destroy()

            homepage()


        close= Button(tkWindow,
text="close",command=close)

        close.place(x=10,y=300)


        #main loop

        tkWindow.mainloop()




##############################################
##############################################
```

```
##################################################
########

def availabilitycode():
    #add record
    def availability():

        #mysql part

mycon=sql.connect(host="localhost",user="root",pa
sswd="password", database="project")
        if mycon.is_connected():
            print("Successfully Conected to
database")
        else:
            print("Access denied")
        cursor=mycon.cursor()
        st="insert into availability
values('{}','{}','{}')".format(Submit_username.ge
t(),Submit_name.get(),Submit_numberofbedsrequired
.get())
        cursor.execute(st)
        mycon.commit()
        st="select * from  availability"
        cursor.execute(st)
        data=cursor.fetchall()
        for row in data:
            print(row)
        messagebox.showinfo("showinfo", "Booking
Successful")
```

```
        Submit_username.delete(0,END)
        Submit_name.delete(0,END)
        Submit_numberofbedsrequired.delete(0,END)

        mycon.close()

        ws.destroy()

        homepage()




ws = Tk()
ws.title('Medicare')
ws.config(bg='teal')

f = ('Times', 14)
var = StringVar()
var.set('male')




right_frame = Frame(
    ws,
    bd=2,
```

```
        bg='teal',
        relief=SOLID,
        padx=10,
        pady=10
        )

    Label(
        right_frame,
        text="Enter Username",
        bg='#CCCCCC',
        font=f
        ).grid(row=0, column=0, sticky=W,
pady=10)

    Label(
        right_frame,
        text="Enter Name",
        bg='#CCCCCC',
        font=f
        ).grid(row=1, column=0, sticky=W,
pady=10)

    Label(
        right_frame,
        text="Number of beds required",
        bg='#CCCCCC',
        font=f
        ).grid(row=5, column=0, sticky=W,
pady=10)
```

```
Submit_username=Entry(
    right_frame,
    font=f
    )

Submit_name = Entry(
    right_frame,
    font=f
    )


Submit_numberofbedsrequired=Entry(
    right_frame,
    font=f
    )

Submit_btn = Button(
    right_frame,
    width=15,
    text='Submit',
    font=f,
    relief=SOLID,
    cursor='hand2',
    command= availability
)
```

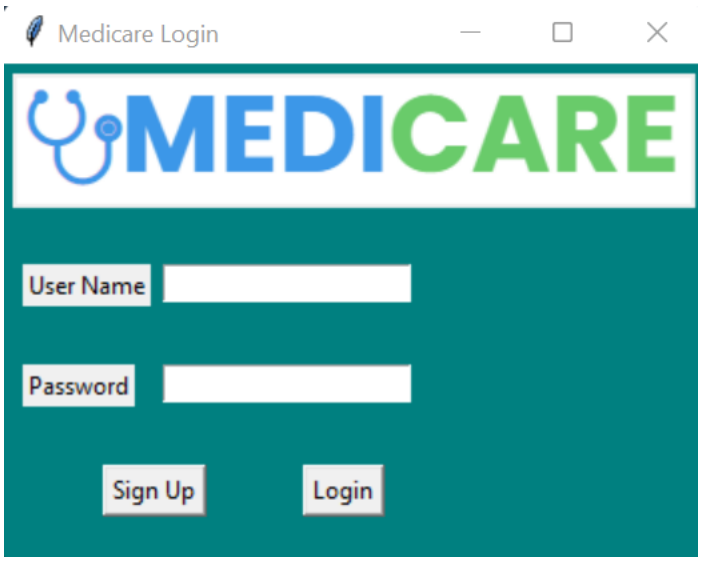
```
    #close
    def close():
        ws.destroy()
        homepage()


    close = Button(
        right_frame,
        width=15,
        text='Close',
        font=f,
        relief=SOLID,
        cursor='hand2',
        command= close
    )


    Submit_username.grid(row=0, column=1,
pady=10, padx=20)
    Submit_name.grid(row=1, column=1, pady=10,
padx=20)


    Submit_numberofbedsrequired.grid(row=5,
column=1, pady=10, padx=20)
    Submit_btn.grid(row=8, column=1, pady=10,
padx=20)
    right_frame.pack()


    close.grid(row=5, column=1, pady=10, padx=10)
```

```
    close.grid(row=10, column=1, pady=10,
padx=20)
    right_frame.pack()



    ws.mainloop()







##################################################
##################################################
##################################################
########



#login page code
#window
tkWindow = Tk()
tkWindow.geometry('350x250')
tkWindow.title('Medicare Login')
tkWindow.configure(bg='#008080')

#logo
img1= ImageTk.PhotoImage(Image.open('logo1.png'))
panel = Label(tkWindow, image = img1)
panel.place(x=5, y=5)

#username label and text entry box
```

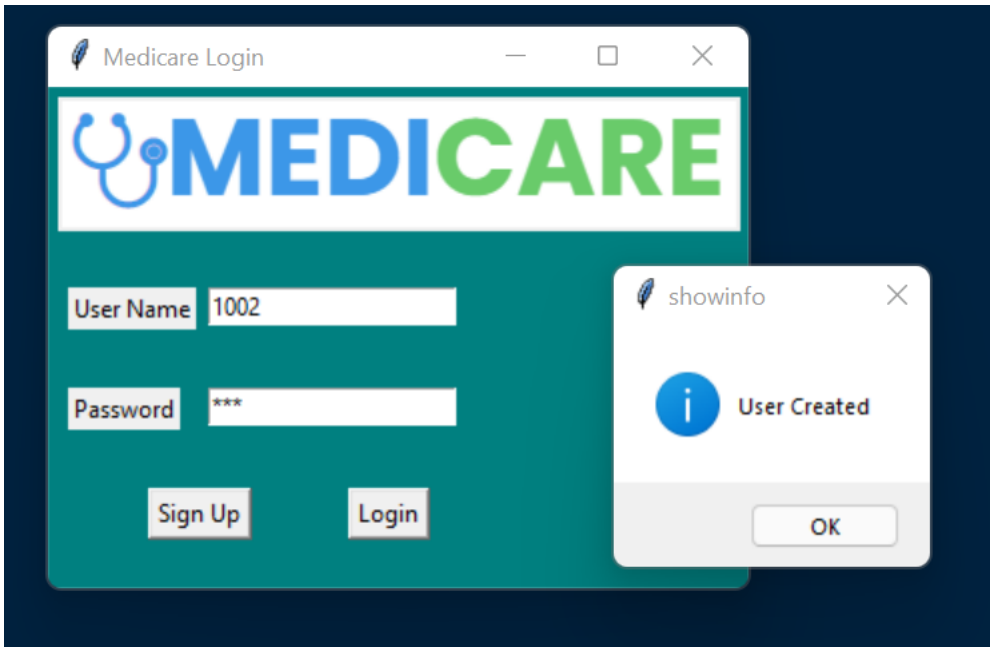
```
usernameLabel = Label(tkWindow, text="User
Name").place(x=10,y=100)

username = StringVar()

usernameEntry = Entry(tkWindow,
textvariable=username).place(x=80,y=100)


#password label and password entry box

passwordLabel =
Label(tkWindow,text="Password").place(x=10,y=150)

password = StringVar()

passwordEntry = Entry(tkWindow,
textvariable=password,
show='*').place(x=80,y=150)


validateLogin = partial(validateLogin, username,
password)


#login button


loginButton = Button(tkWindow, text="Login",
command= mysqlpythonlogin).place(x=150,y=200)


#signup button

signupButton = Button(tkWindow, text="Sign Up",
command=mysqlpythonsignup).place(x=50,y=200)


#main loop

tkWindow.mainloop()
```

OUTPUTS

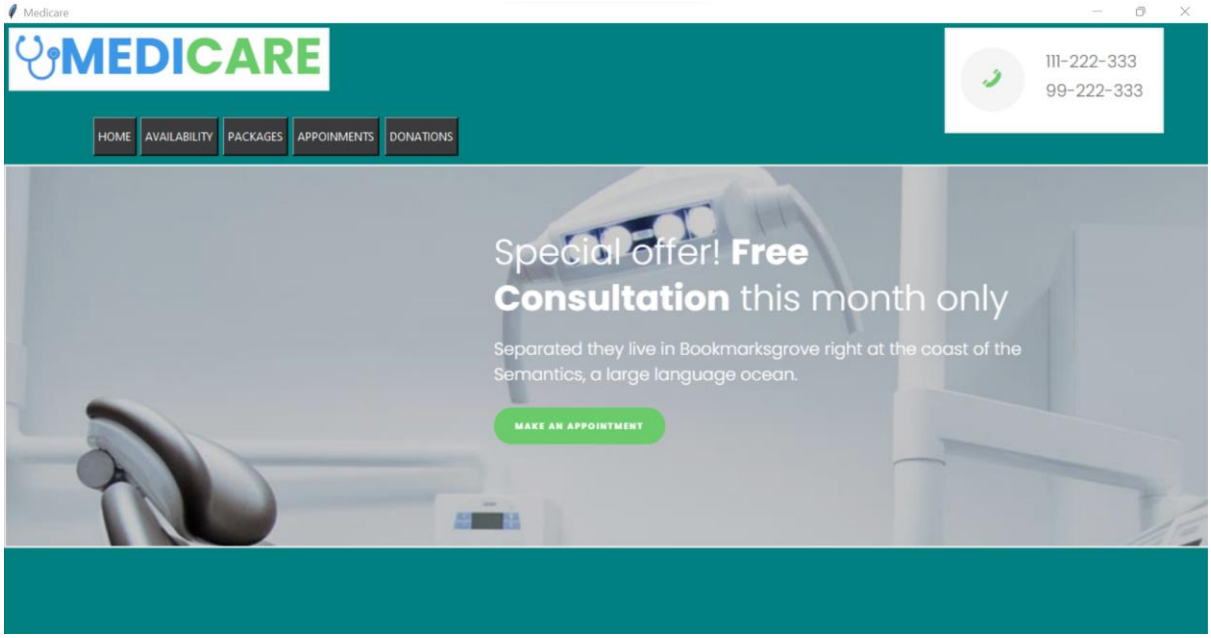
Login Page:



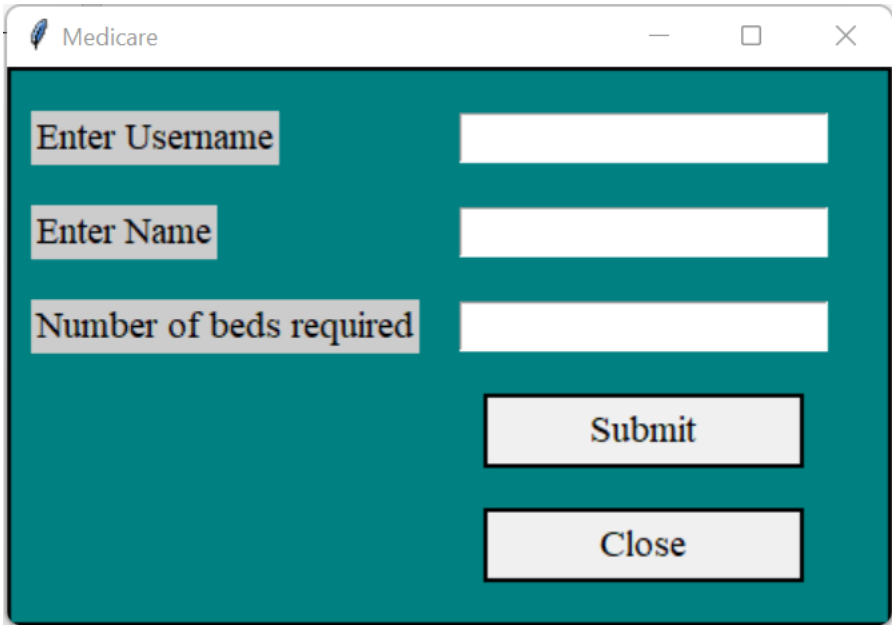
Signing Up:



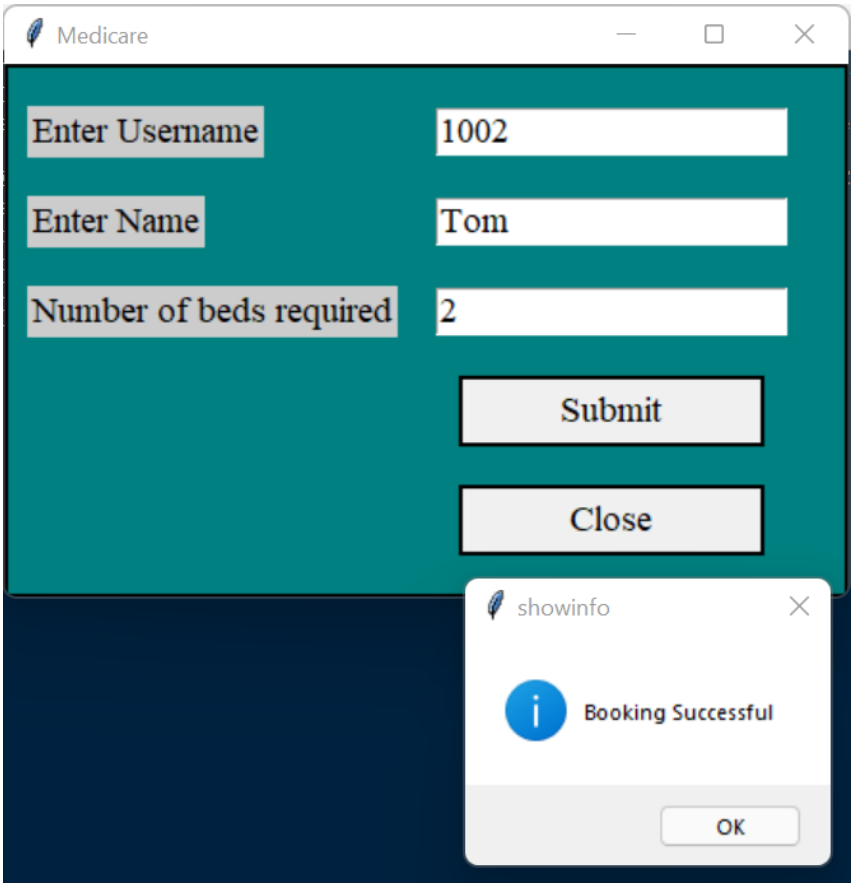
Home Page:


Medicare
MEDICARE
HOME
AVAILABILITY
PACKAGES
APPOINMENTS
DONATIONS
111-222-333
99-222-333
Special offer! Free
Consultation this month only
Separated they live in Bookmarksgrove right at the coast of the Semantics, a large language ocean.
MAKE AN APPOINTMENT

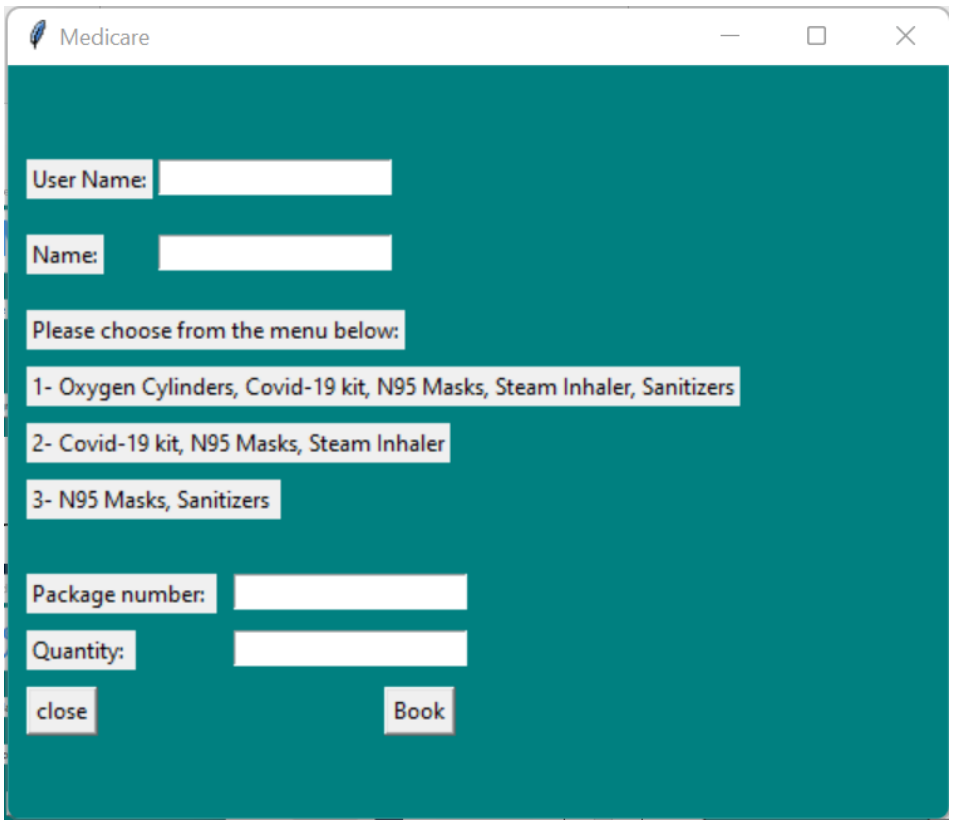
Availability Module:



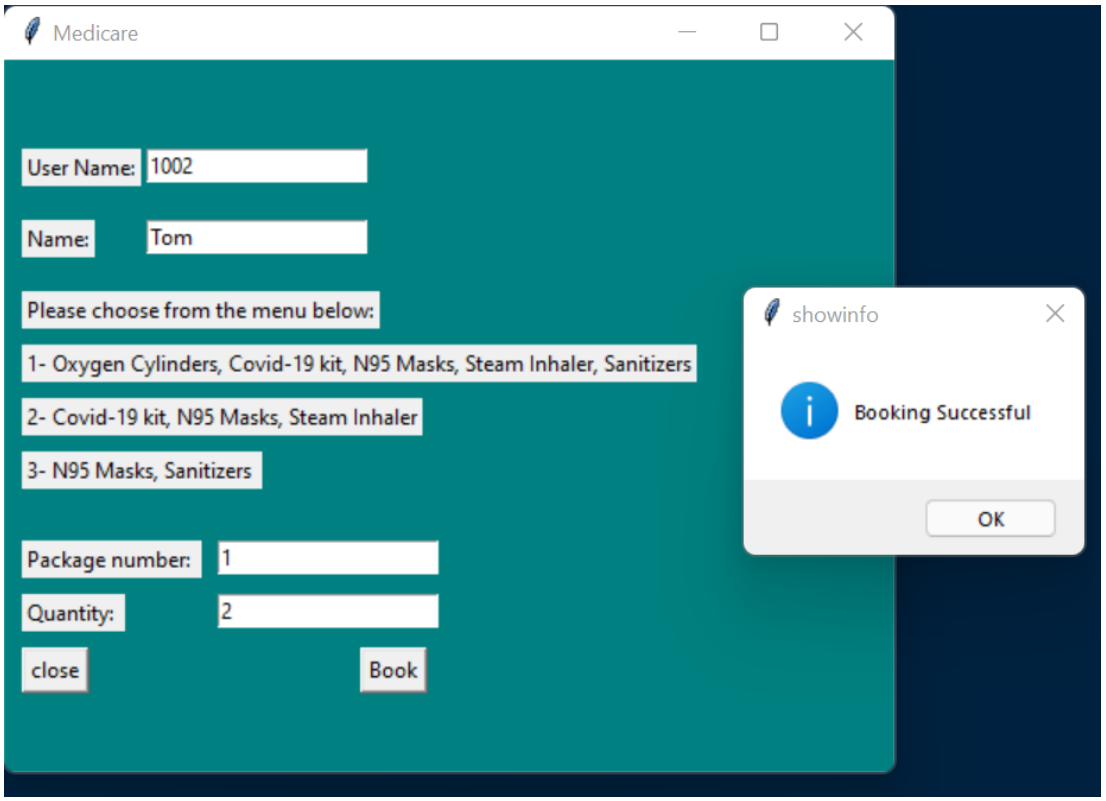
Booking Availability Module:



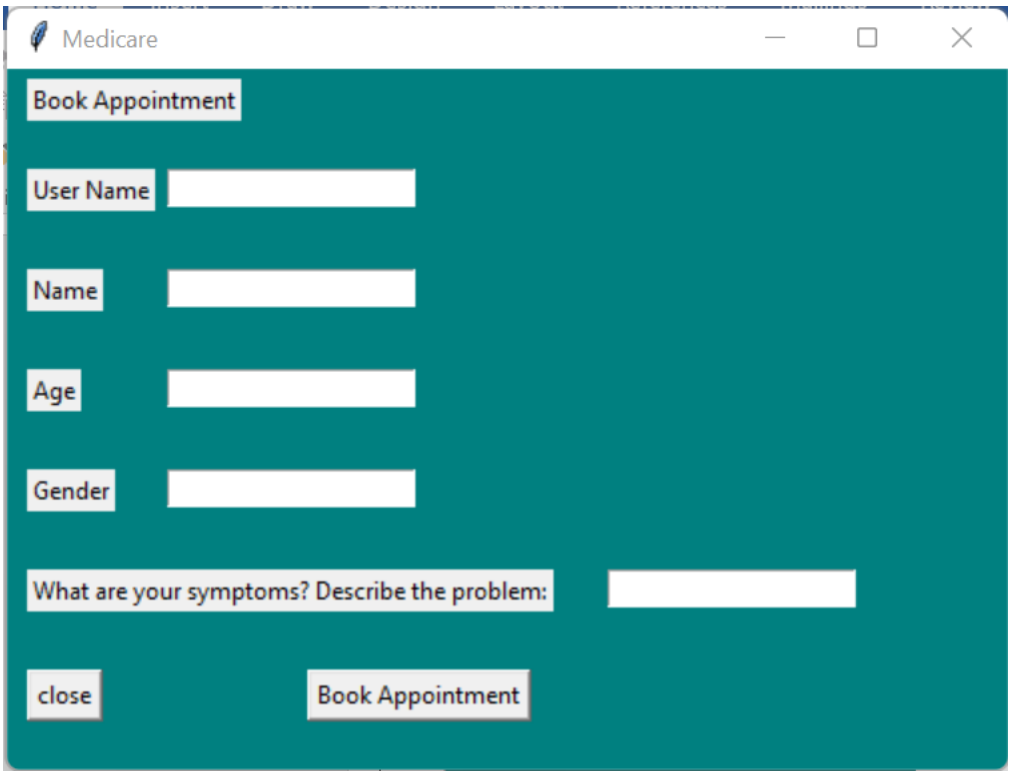
Packages Module:



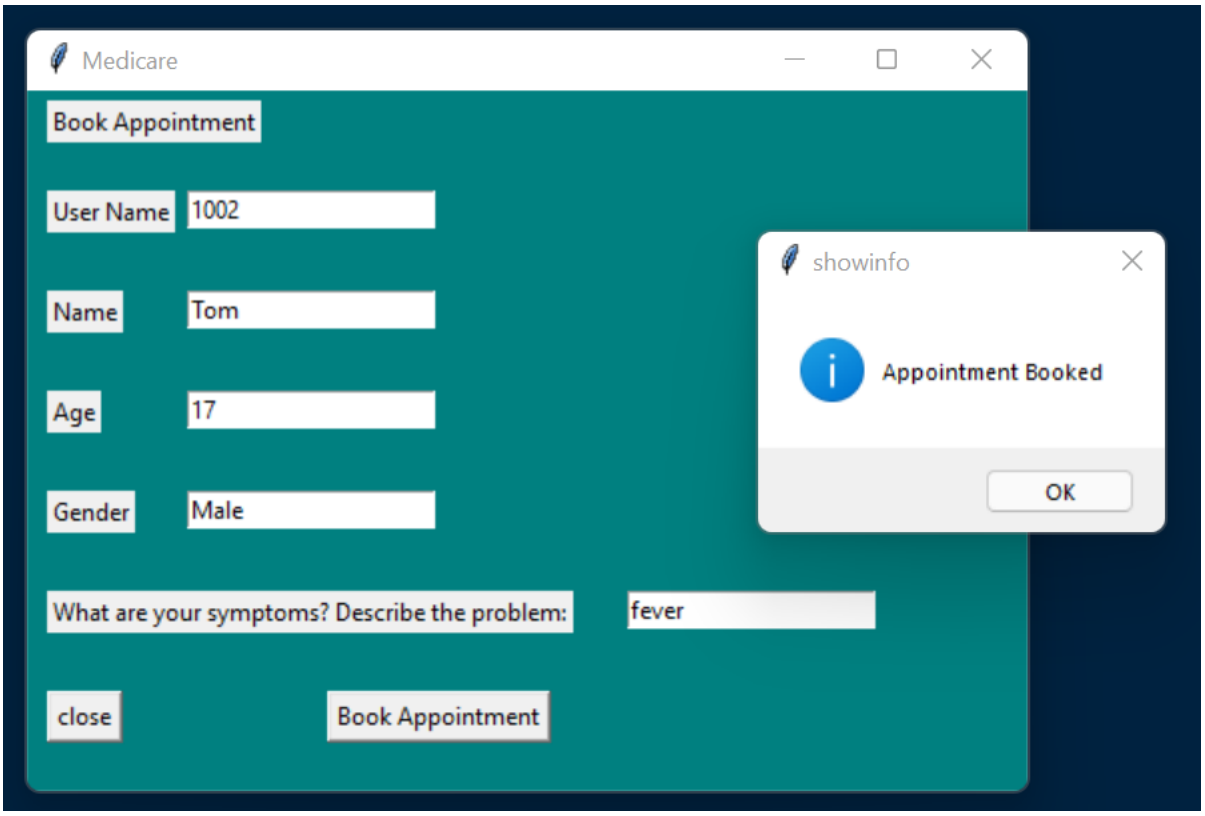
Booking a Package:



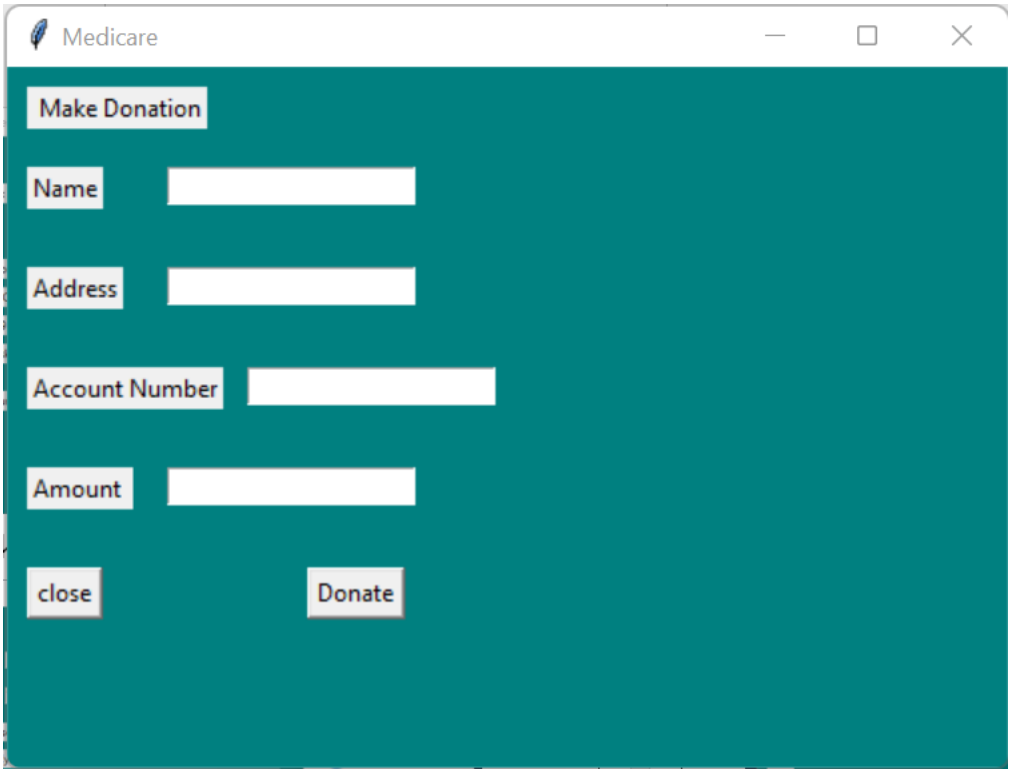
Appointments Module:



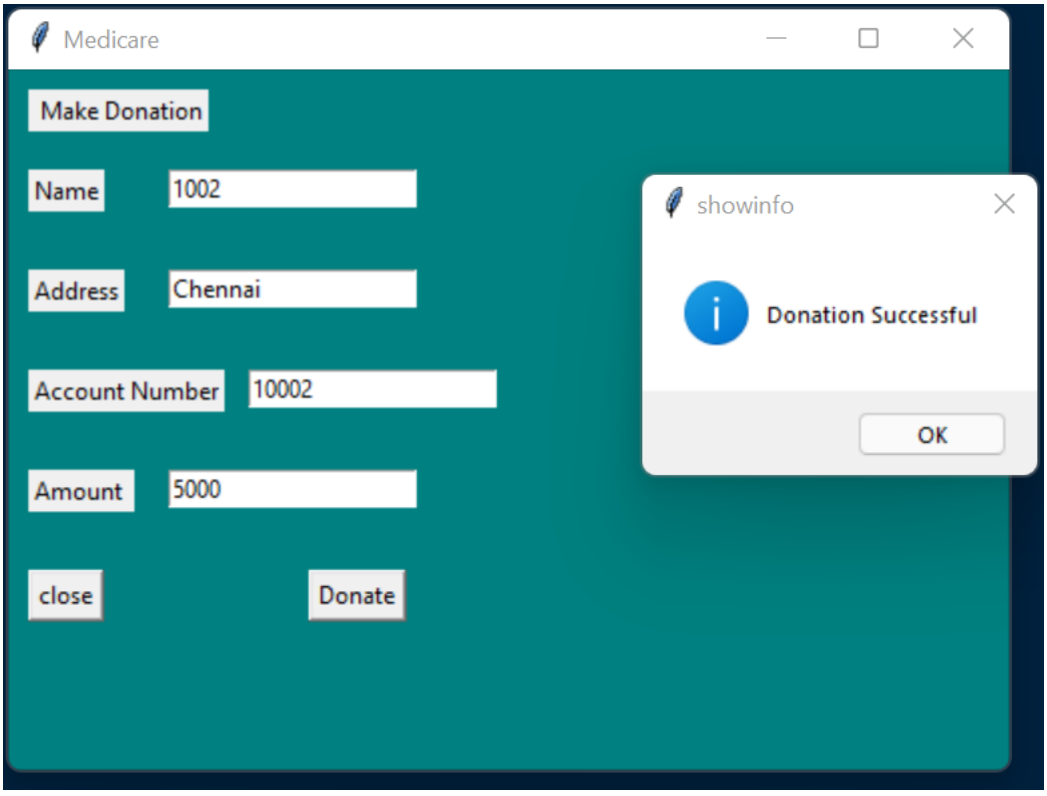
Booking an Appointment:



Donations Module:

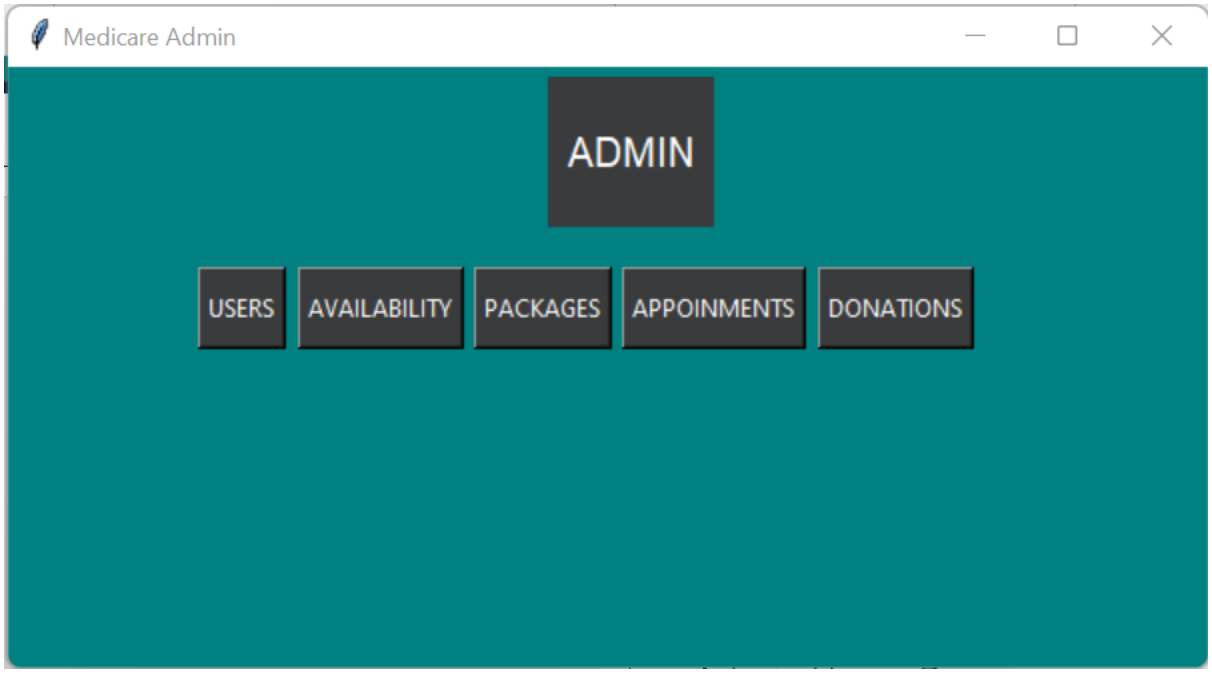


Making a donation:

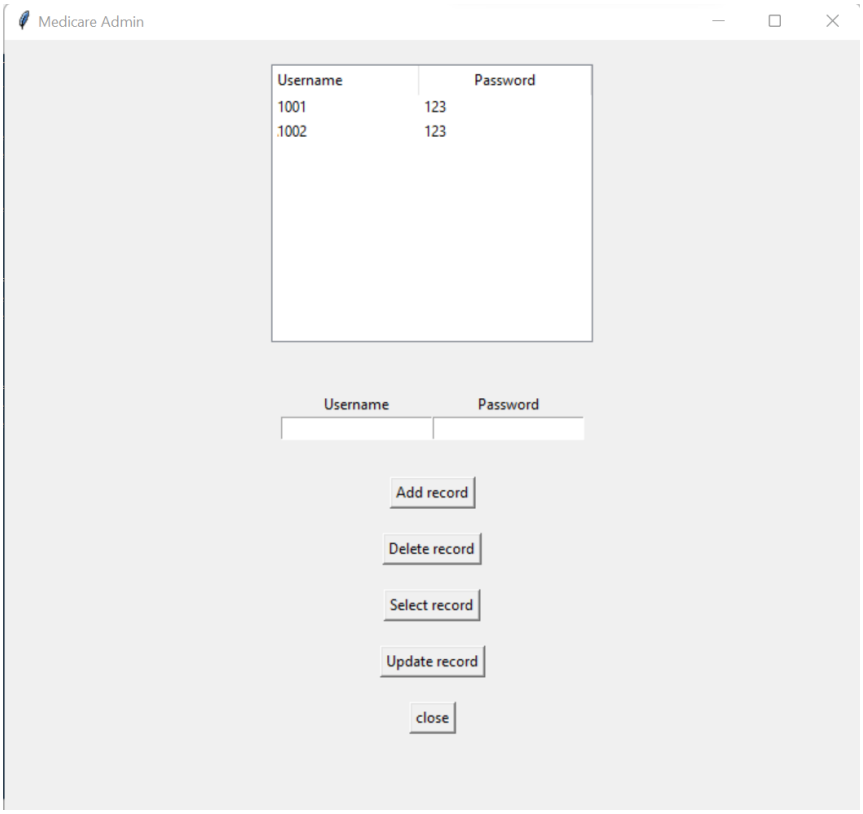


Sign in as Admin:

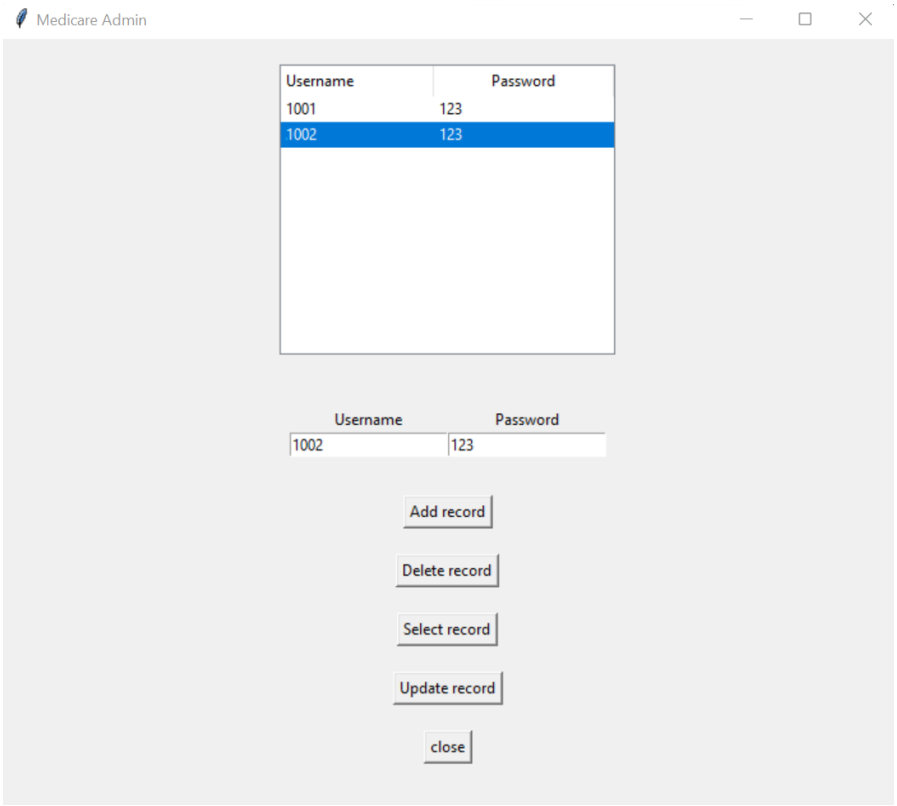
Admin Home Page :



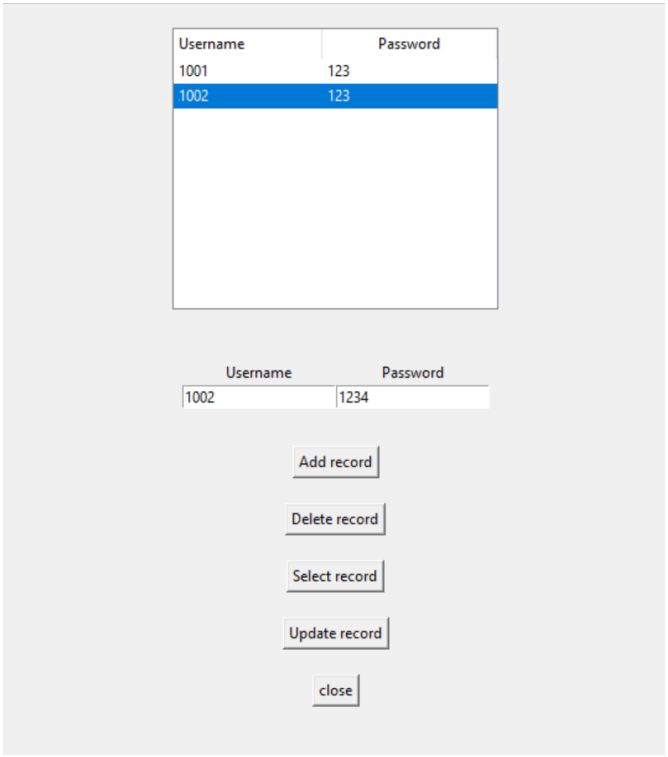
Admin Users Module:

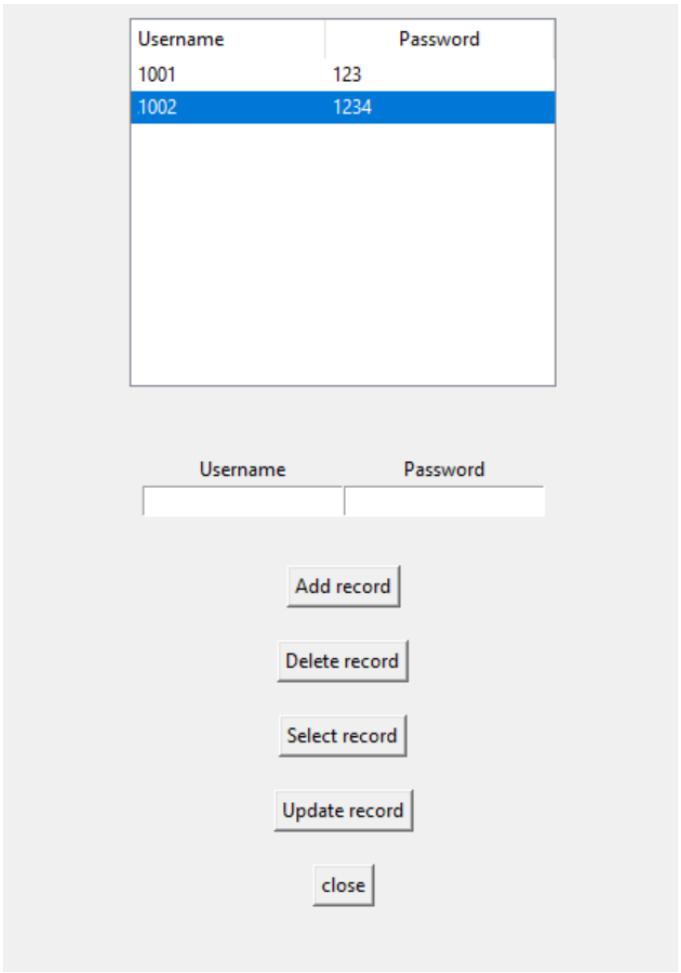


Selecting Record form Table:

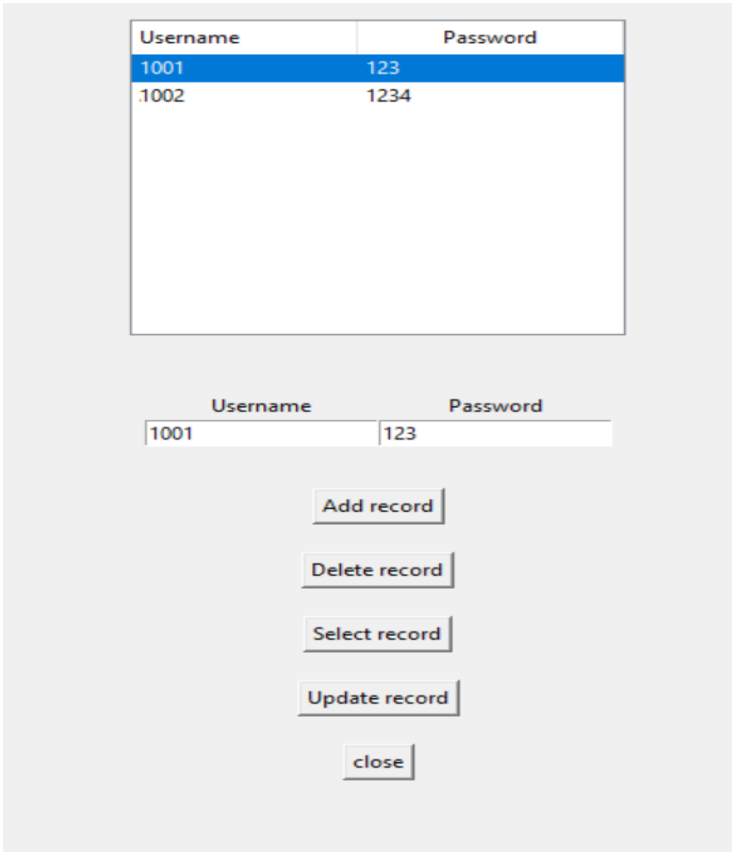


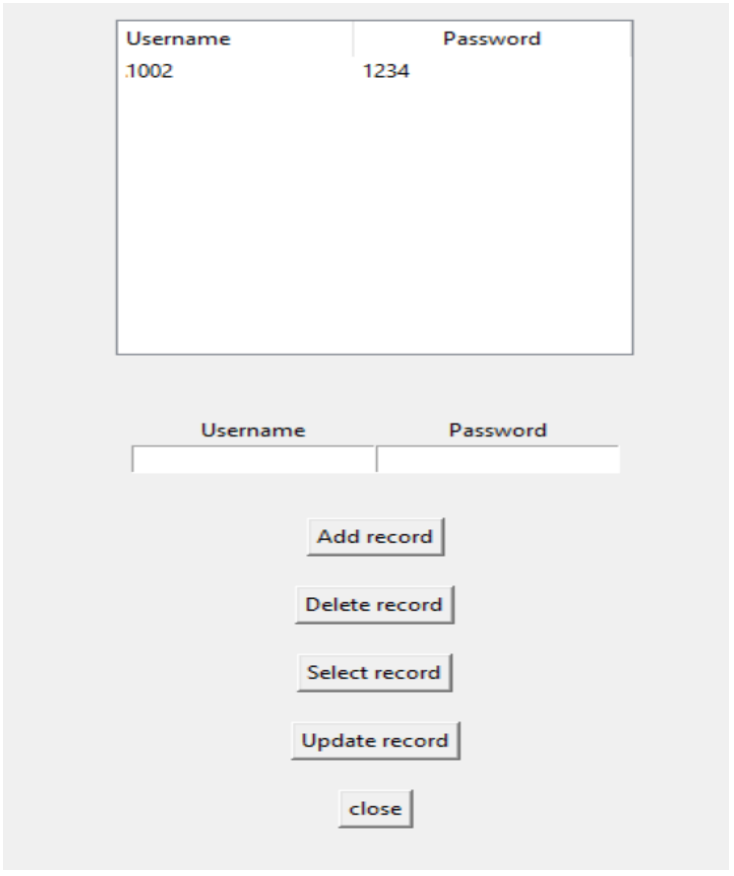
Updating Record:



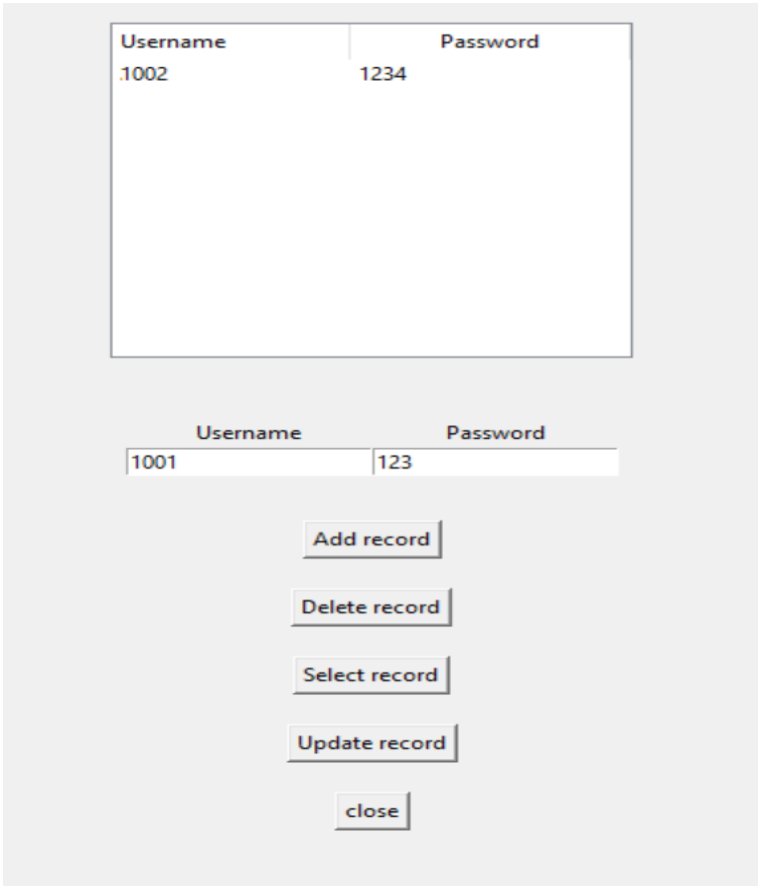


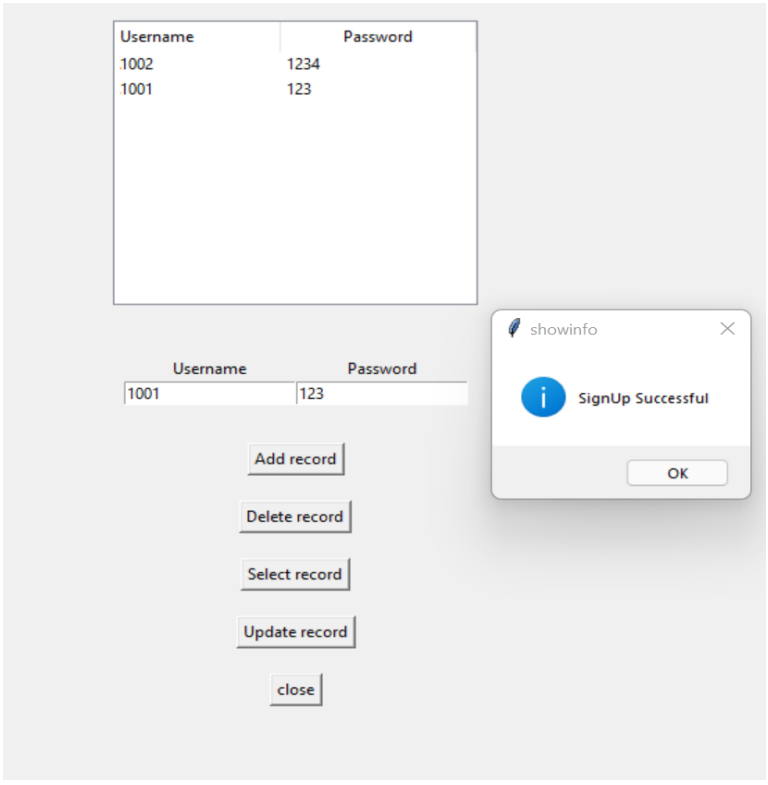
Deleting Record:



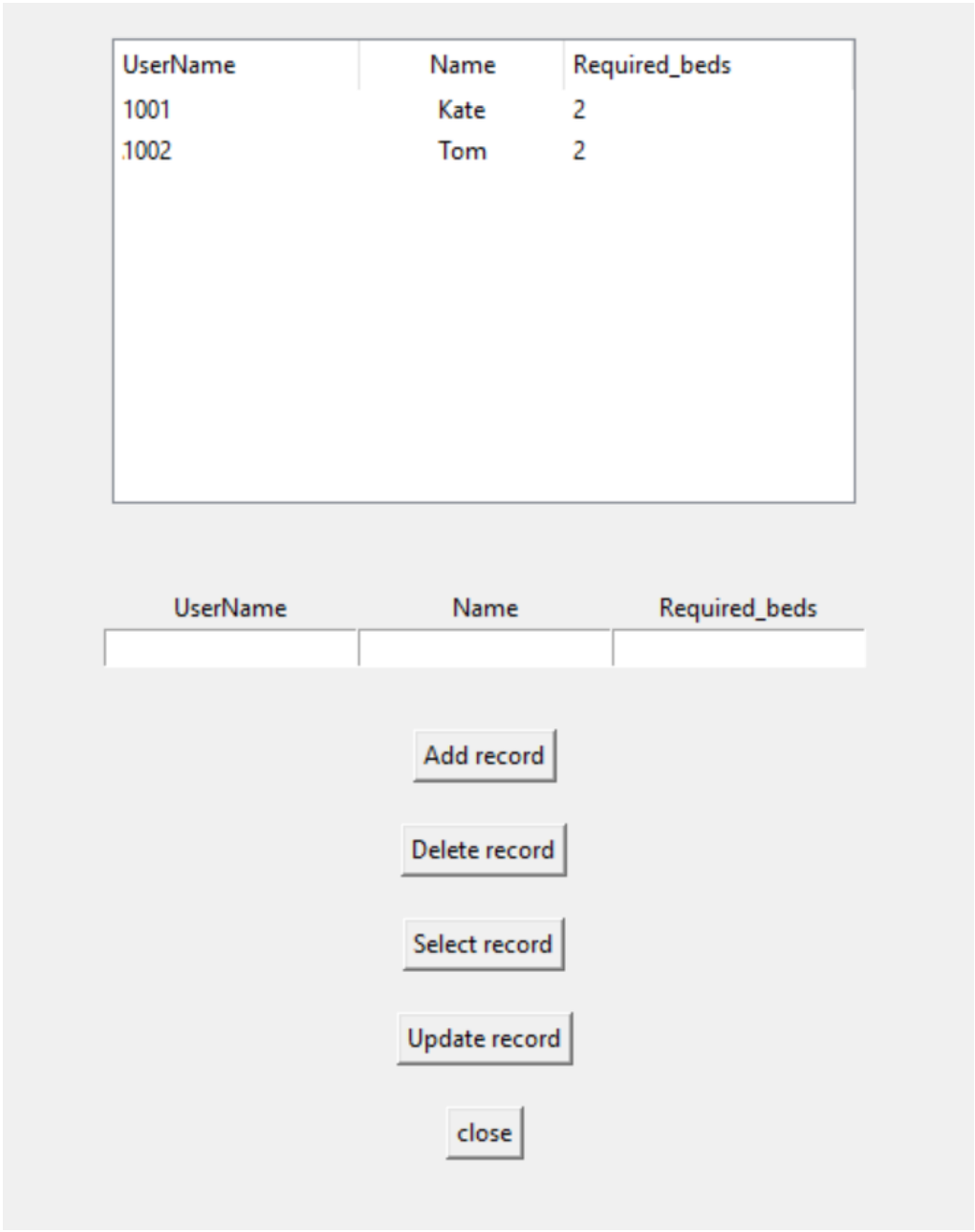


Adding Record:

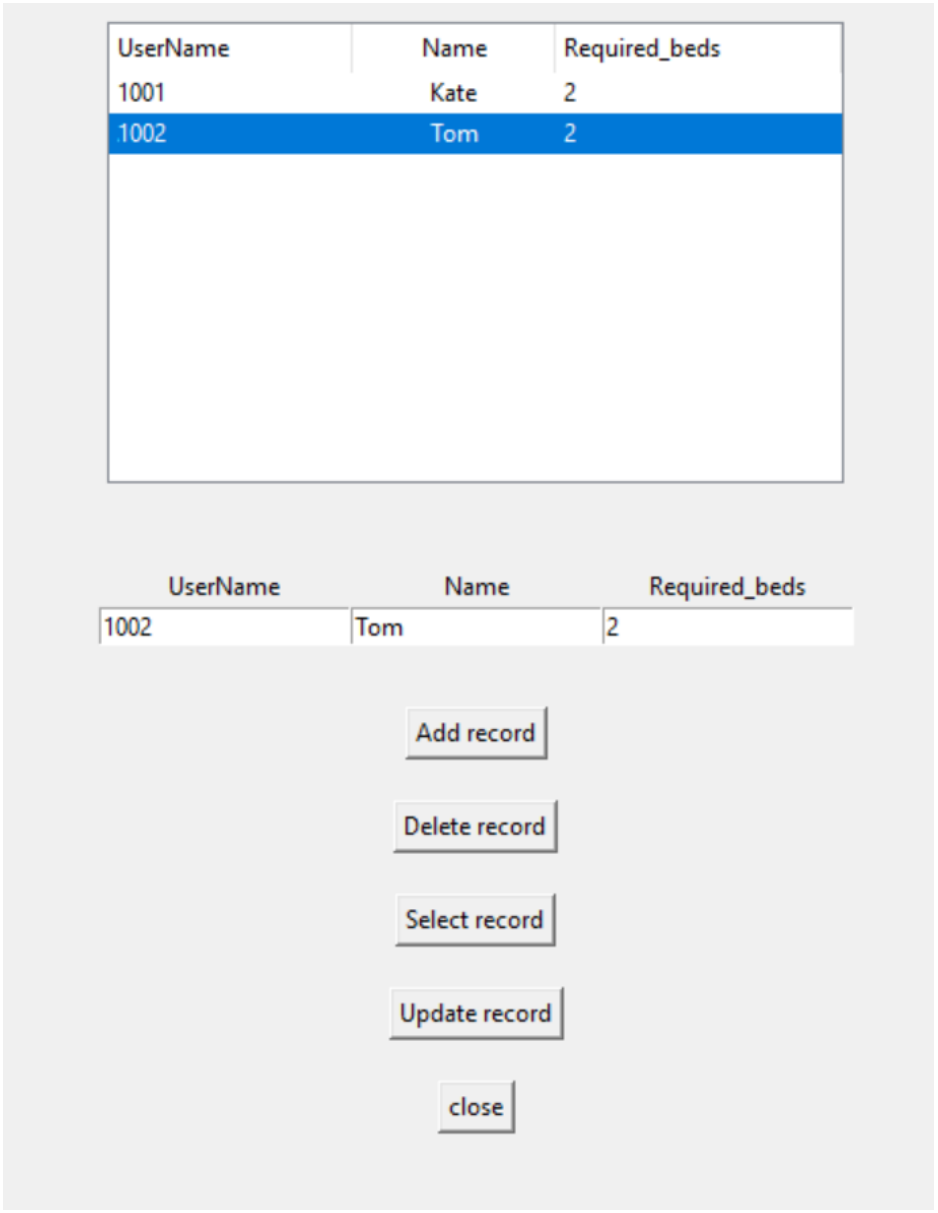




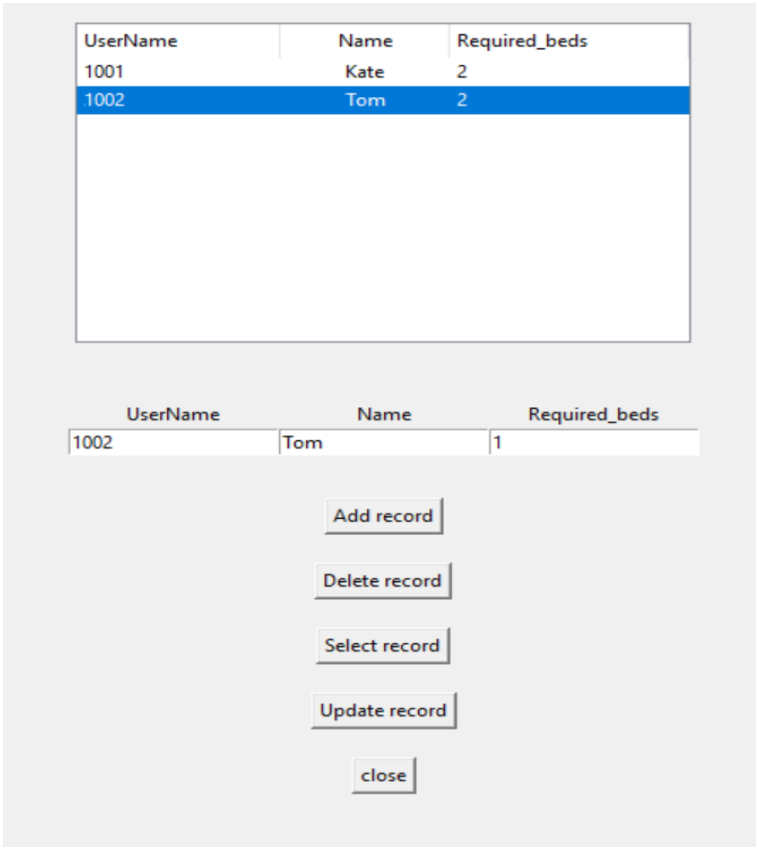
Admin Availability Module:

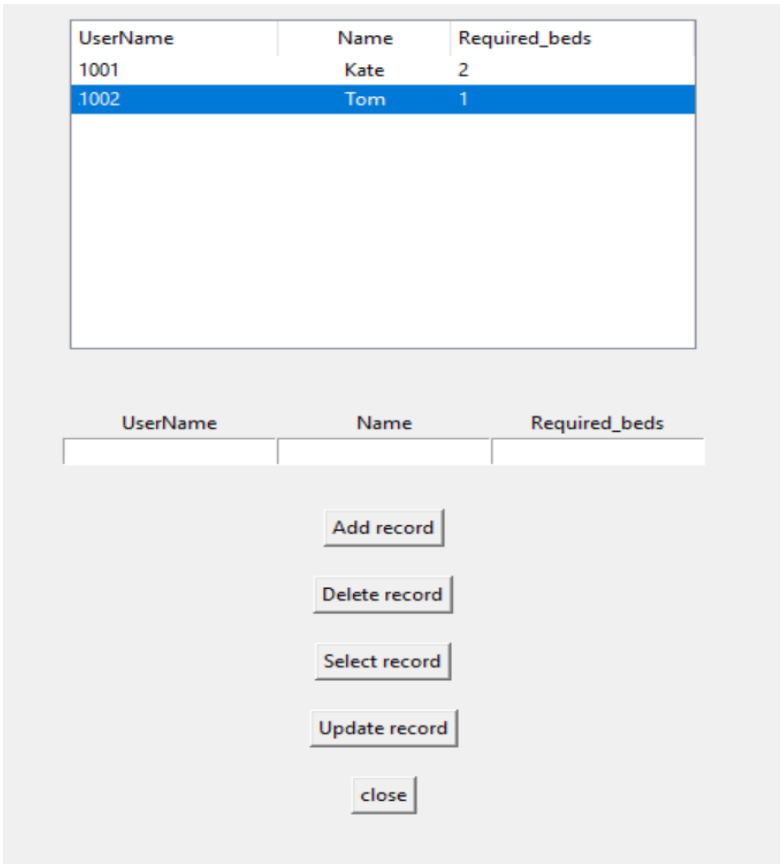


Selecting Record form Table:

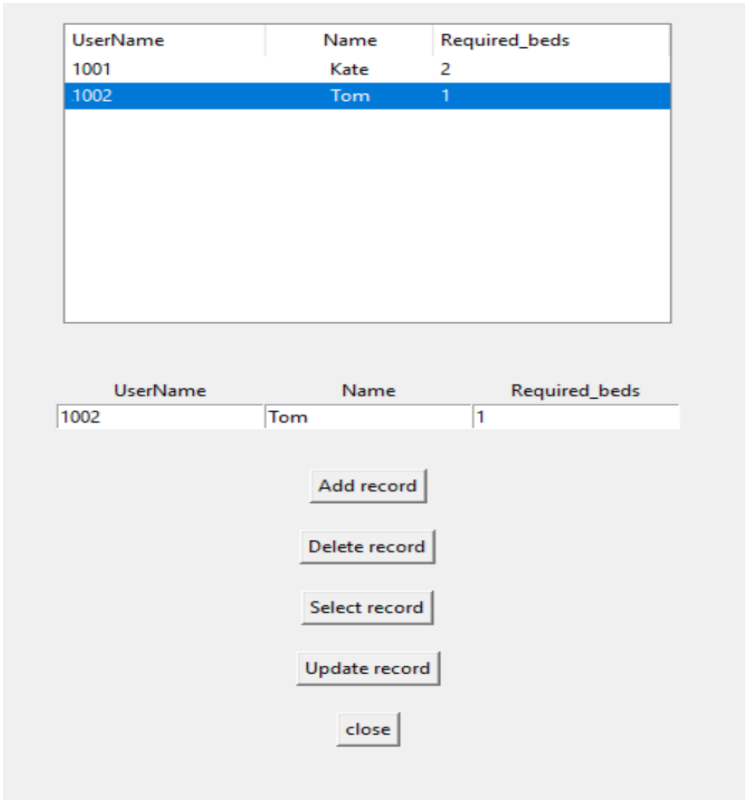


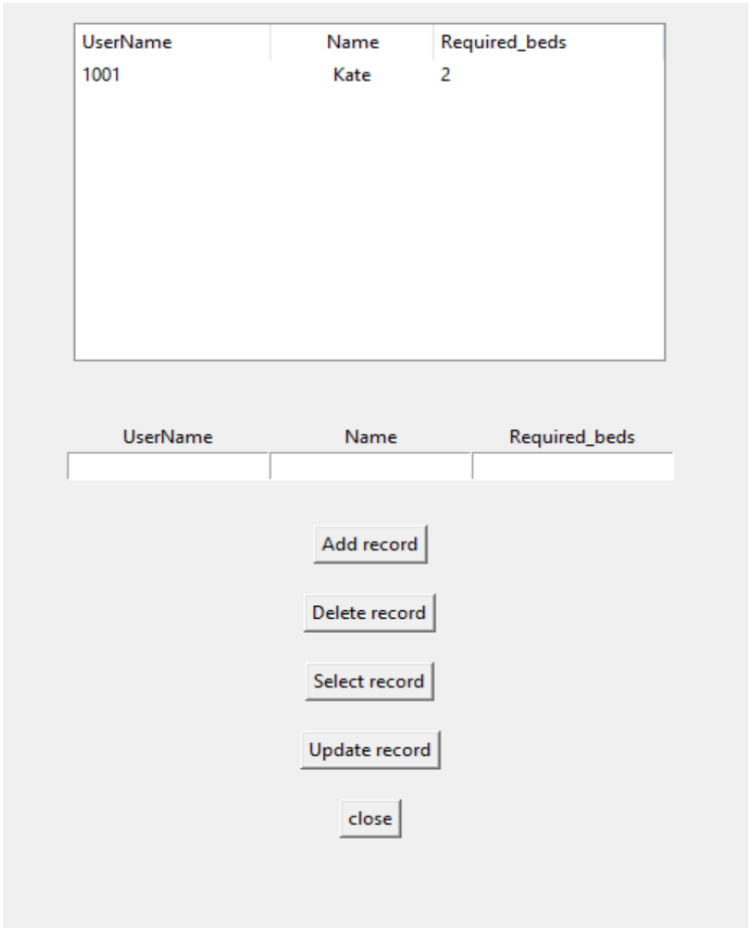
Updating Record:



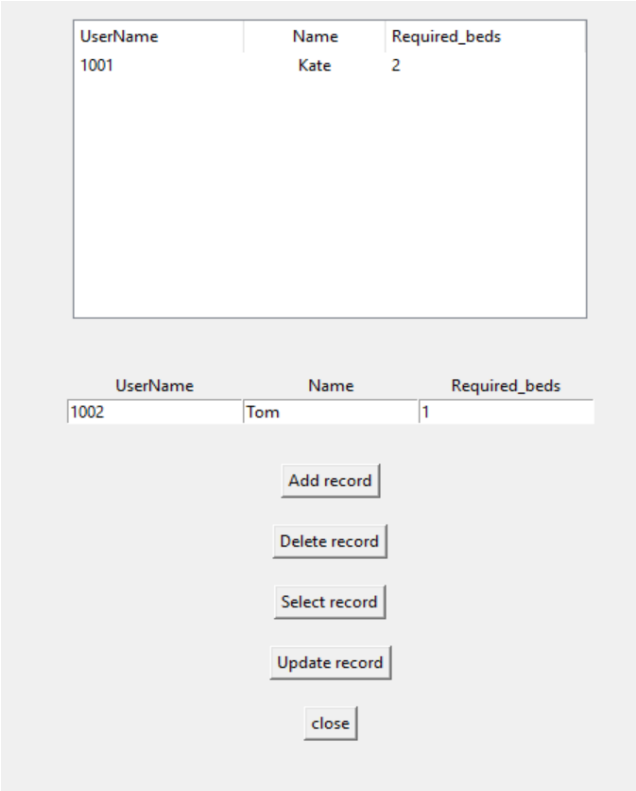


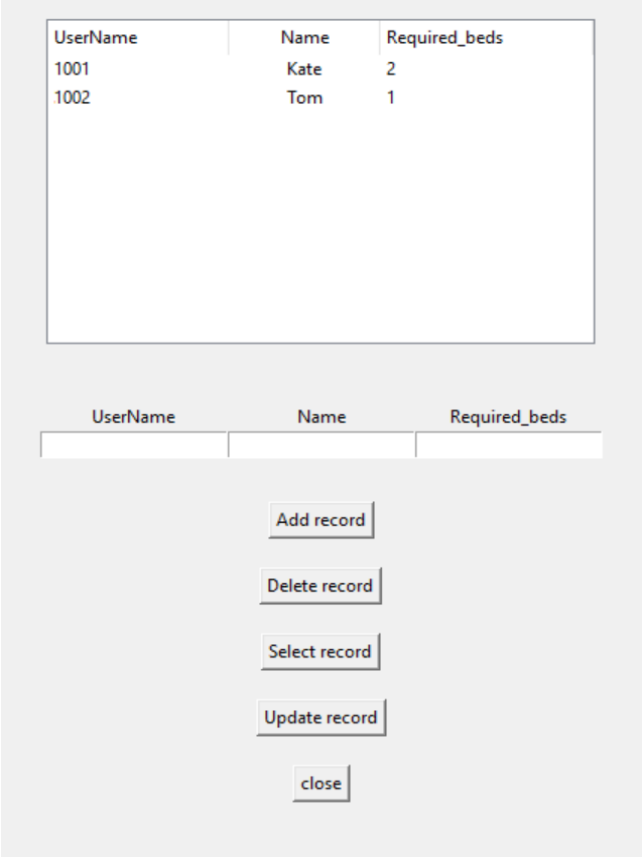
Deleting Record:



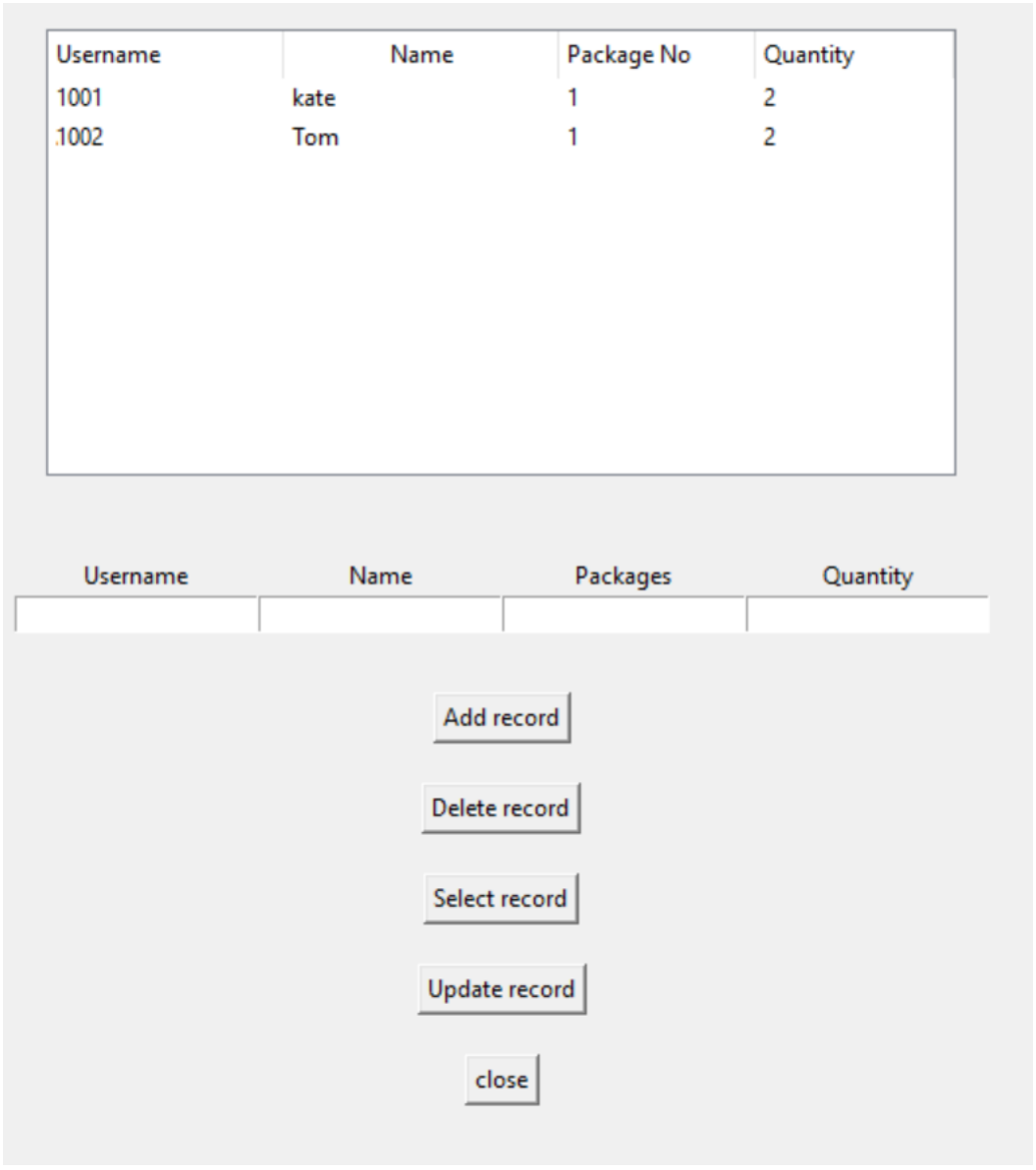


Adding Record:

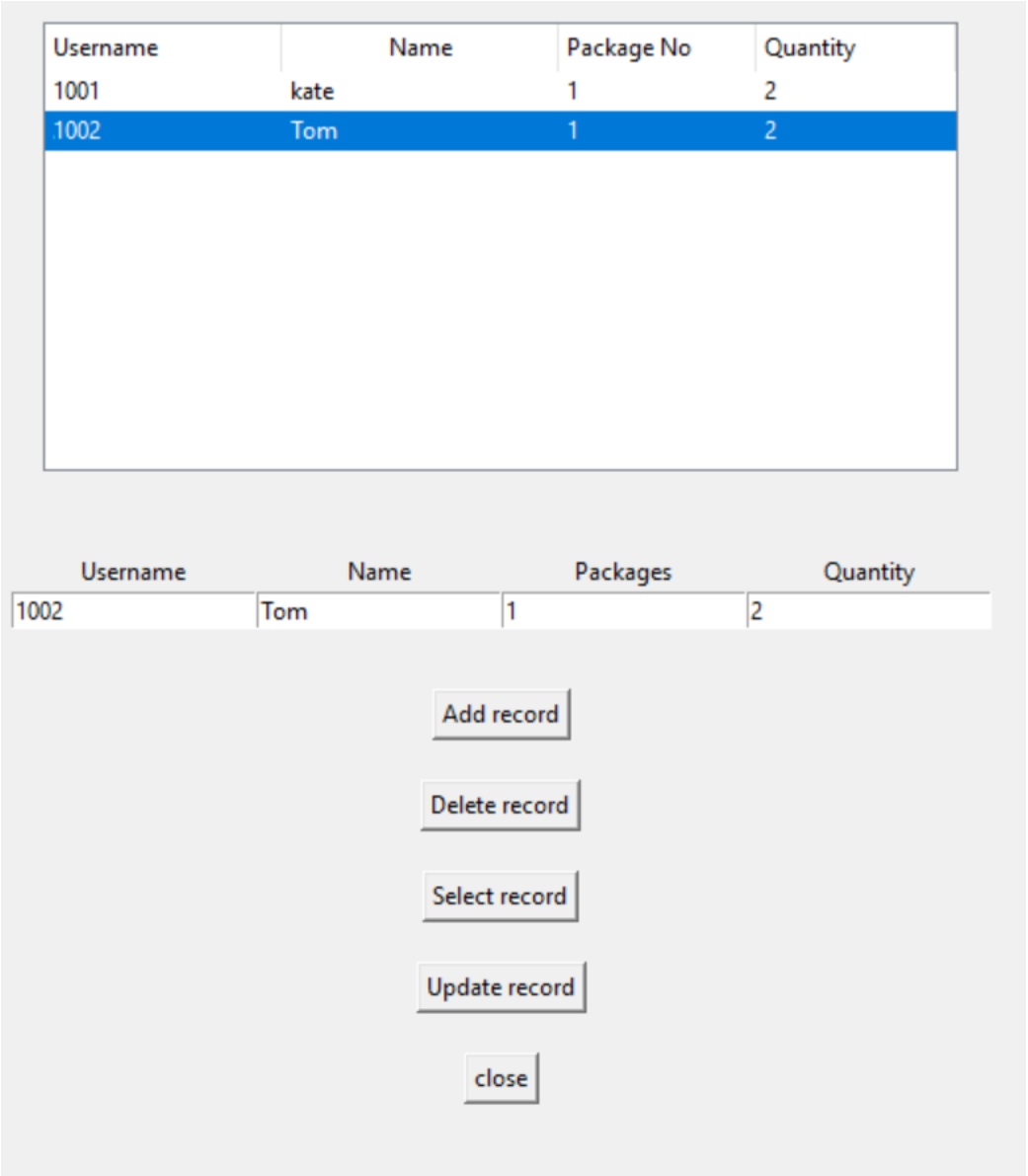




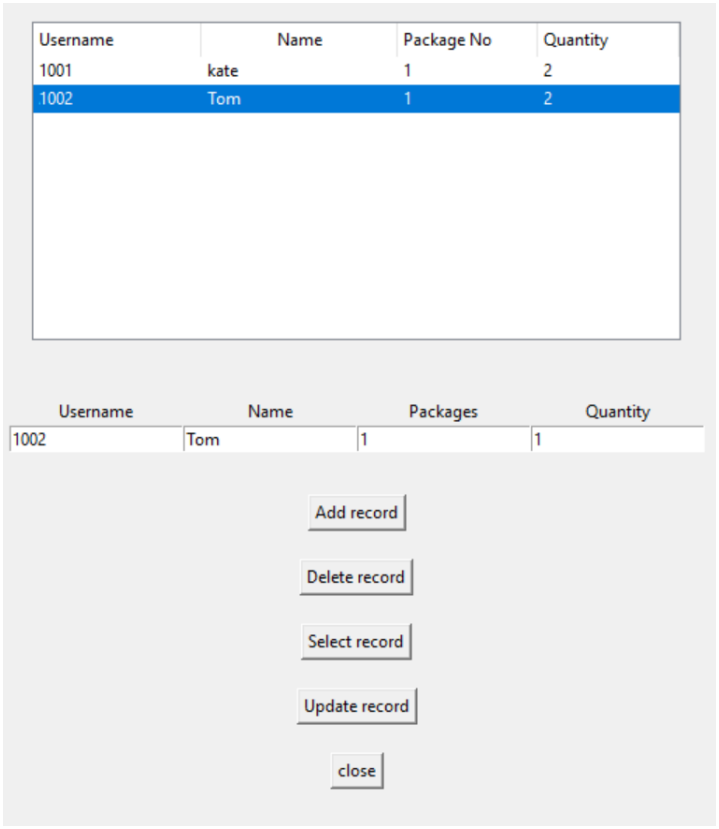
Admin Packages Module:

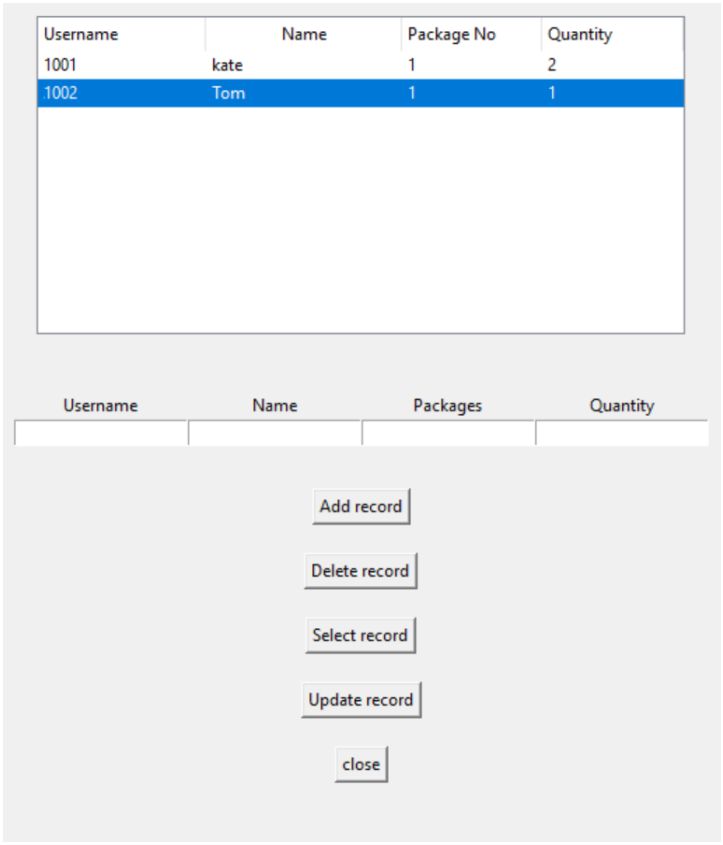


Selecting Record form Table:

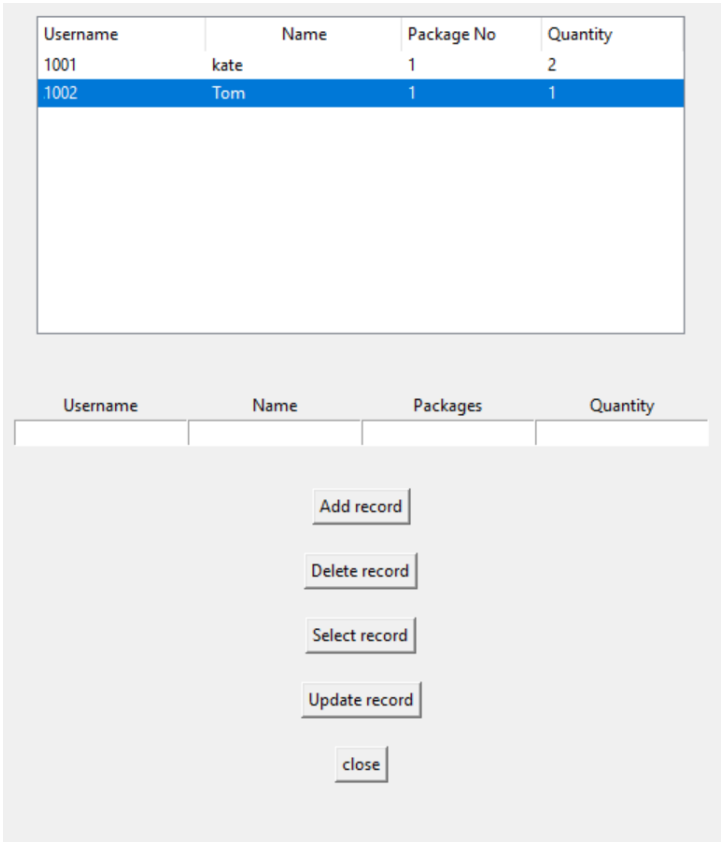


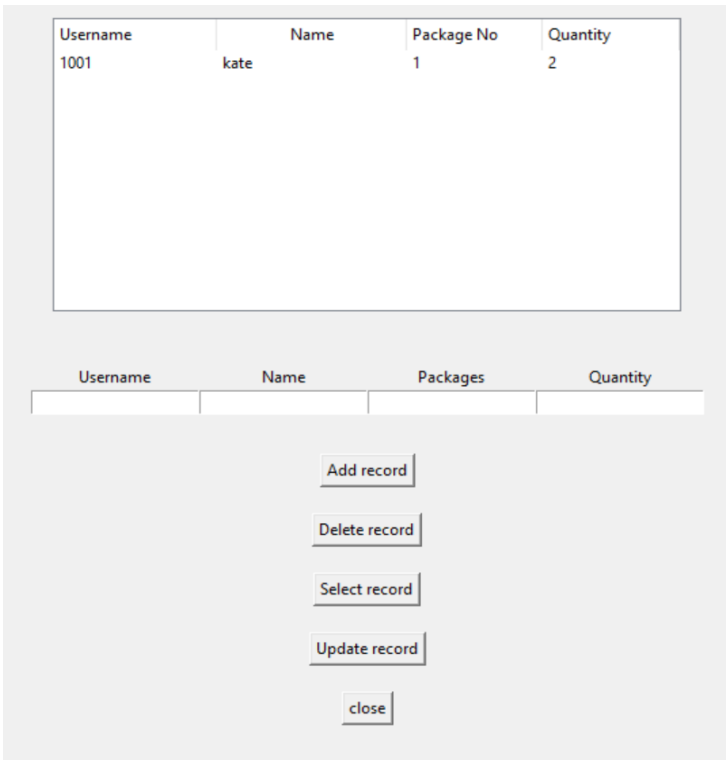
Updating Record:



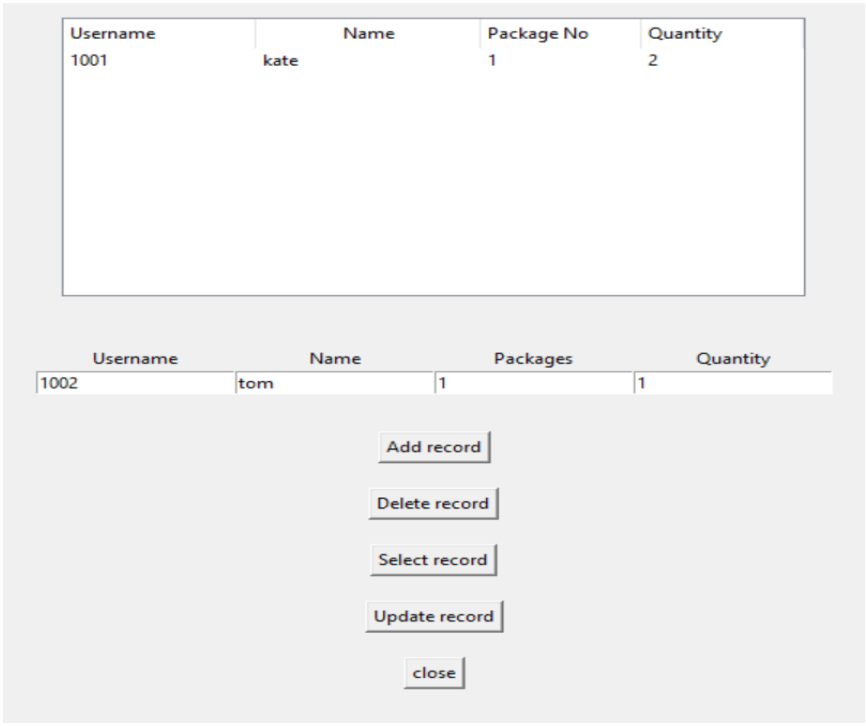


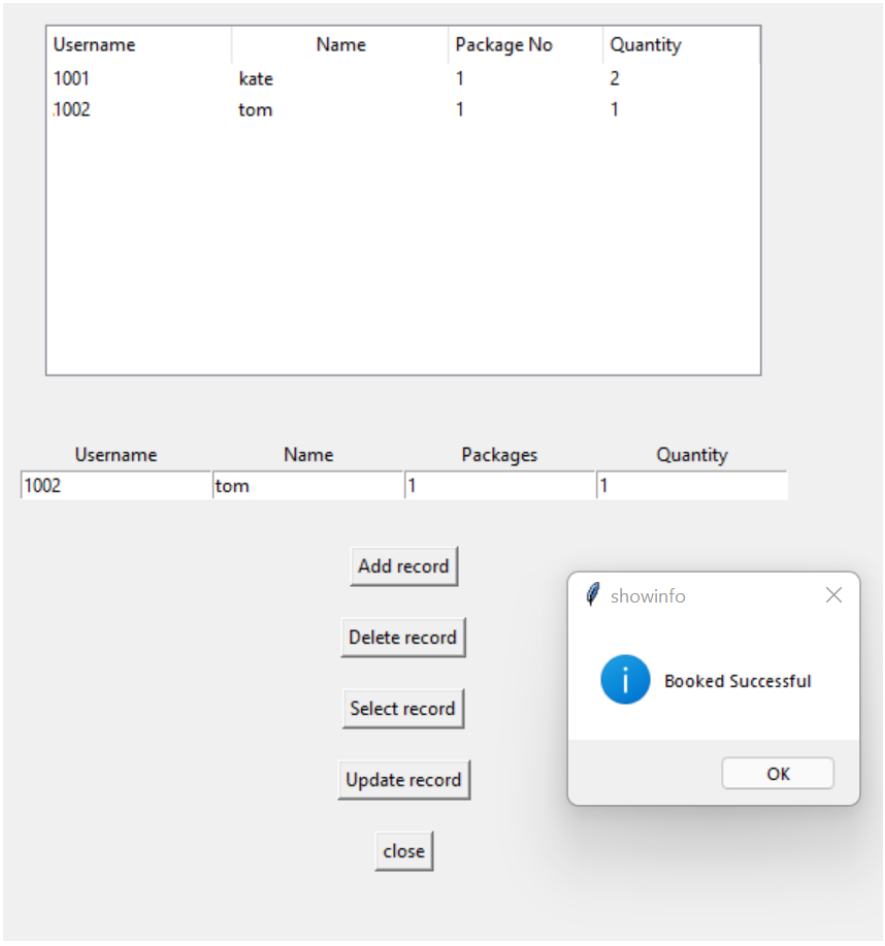
Deleting Record:



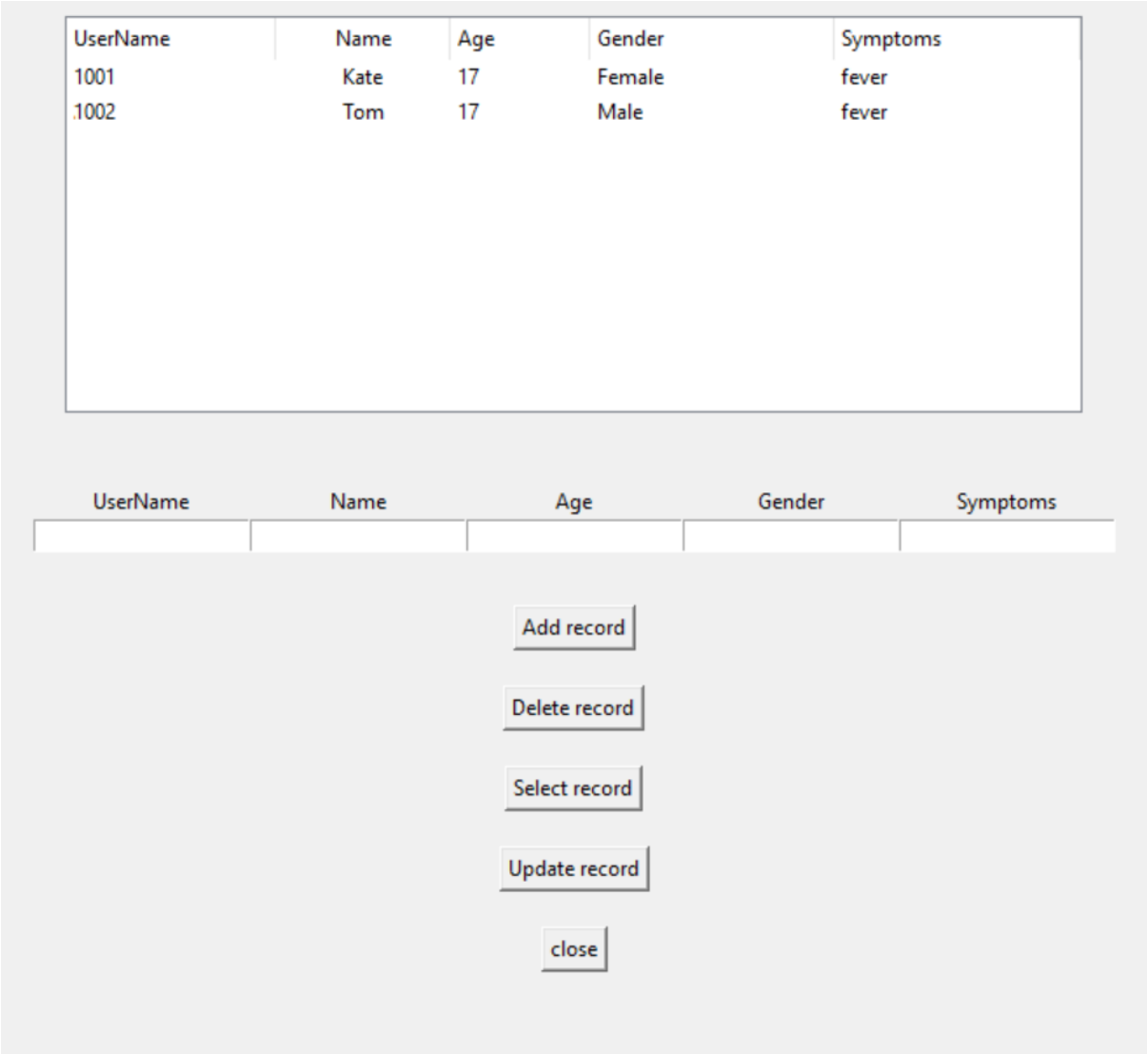


Adding Record:

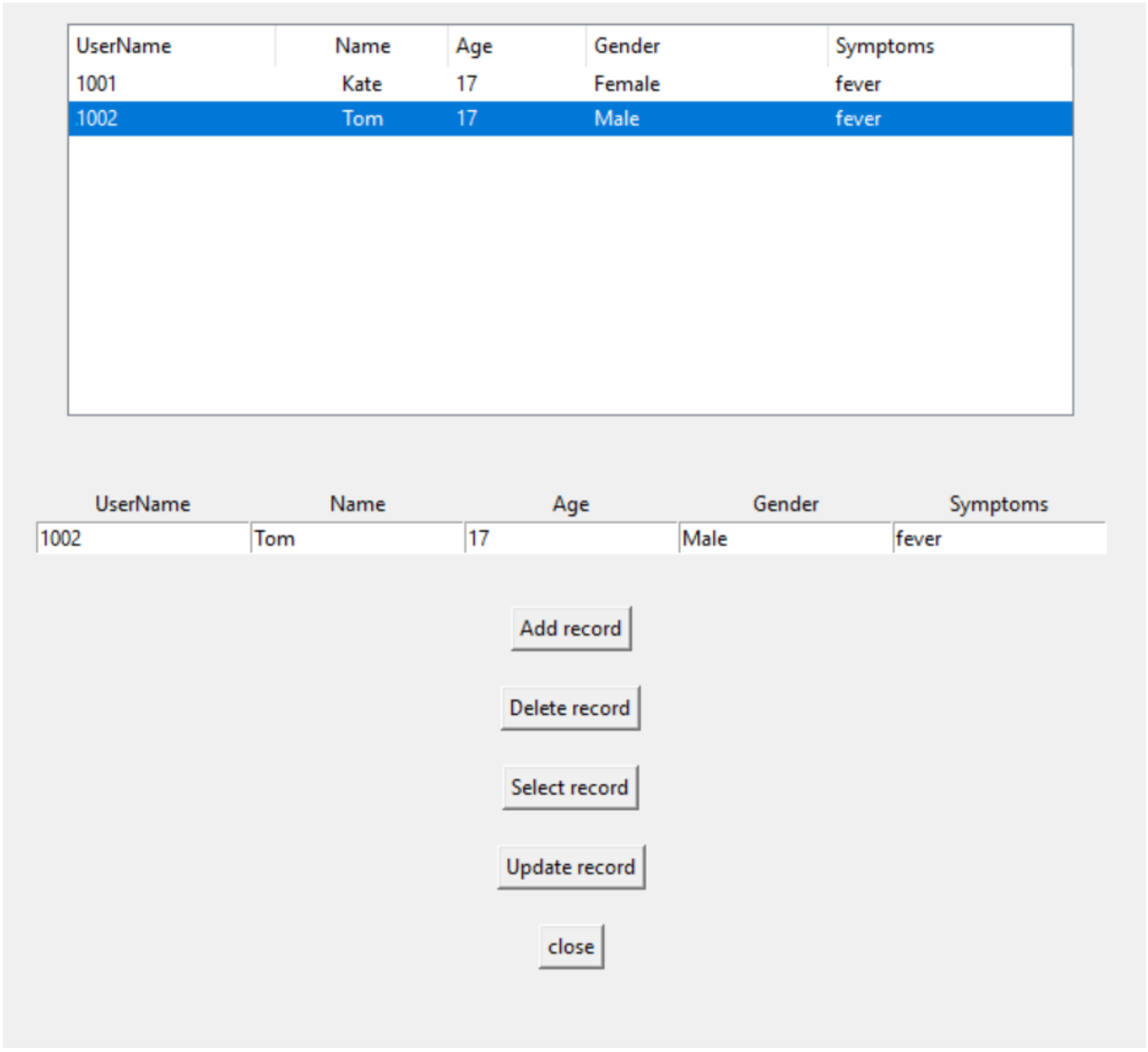




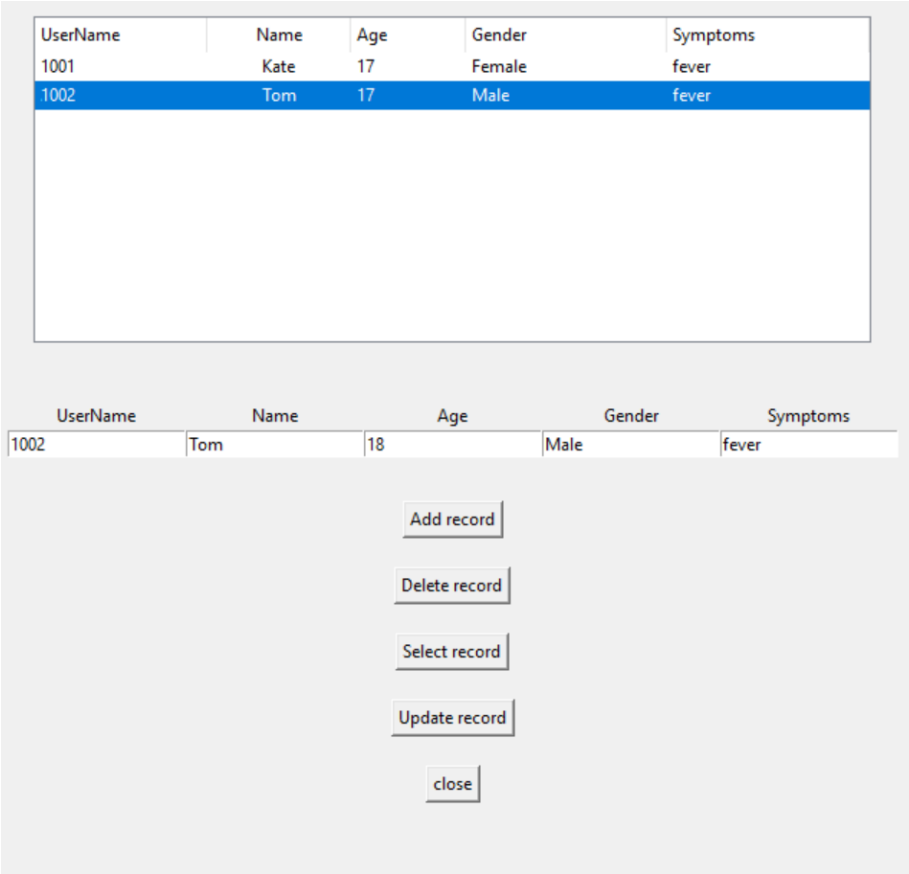
Admin Appointments Module:

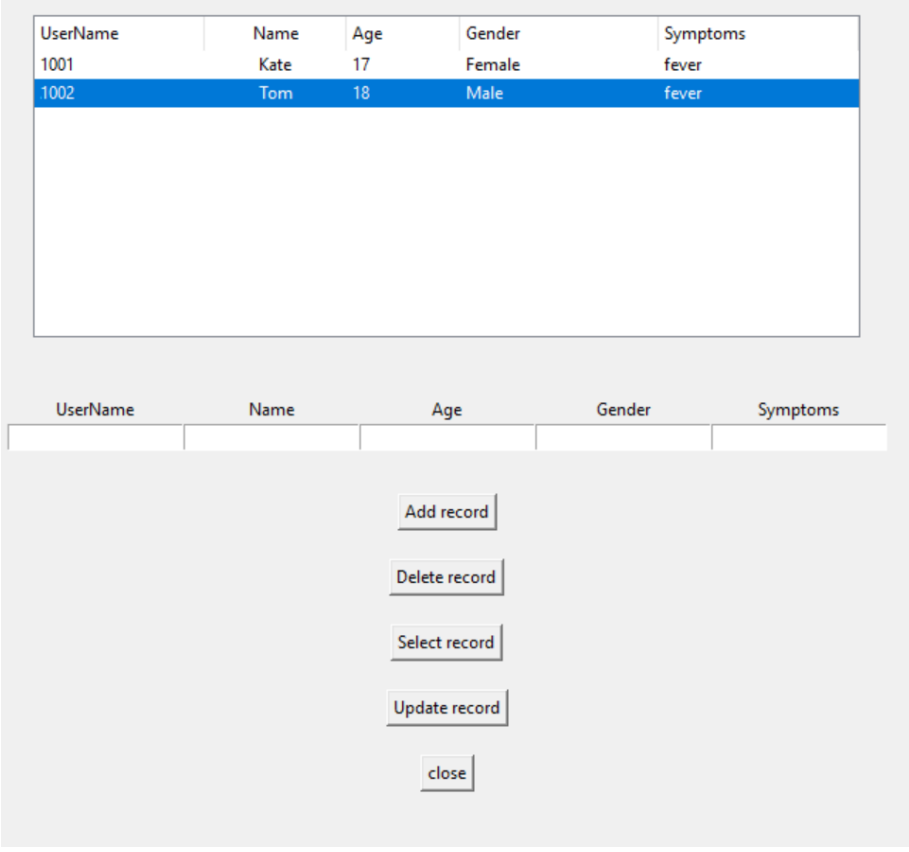


Selecting Record form Table:

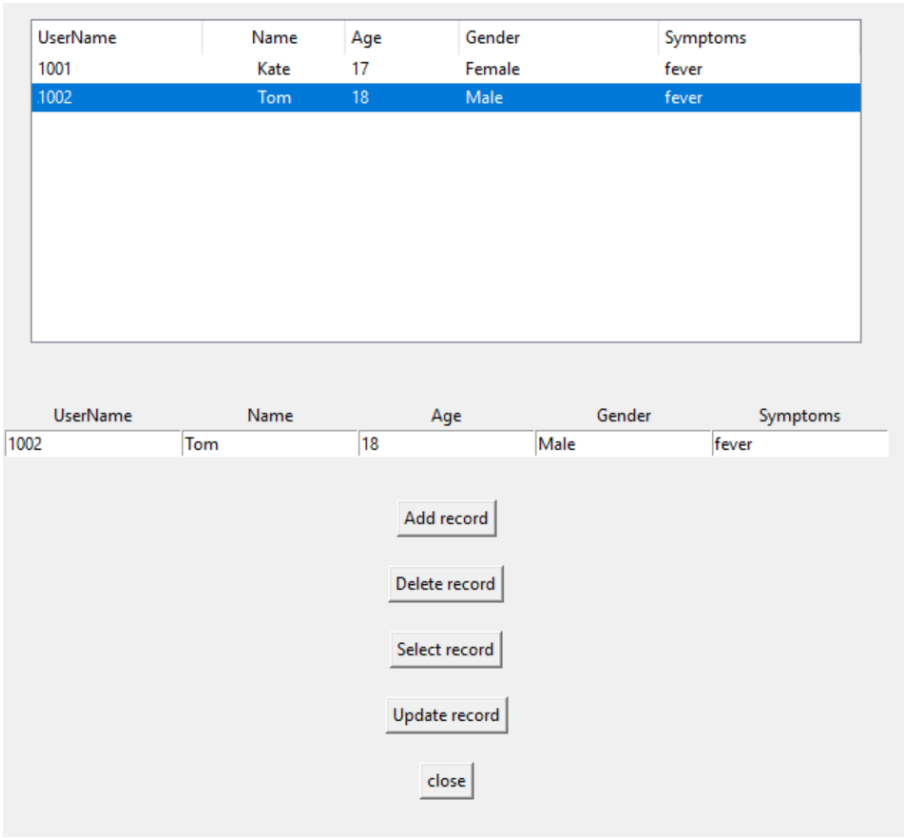


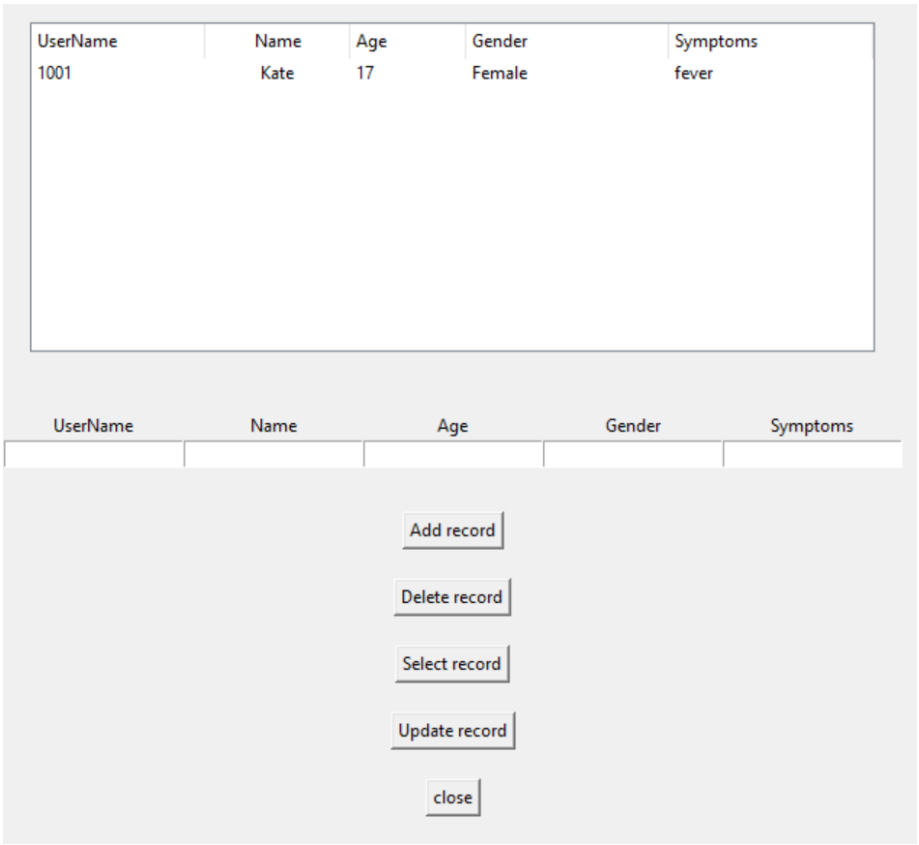
Updating Record:



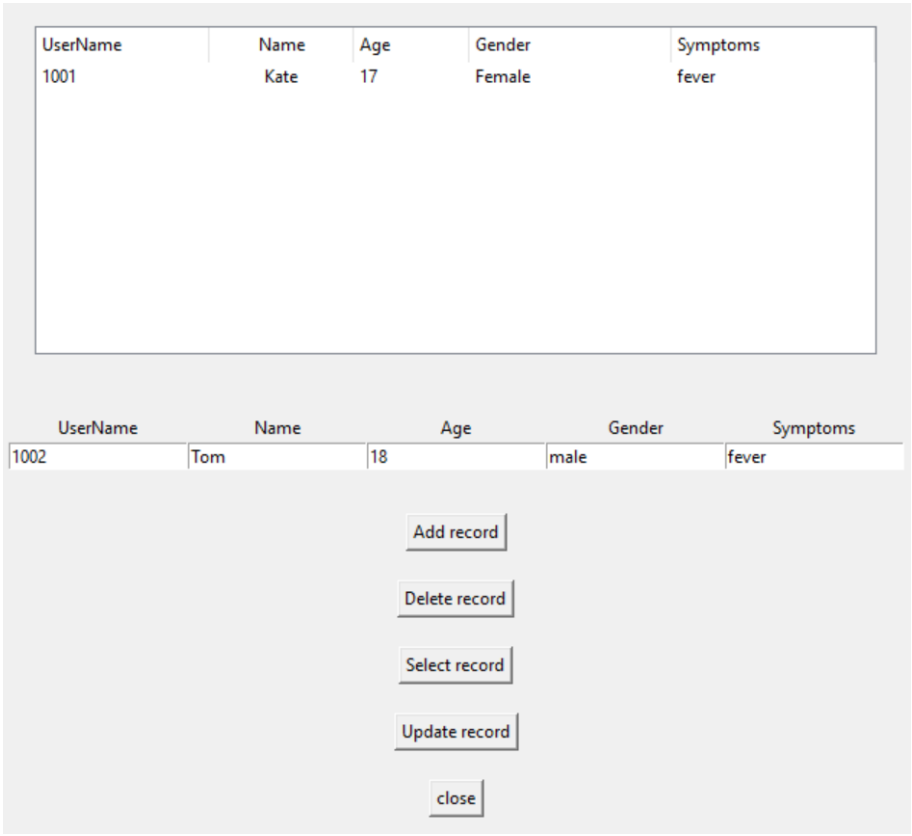


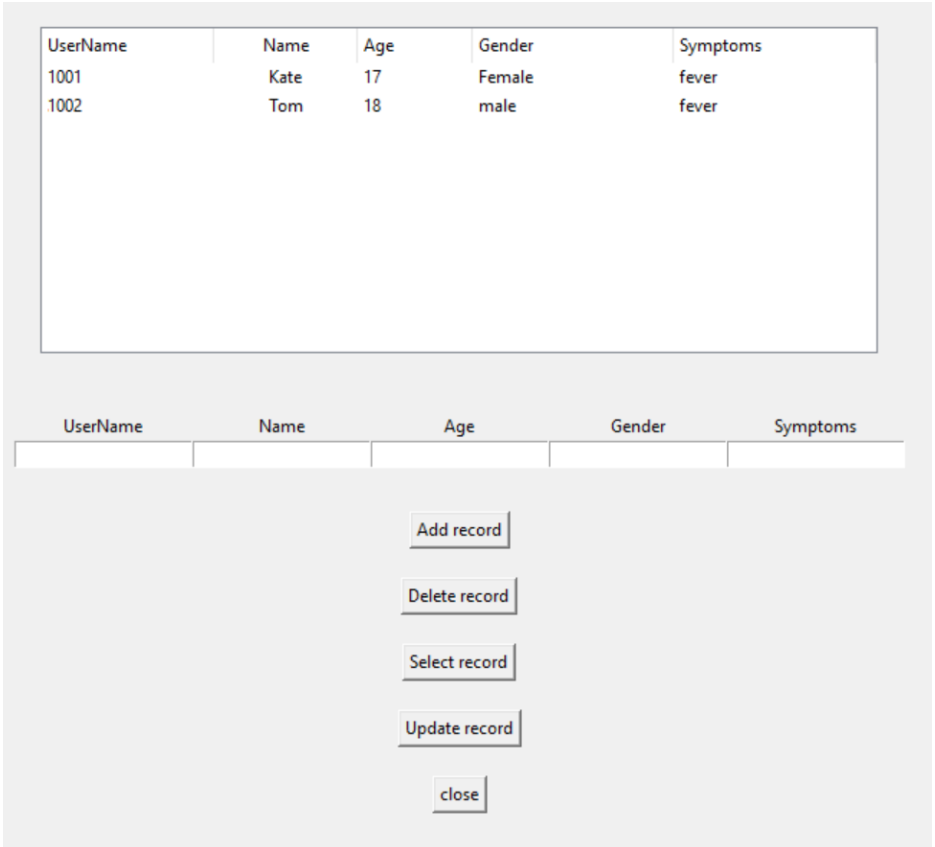
Deleting Record:



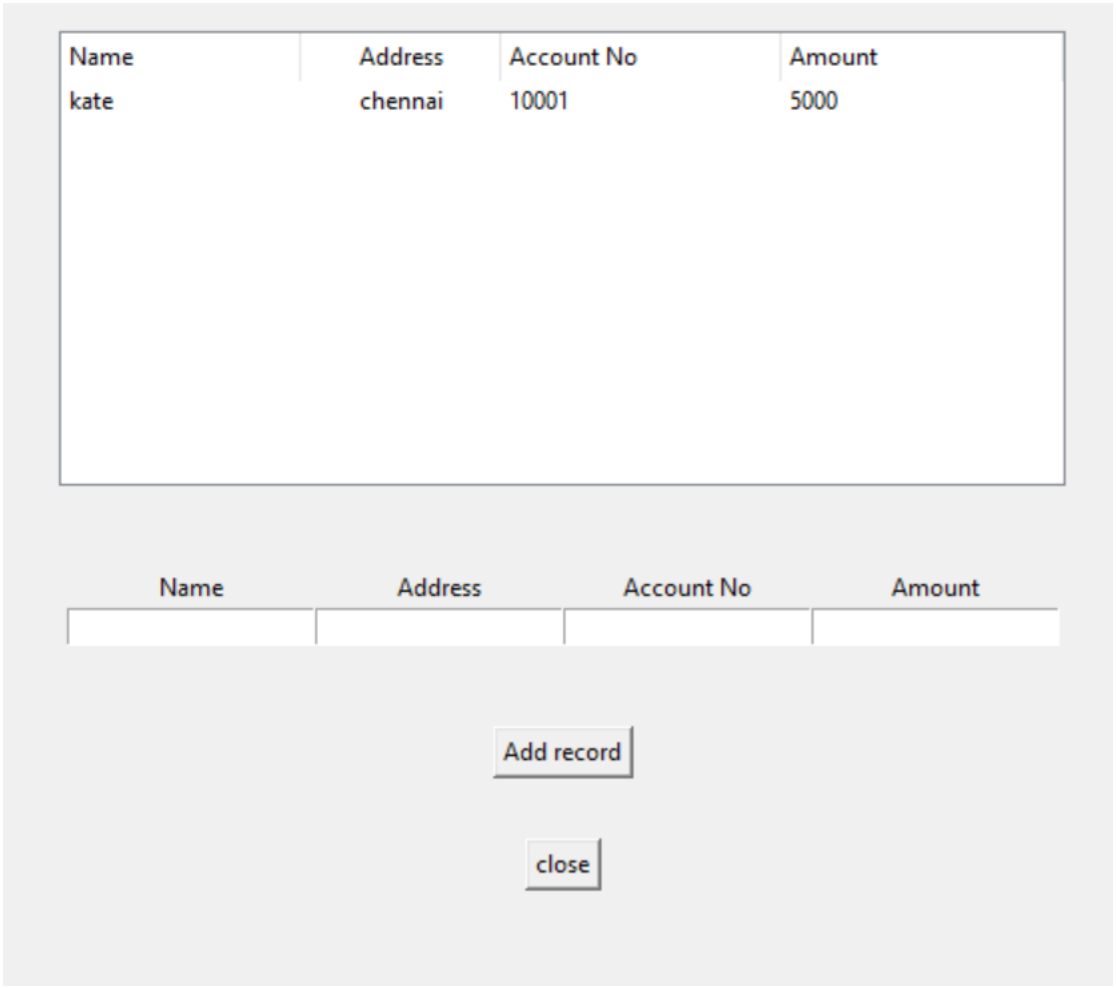


Adding Record:

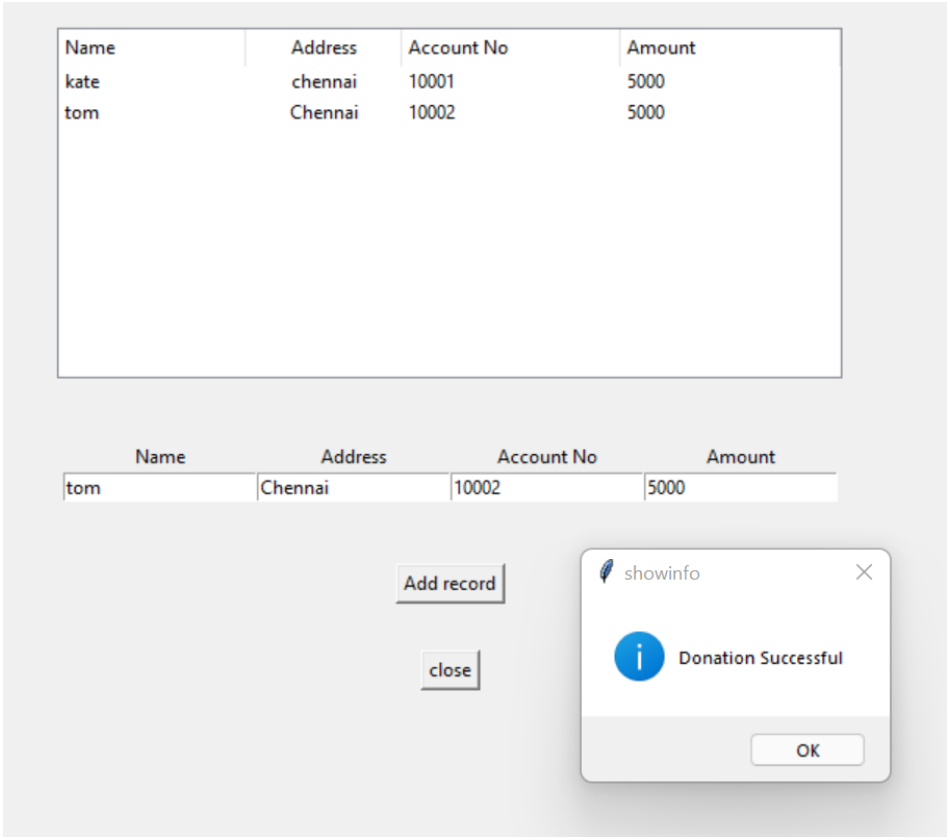




Admin Donations Module:



Adding Record:



SUGGESTED IMPROVEMENTS

- More customization can be done to fulfil the needs of every hospital.
- More modules can be added as per the functionality.
- Provisions to print hard copy of receipt or appointment can be provided.
- Provisions to meet the doctors online can be provided.

BIBLIOGRAPHY

- Online Python Documentation – for python command syntax
- Textbooks - Class XI AND XII – Informatics Practices NECRT

Computer Science with Python by Sumitha Arora

- https://youtu.be/YXPyB4XeYLA - for tkinter module syntax
- https://www.tutorialsteacher.com/python/create-gui-using-tkinter-python

- https://www.geeksforgeeks.org/python-tkinter-tutorial/
- https://youtu.be/n0usdtoU5cE
- https://youtu.be/rtR5wHXPKZ4
- https://youtu.be/YTqDYmfccQU