```
# Mini Project
import tkinter
import tkinter.ttk
import tkinter.messagebox
import sqlite3


class Database:
    def __init__(self):
        self.dbConnection = sqlite3.connect("patientdb.db")
        self.dbCursor = self.dbConnection.cursor()
        self.dbCursor.execute(
            "CREATE TABLE IF NOT EXISTS patient_table (id PRIMARYKEY text, firstname text, lastname text,
dateOfBirth text, monthOfBirth text, yearOfBirth text, gender text, address text, contactNumber text,
emailAddress text, bloodType text, history text, doctor text)")

    def __del__(self):
        self.dbCursor.close()
        self.dbConnection.close()

    def Insert(self, id, firstname, lastname, dateOfBirth, monthOfBirth, yearOfBirth, gender, address,
contactNumber, emailAddress, bloodType, history, doctor):
        self.dbCursor.execute("INSERT INTO patient_table VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)", (
        id, firstname, lastname, dateOfBirth, monthOfBirth, yearOfBirth, gender, address, contactNumber,
emailAddress, bloodType, history, doctor))
        self.dbConnection.commit()

    def Update(self, firstname, lastname, dateOfBirth, monthOfBirth, yearOfBirth, gender, address,
contactNumber, emailAddress, bloodType, history, doctor, id):
        self.dbCursor.execute(
            "UPDATE patient_table SET firstname = ?, lastname = ?, dateOfBirth = ?, monthOfBirth = ?,
yearOfBirth = ?, gender = ?, address = ?, contactNumber = ?, emailAddress = ?, bloodType = ?, history =
?, doctor = ? WHERE id = ?",
            (firstname, lastname, dateOfBirth, monthOfBirth, yearOfBirth, gender, address, contactNumber,
emailAddress, bloodType, history, doctor, id))
        self.dbConnection.commit()

    def Search(self, id):
        self.dbCursor.execute("SELECT * FROM patient_table WHERE id = ?", (id,))
        searchResults = self.dbCursor.fetchall()
        return searchResults

    def Delete(self, id):
        self.dbCursor.execute("DELETE FROM patient_table WHERE id = ?", (id,))
        tkinter.messagebox.showinfo("Deleted data", "Successfully Deleted the Patient data in the
database")
        self.dbConnection.commit()
```

```
    def Display(self):
        self.dbCursor.execute("SELECT * FROM patient_table")
        records = self.dbCursor.fetchall()
        return records


class Values:
    def Validate(self, id, firstname, lastname, contactNumber, emailAdress):
        if not (id.isdigit() and (len(id) == 5 )):
            return "id"
        elif not (firstname.isalpha()):
            return "firstname"
        elif not (lastname.isalpha()):
            return "lastname"
        elif not (contactNumber.isdigit() and (len(contactNumber) == 9)):
            return "contactNumber"
        elif not (emailAdress.count("@") == 1 and emailAdress.count(".") > 0):
            return "emailAddress"
        else:
            return "SUCCESS"


class InsertWindow:
    def __init__(self):
        self.window = tkinter.Tk()
        self.window.wm_title("Insert Patient Data ")
        bg_color = "Blue"
        fg_color = "white"


        self.id = tkinter.StringVar()
        self.firstname = tkinter.StringVar()
        self.lastname = tkinter.StringVar()
        self.address = tkinter.StringVar()
        self.contactNumber = tkinter.StringVar()
        self.emailAddress = tkinter.StringVar()
        self.history = tkinter.StringVar()
        self.doctor = tkinter.StringVar()

        self.genderType = ["Male", "Female", "Transgender", "Other"]
        self.dateType = list(range(1, 32))
        self.monthType = ["January", "February", "March", "April", "May", "June", "July", "August",
"September",
                          "October", "November", "December"]
        self.yearType = list(range(1900, 2020))
        self.bloodListType = ["A+", "A-", "B+", "B-", "O+", "O-", "AB+", "AB-"]

        # Labels
```

```
    tkinter.Label(self.window, fg=fg_color, bg=bg_color, text="Patient Id", font=("times new
roman",10,"bold"), width=25).grid(pady=5, column=1, row=1)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, text="Patient First Name", font=("times new
roman",10,"bold"), width=25).grid(pady=5, column=1, row=2)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new roman",10,"bold"),
text="Patient Last Name", width=25).grid(pady=5, column=1, row=3)
    tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman",10,"bold"),
text="Date of Birth", width=25).grid(pady=5, column=1, row=4)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Month of Birth", width=25).grid(pady=5, column=1, row=5)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Year of Birth", width=25).grid(pady=5, column=1, row=6)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Patient Gender", width=25).grid(pady=5, column=1, row=7)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Patient Address", width=25).grid(pady=5, column=1, row=8)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Patient Contact Number", width=25).grid(pady=5, column=1, row=9)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Patient Email Address", width=25).grid(pady=5, column=1, row=10)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Patient Blood Type", width=25).grid(pady=5, column=1, row=11)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="History of Patient", width=25).grid(pady=5, column=1, row=12)
    tkinter.Label(self.window,  fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"),text="Name of Doctor", width=25).grid(pady=5, column=1, row=13)


    self.idEntry = tkinter.Entry(self.window, width=25, textvariable=self.id)
    self.firstnameEntry = tkinter.Entry(self.window, width=25, textvariable=self.firstname)
    self.lastnameEntry = tkinter.Entry(self.window, width=25, textvariable=self.lastname)
    self.addressEntry = tkinter.Entry(self.window, width=25, textvariable=self.address)
    self.contactNumberEntry = tkinter.Entry(self.window, width=25, textvariable=self.contactNumber)
    self.emailAddressEntry = tkinter.Entry(self.window, width=25, textvariable=self.emailAddress)
    self.historyEntry = tkinter.Entry(self.window, width=25, textvariable=self.history)
    self.doctorEntry = tkinter.Entry(self.window, width=25, textvariable=self.doctor)

    self.idEntry.grid(pady=5, column=3, row=1)
    self.firstnameEntry.grid(pady=5, column=3, row=2)
    self.lastnameEntry.grid(pady=5, column=3, row=3)
    self.addressEntry.grid(pady=5, column=3, row=8)
    self.contactNumberEntry.grid(pady=5, column=3, row=9)
    self.emailAddressEntry.grid(pady=5, column=3, row=10)
    self.historyEntry.grid(pady=5, column=3, row=12)
    self.doctorEntry.grid(pady=5, column=3, row=13)

    # Combobox widgets
    self.dateOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.dateType, width=25)
```

```
        self.monthOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.monthType, width=25)
        self.yearOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.yearType, width=25)
        self.genderBox = tkinter.ttk.Combobox(self.window, values=self.genderType, width=25)
        self.bloodListBox = tkinter.ttk.Combobox(self.window, values=self.bloodListType, width=25)

        self.dateOfBirthBox.grid(pady=5, column=3, row=4)
        self.monthOfBirthBox.grid(pady=5, column=3, row=5)
        self.yearOfBirthBox.grid(pady=5, column=3, row=6)
        self.genderBox.grid(pady=5, column=3, row=7)
        self.bloodListBox.grid(pady=5, column=3, row=11)

        # Button widgets
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"), text="Insert", command=self.Insert).grid(pady=15, padx=5, column=1,
                                                            row=14)
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"), text="Reset", command=self.Reset).grid(pady=15, padx=5, column=2, row=14)
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new
roman",10,"bold"), text="Close", command=self.window.destroy).grid(pady=15, padx=5, column=3,
                                                            row=14)

        self.window.mainloop()

    def Insert(self):
        self.values = Values()
        self.database = Database()
        self.test = self.values.Validate(self.idEntry.get(), self.firstnameEntry.get(), self.lastnameEntry.get(),
                                         self.contactNumberEntry.get(), self.emailAddressEntry.get())
        if (self.test == "SUCCESS"):
            self.database.Insert(self.idEntry.get(), self.firstnameEntry.get(), self.lastnameEntry.get(),
self.dateOfBirthBox.get(),
                                 self.monthOfBirthBox.get(), self.yearOfBirthBox.get(), self.genderBox.get(),
self.addressEntry.get(),
                                 self.contactNumberEntry.get(), self.emailAddressEntry.get(), self.bloodListBox.get(),
                                 self.historyEntry.get(), self.doctorEntry.get())
            tkinter.messagebox.showinfo("Inserted data", "Successfully inserted the above data in the
database")
        else:
            self.valueErrorMessage = "Invalid input in field " + self.test
            tkinter.messagebox.showerror("Value Error", self.valueErrorMessage)

    def Reset(self):
        self.idEntry.delete(0, tkinter.END)
        self.firstnameEntry.delete(0, tkinter.END)
        self.lastnameEntry.delete(0, tkinter.END)
        self.dateOfBirthBox.set("")
        self.monthOfBirthBox.set("")
        self.yearOfBirthBox.set("")
```

```
        self.genderBox.set("")
        self.addressEntry.delete(0, tkinter.END)
        self.contactNumberEntry.delete(0, tkinter.END)
        self.emailAddressEntry.delete(0, tkinter.END)
        self.bloodListBox.set("")
        self.historyEntry.delete(0, tkinter.END)
        self.doctorEntry.delete(0, tkinter.END)


class UpdateWindow:
    def __init__(self, id):
        self.window = tkinter.Tk()
        self.window.wm_title("Update data")
        bg_color = "Blue"
        fg_color = "white"

        # Initializing all the variables
        self.id = id

        self.firstname = tkinter.StringVar()
        self.lastname = tkinter.StringVar()
        self.address = tkinter.StringVar()
        self.contactNumber = tkinter.StringVar()
        self.emailAddress = tkinter.StringVar()
        self.history = tkinter.StringVar()
        self.doctor = tkinter.StringVar()

        self.genderType = ["Male", "Female", "Transgender", "Other"]
        self.dateType = list(range(1, 32))
        self.monthType = ["January", "February", "March", "April", "May", "June", "July", "August",
"September",
                          "October", "November", "December"]
        self.yearType = list(range(1900, 2020))
        self.bloodListType = ["A+", "A-", "B+", "B-", "O+", "O-", "AB+", "AB-"]

        # Labels
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, text="Patient Id", font=("times new roman",
10, "bold"),
                      width=25).grid(pady=5, column=1, row=1)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, text="Patient First Name",
                      font=("times new roman", 10, "bold"), width=25).grid(pady=5, column=1, row=2)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                      text="Patient Last Name", width=25).grid(pady=5, column=1, row=3)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
text="Date of Birth",
                      width=25).grid(pady=5, column=1, row=4)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                      text="Month of Birth", width=25).grid(pady=5, column=1, row=5)
```

```
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
text="Year of Birth",
                width=25).grid(pady=5, column=1, row=6)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Patient Gender", width=25).grid(pady=5, column=1, row=7)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Patient Address", width=25).grid(pady=5, column=1, row=8)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Patient Contact Number", width=25).grid(pady=5, column=1, row=9)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Patient Email Address", width=25).grid(pady=5, column=1, row=10)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Patient Blood Type", width=25).grid(pady=5, column=1, row=11)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="History of Patient", width=25).grid(pady=5, column=1, row=12)
        tkinter.Label(self.window, fg=fg_color, bg=bg_color, font=("times new roman", 10, "bold"),
                text="Name of Doctor", width=25).grid(pady=5, column=1, row=13)

        # Set previous values
        self.database = Database()
        self.searchResults = self.database.Search(id)

        tkinter.Label(self.window, text=self.searchResults[0][1], width=25).grid(pady=5, column=2, row=2)
        tkinter.Label(self.window, text=self.searchResults[0][2], width=25).grid(pady=5, column=2, row=3)
        tkinter.Label(self.window, text=self.searchResults[0][3], width=25).grid(pady=5, column=2, row=4)
        tkinter.Label(self.window, text=self.searchResults[0][4], width=25).grid(pady=5, column=2, row=5)
        tkinter.Label(self.window, text=self.searchResults[0][5], width=25).grid(pady=5, column=2, row=6)
        tkinter.Label(self.window, text=self.searchResults[0][6], width=25).grid(pady=5, column=2, row=7)
        tkinter.Label(self.window, text=self.searchResults[0][7], width=25).grid(pady=5, column=2, row=8)
        tkinter.Label(self.window, text=self.searchResults[0][8], width=25).grid(pady=5, column=2, row=9)
        tkinter.Label(self.window, text=self.searchResults[0][9], width=25).grid(pady=5, column=2, row=10)
        tkinter.Label(self.window, text=self.searchResults[0][10], width=25).grid(pady=5, column=2,
row=11)
        tkinter.Label(self.window, text=self.searchResults[0][11], width=25).grid(pady=5, column=2,
row=12)
        tkinter.Label(self.window, text=self.searchResults[0][12], width=25).grid(pady=5, column=2,
row=13)


        self.idEntry = tkinter.Entry(self.window, width=25, textvariable=self.id)
        self.firstnameEntry = tkinter.Entry(self.window, width=25, textvariable=self.firstname)
        self.lastnameEntry = tkinter.Entry(self.window, width=25, textvariable=self.lastname)
        self.addressEntry = tkinter.Entry(self.window, width=25, textvariable=self.address)
        self.contactNumberEntry = tkinter.Entry(self.window, width=25, textvariable=self.contactNumber)
        self.emailAddressEntry = tkinter.Entry(self.window, width=25, textvariable=self.emailAddress)
        self.historyEntry = tkinter.Entry(self.window, width=25, textvariable=self.history)
        self.doctorEntry = tkinter.Entry(self.window, width=25, textvariable=self.doctor)
```

```
        self.idEntry.grid(pady=5, column=3, row=1)
        self.firstnameEntry.grid(pady=5, column=3, row=2)
        self.lastnameEntry.grid(pady=5, column=3, row=3)
        self.addressEntry.grid(pady=5, column=3, row=8)
        self.contactNumberEntry.grid(pady=5, column=3, row=9)
        self.emailAddressEntry.grid(pady=5, column=3, row=10)
        self.historyEntry.grid(pady=5, column=3, row=12)
        self.doctorEntry.grid(pady=5, column=3, row=13)

        # Combobox
        self.dateOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.dateType, width=20)
        self.monthOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.monthType, width=20)
        self.yearOfBirthBox = tkinter.ttk.Combobox(self.window, values=self.yearType, width=20)
        self.genderBox = tkinter.ttk.Combobox(self.window, values=self.genderType, width=20)
        self.bloodListBox = tkinter.ttk.Combobox(self.window, values=self.bloodListType, width=20)

        self.dateOfBirthBox.grid(pady=5, column=3, row=4)
        self.monthOfBirthBox.grid(pady=5, column=3, row=5)
        self.yearOfBirthBox.grid(pady=5, column=3, row=6)
        self.genderBox.grid(pady=5, column=3, row=7)
        self.bloodListBox.grid(pady=5, column=3, row=11)

        # Button
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new roman", 10,
"bold"),
                       text="Update", command=self.Update).grid(pady=15, padx=5, column=1,
                                                                row=14)
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new roman", 10,
"bold"),
                       text="Reset", command=self.Reset).grid(pady=15, padx=5, column=2, row=14)
        tkinter.Button(self.window, width=10, fg=fg_color, bg=bg_color, font=("times new roman", 10,
"bold"),
                       text="Close", command=self.window.destroy).grid(pady=15, padx=5, column=3,
                                                                       row=14)

        self.window.mainloop()

    def Update(self):
        self.database = Database()
        self.database.Update(self.firstnameEntry.get(), self.lastnameEntry.get(), self.dateOfBirthBox.get(),
self.monthOfBirthBox.get(),
                             self.yearOfBirthBox.get(), self.genderBox.get(), self.addressEntry.get(),
self.contactNumberEntry.get(),
                             self.emailAddressEntry.get(), self.bloodListBox.get(), self.historyEntry.get(),
                             self.doctorEntry.get(), self.id)
        tkinter.messagebox.showinfo("Updated data", "Successfully updated the above data in the
database")
```

```
    def Reset(self):
        self.idEntry.delete(0, tkinter.END)
        self.firstnameEntry.delete(0, tkinter.END)
        self.lastnameEntry.delete(0, tkinter.END)
        self.dateOfBirthBox.set("")
        self.monthOfBirthBox.set("")
        self.yearOfBirthBox.set("")
        self.genderBox.set("")
        self.addressEntry.delete(0, tkinter.END)
        self.contactNumberEntry.delete(0, tkinter.END)
        self.emailAddressEntry.delete(0, tkinter.END)
        self.bloodListBox.set("")
        self.historyEntry.delete(0, tkinter.END)
        self.doctorEntry.delete(0, tkinter.END)


class DatabaseView:
    def __init__(self, data):
        self.databaseViewWindow = tkinter.Tk()
        self.databaseViewWindow.wm_title("Database View")

        # Label widgets
        tkinter.Label(self.databaseViewWindow, text="Database View Window", width=25).grid(pady=5,
column=1, row=1)

        self.databaseView = tkinter.ttk.Treeview(self.databaseViewWindow)
        self.databaseView.grid(pady=5, column=1, row=2)
        self.databaseView["show"] = "headings"
        self.databaseView["columns"] = (
        "id", "firstname", "lastname", "dateOfBirth", "monthOfBirth", "yearOfBirth", "gender", "address",
"contactNumber", "emailAddress", "bloodType", "history",
        "doctor")

        # Treeview column headings
        self.databaseView.heading("id", text="Patient ID")
        self.databaseView.heading("firstname", text="First Name")
        self.databaseView.heading("lastname", text="Last Name")
        self.databaseView.heading("dateOfBirth", text="Date of Birth")
        self.databaseView.heading("monthOfBirth", text="Month of Birth")
        self.databaseView.heading("yearOfBirth", text="Year of Birth")
        self.databaseView.heading("gender", text="Gender")
        self.databaseView.heading("address", text="Home Address")
        self.databaseView.heading("contactNumber", text="Contact Number")
        self.databaseView.heading("emailAddress", text="Email Address")
        self.databaseView.heading("bloodType", text="Blood Type")
        self.databaseView.heading("history", text="History")
        self.databaseView.heading("doctor", text="Doctor")
```

```
        # Treeview columns
        self.databaseView.column("id", width=100)
        self.databaseView.column("firstname", width=100)
        self.databaseView.column("lastname", width=100)
        self.databaseView.column("dateOfBirth", width=100)
        self.databaseView.column("monthOfBirth", width=100)
        self.databaseView.column("yearOfBirth", width=100)
        self.databaseView.column("gender", width=100)
        self.databaseView.column("address", width=200)
        self.databaseView.column("contactNumber", width=100)
        self.databaseView.column("emailAddress", width=200)
        self.databaseView.column("bloodType", width=100)
        self.databaseView.column("history", width=100)
        self.databaseView.column("doctor", width=100)

        for record in data:
            self.databaseView.insert('', 'end', values=(record))

        self.databaseViewWindow.mainloop()


class SearchDeleteWindow:
    def __init__(self, task):
        window = tkinter.Tk()
        window.wm_title(task + " data")

        # Initializing all the variables
        self.id = tkinter.StringVar()
        self.firstname = tkinter.StringVar()
        self.lastname = tkinter.StringVar()
        self.heading = "Please enter Patient ID to " + task

        # Labels
        tkinter.Label(window, text=self.heading, width=50).grid(pady=20, row=1)
        tkinter.Label(window, text="Patient ID", width=10).grid(pady=5, row=2)

        # Entry widgets
        self.idEntry = tkinter.Entry(window, width=5, textvariable=self.id)

        self.idEntry.grid(pady=5, row=3)

        # Button widgets
        if (task == "Search"):
            tkinter.Button(window, width=20, text=task, command=self.Search).grid(pady=15, padx=5,
column=1, row=14)
        elif (task == "Delete"):
            tkinter.Button(window, width=20, text=task, command=self.Delete).grid(pady=15, padx=5,
column=1, row=14)
```

```
    def Search(self):
        self.database = Database()
        self.data = self.database.Search(self.idEntry.get())
        self.databaseView = DatabaseView(self.data)

    def Delete(self):
        self.database = Database()
        self.database.Delete(self.idEntry.get())


class HomePage:
    def __init__(self):
        self.homePageWindow = tkinter.Tk()
        self.homePageWindow.wm_title("Patient Information Home Page")
        bg_color = "blue"
        fg_color = "white"
        lbl_color = 'GREEN'
        tkinter.Label(self.homePageWindow, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color, text="Home
Page", font=("times new roman",20,"bold"), width=30).grid(pady=20, column=1, row=1)

        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Insert", font=("times new roman",15,"bold"), command=self.Insert).grid(pady=15, column=1,
row=2)
        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Update", font=("times new roman",15,"bold"), command=self.Update).grid(pady=15, column=1,
row=3)
        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Search", font=("times new roman",15,"bold"), command=self.Search).grid(pady=15, column=1,
row=4)
        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Delete", font=("times new roman",15,"bold"), command=self.Delete).grid(pady=15, column=1,
row=5)
        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Display", font=("times new roman",15,"bold"), command=self.Display).grid(pady=15, column=1,
                                                                    row=6)
        tkinter.Button(self.homePageWindow, width=20, relief=tkinter.GROOVE, fg=fg_color, bg=bg_color,
text="Exit", font=("times new roman",15,"bold"),
command=self.homePageWindow.destroy).grid(pady=15,
                                                                                column=1,
                                                                                row=7)

        self.homePageWindow.mainloop()

    def Insert(self):
        self.insertWindow = InsertWindow()

    def Update(self):
```

```
        self.updateIDWindow = tkinter.Tk()
        self.updateIDWindow.wm_title("Update data")

        # Initializing all the variables
        self.id = tkinter.StringVar()

        # Label
        tkinter.Label(self.updateIDWindow, text="Enter the ID to update", width=50).grid(pady=20, row=1)

        # Entry widgets
        self.idEntry = tkinter.Entry(self.updateIDWindow, width=5, textvariable=self.id)

        self.idEntry.grid(pady=10, row=2)

        # Button widgets
        tkinter.Button(self.updateIDWindow, width=20, text="Update",
command=self.updateID).grid(pady=10, row=3)

        self.updateIDWindow.mainloop()

    def updateID(self):
        self.updateWindow = UpdateWindow(self.idEntry.get())
        self.updateIDWindow.destroy()

    def Search(self):
        self.searchWindow = SearchDeleteWindow("Search")

    def Delete(self):

        self.deleteWindow = SearchDeleteWindow("Delete")


    def Display(self):
        self.database = Database()
        self.data = self.database.Display()
        self.displayWindow = DatabaseView(self.data)


homePage = HomePage()
```