

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
from tkinter import *
import sqlite3
import tkinter.ttk as ttk
import tkinter.messagebox as tkMessageBox

#DEVELOPED BY Mark Arvin

root = Tk()
root.title("BUS")
width = 700
height = 400
screen_width = root.winfo_screenwidth()
screen_height = root.winfo_screenheight()
x = (screen_width/2) - (width/2)
y = (screen_height/2) - (height/2)
root.geometry("%dx%d+%d+%d" % (width, height, x, y))
root.resizable(0, 0)
root.config(bg="#6666ff")

#=====VARIABLES=====

FIRSTNAME = StringVar()
LASTNAME = StringVar()
GENDER = StringVar()
AGE = StringVar()
ADDRESS = StringVar()
CONTACT = StringVar()

#=====METHODS=====
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

def Database():

```
conn = sqlite3.connect("pythontut.db")
```

```
cursor = conn.cursor()
```

```
cursor.execute("CREATE TABLE IF NOT EXISTS `member` (mem_id INTEGER NOT NULL PRIMARY KEY  
AUTOINCREMENT, firstname TEXT, lastname TEXT, gender TEXT, age TEXT, address TEXT, contact TEXT)")
```

```
cursor.execute("SELECT * FROM `member` ORDER BY `lastname` ASC")
```

```
fetch = cursor.fetchall()
```

```
for data in fetch:
```

```
    tree.insert("", 'end', values=(data))
```

```
cursor.close()
```

```
conn.close()
```

def SubmitData():

```
if FIRSTNAME.get() == "" or LASTNAME.get() == "" or GENDER.get() == "" or AGE.get() == "" or  
ADDRESS.get() == "" or CONTACT.get() == "":
```

```
    result = tkMessageBox.showwarning("", 'Please Complete The Required Field', icon="warning")
```

```
else:
```

```
    tree.delete(*tree.get_children())
```

```
    conn = sqlite3.connect("pythontut.db")
```

```
    cursor = conn.cursor()
```

```
    cursor.execute("INSERT INTO `member` (firstname, lastname, gender, age, address, contact)  
VALUES(?, ?, ?, ?, ?, ?)", (str(FIRSTNAME.get()), str(LASTNAME.get()), str(GENDER.get()), int(AGE.get()),  
str(ADDRESS.get()), str(CONTACT.get())))
```

```
    conn.commit()
```

```
    cursor.execute("SELECT * FROM `member` ORDER BY `lastname` ASC")
```

```
    fetch = cursor.fetchall()
```

```
    for data in fetch:
```

```
        tree.insert("", 'end', values=(data))
```

```
    cursor.close()
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
conn.close()

FIRSTNAME.set("")
LASTNAME.set("")
GENDER.set("")
AGE.set("")
ADDRESS.set("")
CONTACT.set("")
```

```
def UpdateData():
    if GENDER.get() == "":
        result = tkMessageBox.showwarning(" 'Please Complete The Required Field', icon='warning')
    else:
        tree.delete(*tree.get_children())

        conn = sqlite3.connect("pythontut.db")

        cursor = conn.cursor()

        cursor.execute("UPDATE `member` SET `firstname` = ?, `lastname` = ?, `gender` = ?, `age` = ?,
`address` = ?, `contact` = ? WHERE `mem_id` = ?", (str(FIRSTNAME.get()), str(LASTNAME.get()),
str(GENDER.get()), str(AGE.get()), str(ADDRESS.get()), str(CONTACT.get()), int(mem_id)))

        conn.commit()

        cursor.execute("SELECT * FROM `member` ORDER BY `lastname` ASC")

        fetch = cursor.fetchall()

        for data in fetch:
            tree.insert("", 'end', values=(data))

        cursor.close()

        conn.close()

        FIRSTNAME.set("")
        LASTNAME.set("")
        GENDER.set("")
        AGE.set("")
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

ADDRESS.set("")

CONTACT.set("")

def OnSelected(event):

 global mem_id, UpdateWindow

 curlItem = tree.focus()

 contents =(tree.item(curlItem))

 selecteditem = contents['values']

 mem_id = selecteditem[0]

 FIRSTNAME.set("")

 LASTNAME.set("")

 GENDER.set("")

 AGE.set("")

 ADDRESS.set("")

 CONTACT.set("")

 FIRSTNAME.set(selecteditem[1])

 LASTNAME.set(selecteditem[2])

 AGE.set(selecteditem[4])

 ADDRESS.set(selecteditem[5])

 CONTACT.set(selecteditem[6])

 UpdateWindow = Toplevel()

 UpdateWindow.title("Contact List")

 width = 400

 height = 300

 screen_width = root.winfo_screenwidth()

 screen_height = root.winfo_screenheight()

 x = ((screen_width/2) + 450) - (width/2)

 y = ((screen_height/2) + 20) - (height/2)

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
UpdateWindow.resizable(0, 0)

UpdateWindow.geometry("%dx%d+%d+%d" % (width, height, x, y))

if 'NewWindow' in globals():

    NewWindow.destroy()


#=====FRAMES=====

FormTitle = Frame(UpdateWindow)

FormTitle.pack(side=TOP)

ContactForm = Frame(UpdateWindow)

ContactForm.pack(side=TOP, pady=10)

RadioGroup = Frame(ContactForm)

Male = Radiobutton(RadioGroup, text="Male", variable=GENDER, value="Male", font=('arial',
14)).pack(side=LEFT)

Female = Radiobutton(RadioGroup, text="Female", variable=GENDER, value="Female", font=('arial',
14)).pack(side=LEFT)


#=====LABELS=====

lbl_title = Label(FormTitle, text="Updating Contacts", font=('arial', 16), bg="orange", width = 300)

lbl_title.pack(fill=X)

lbl_firstname = Label(ContactForm, text="Firstname", font=('arial', 14), bd=5)

lbl_firstname.grid(row=0, sticky=W)

lbl_lastname = Label(ContactForm, text="Lastname", font=('arial', 14), bd=5)

lbl_lastname.grid(row=1, sticky=W)

lbl_gender = Label(ContactForm, text="Gender", font=('arial', 14), bd=5)

lbl_gender.grid(row=2, sticky=W)

lbl_age = Label(ContactForm, text="Age", font=('arial', 14), bd=5)

lbl_age.grid(row=3, sticky=W)

lbl_address = Label(ContactForm, text="Address", font=('arial', 14), bd=5)

lbl_address.grid(row=4, sticky=W)
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
lbl_contact = Label(ContactForm, text="Contact", font=('arial', 14), bd=5)
```

```
lbl_contact.grid(row=5, sticky=W)
```

```
#=====ENTRY=====
```

```
firstname = Entry(ContactForm, textvariable=FIRSTNAME, font=('arial', 14))
```

```
firstname.grid(row=0, column=1)
```

```
lastname = Entry(ContactForm, textvariable=LASTNAME, font=('arial', 14))
```

```
lastname.grid(row=1, column=1)
```

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

```
age = Entry(ContactForm, textvariable=AGE, font=('arial', 14))
```

```
age.grid(row=3, column=1)
```

```
address = Entry(ContactForm, textvariable=ADDRESS, font=('arial', 14))
```

```
address.grid(row=4, column=1)
```

```
contact = Entry(ContactForm, textvariable=CONTACT, font=('arial', 14))
```

```
contact.grid(row=5, column=1)
```

```
#=====BUTTONS=====
```

```
btn_updatecon = Button(ContactForm, text="Update", width=50, command=UpdateData)
```

```
btn_updatecon.grid(row=6, columnspan=2, pady=10)
```

```
#fn1353p
```

```
def DeleteData():
```

```
    if not tree.selection():
```

```
        result = tkMessageBox.showwarning("", 'Please Select Something First!', icon="warning")
```

```
    else:
```

```
        result = tkMessageBox.askquestion("", 'Are you sure you want to delete this record?',  
icon="warning")
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
if result == 'yes':  
    curlItem = tree.focus()  
    contents =(tree.item(curlItem))  
    selecteditem = contents['values']  
    tree.delete(curlItem)  
    conn = sqlite3.connect("pythontut.db")  
    cursor = conn.cursor()  
    cursor.execute("DELETE FROM `member` WHERE `mem_id` = %d" % selecteditem[0])  
    conn.commit()  
    cursor.close()  
    conn.close()
```

```
def AddNewWindow():  
    global NewWindow  
    FIRSTNAME.set("")  
    LASTNAME.set("")  
    GENDER.set("")  
    AGE.set("")  
    ADDRESS.set("")  
    CONTACT.set("")  
    NewWindow = Toplevel()  
    NewWindow.title("BUS ENTRY")  
    width = 400  
    height = 300  
    screen_width = root.winfo_screenwidth()  
    screen_height = root.winfo_screenheight()  
    x = ((screen_width/2) - 455) - (width/2)  
    y = ((screen_height/2) + 20) - (height/2)  
    NewWindow.resizable(0, 0)
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
NewWindow.geometry("%dx%d+%d+%d" % (width, height, x, y))

if 'UpdateWindow' in globals():
    UpdateWindow.destroy()

#=====FRAMES=====

FormTitle = Frame(NewWindow)
FormTitle.pack(side=TOP)

ContactForm = Frame(NewWindow)
ContactForm.pack(side=TOP, pady=10)

RadioGroup = Frame(ContactForm)

Male = Radiobutton(RadioGroup, text="Male", variable=GENDER, value="Male", font=('arial',
14)).pack(side=LEFT)

Female = Radiobutton(RadioGroup, text="Female", variable=GENDER, value="Female", font=('arial',
14)).pack(side=LEFT)

#=====LABELS=====

lbl_title = Label(FormTitle, text="Bus Information list", font=('arial', 16), bg="#66ff66", width = 300)
lbl_title.pack(fill=X)

lbl_firstname = Label(ContactForm, text="Firstname", font=('arial', 14), bd=5)
lbl_firstname.grid(row=0, sticky=W)

lbl_lastname = Label(ContactForm, text="Lastname", font=('arial', 14), bd=5)
lbl_lastname.grid(row=1, sticky=W)

lbl_gender = Label(ContactForm, text="Gender", font=('arial', 14), bd=5)
lbl_gender.grid(row=2, sticky=W)

lbl_age = Label(ContactForm, text="Age", font=('arial', 14), bd=5)
lbl_age.grid(row=3, sticky=W)

lbl_address = Label(ContactForm, text="BUS NO", font=('arial', 14), bd=5)
lbl_address.grid(row=4, sticky=W)

lbl_contact = Label(ContactForm, text="Contact", font=('arial', 14), bd=5)
```


BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
lbl_contact.grid(row=5, sticky=W)
```

```
#=====ENTRY=====
```

```
firstname = Entry(ContactForm, textvariable=FIRSTNAME, font=('arial', 14))
```

```
firstname.grid(row=0, column=1)
```

```
lastname = Entry(ContactForm, textvariable=LASTNAME, font=('arial', 14))
```

```
lastname.grid(row=1, column=1)
```

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

```
age = Entry(ContactForm, textvariable=AGE, font=('arial', 14))
```

```
age.grid(row=3, column=1)
```

```
address = Entry(ContactForm, textvariable=ADDRESS, font=('arial', 14))
```

```
address.grid(row=4, column=1)
```

```
contact = Entry(ContactForm, textvariable=CONTACT, font=('arial', 14))
```

```
contact.grid(row=5, column=1)
```

```
#=====BUTTONS=====
```

```
btn_addcon = Button(ContactForm, text="Save", width=50, command=SubmitData)
```

```
btn_addcon.grid(row=6, columnspan=2, pady=10)
```

```
#=====FRAMES=====
```

```
Top = Frame(root, width=500, bd=1, relief=SOLID)
```

```
Top.pack(side=TOP)
```

```
Mid = Frame(root, width=500, bg="#6666ff")
```

```
Mid.pack(side=TOP)
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
MidLeft = Frame(Mid, width=100)
MidLeft.pack(side=LEFT, pady=10)
MidLeftPadding = Frame(Mid, width=370, bg="#6666ff")
MidLeftPadding.pack(side=LEFT)
MidRight = Frame(Mid, width=100)
MidRight.pack(side=RIGHT, pady=10)
TableMargin = Frame(root, width=500)
TableMargin.pack(side=TOP)

#=====LABELS=====

lbl_title = Label(Top, text=" Bus Management System", font=('arial', 16), width=500)
lbl_title.pack(fill=X)

#=====ENTRY=====

#=====BUTTONS=====

btn_add = Button(MidLeft, text="+ ADD NEW", bg="#66ff66", command=AddNewWindow)
btn_add.pack()

btn_delete = Button(MidRight, text="DELETE", bg="red", command=DeleteData)
btn_delete.pack(side=RIGHT)

#=====TABLES=====

scrollbarx = Scrollbar(TableMargin, orient=HORIZONTAL)
scrollbary = Scrollbar(TableMargin, orient=VERTICAL)

tree = ttk.Treeview(TableMargin, columns=("MemberID", "Firstname", "Lastname", "Gender", "Age",
"Address", "Contact"), height=400, selectmode="extended", yscrollcommand=scrollbary.set,
xscrollcommand=scrollbarx.set)

scrollbary.config(command=tree.yview)
scrollbary.pack(side=RIGHT, fill=Y)
scrollbarx.config(command=tree.xview)
```

BUS MANAGEMENT SYSTEM(BMS) USING IOT

```
scrollbarx.pack(side=BOTTOM, fill=X)

tree.heading('MemberID', text="MemberID", anchor=W)

tree.heading('Firstname', text="Firstname", anchor=W)

tree.heading('Lastname', text="Lastname", anchor=W)

tree.heading('Gender', text="Gender", anchor=W)

tree.heading('Age', text="Age", anchor=W)

tree.heading('Address', text="BUS NO", anchor=W)

tree.heading('Contact', text="Contact", anchor=W)

tree.column('#0', stretch=NO, minwidth=0, width=0)

tree.column('#1', stretch=NO, minwidth=0, width=0)

tree.column('#2', stretch=NO, minwidth=0, width=80)

tree.column('#3', stretch=NO, minwidth=0, width=120)

tree.column('#4', stretch=NO, minwidth=0, width=90)

tree.column('#5', stretch=NO, minwidth=0, width=80)

tree.column('#6', stretch=NO, minwidth=0, width=120)

tree.column('#7', stretch=NO, minwidth=0, width=120)

tree.pack()

tree.bind('<Double-Button-1>', OnSelected)


#=====INITIALIZATION=====

if __name__ == '__main__':

    Database()

    root.mainloop()
```