AISSCE(2020-21)

NEERJA MODI SCHOOL

INFORMATICS PRACTICES PROJECT

TOURS AND TRAVEL AGENCY

SUBMITTED BY:

NAME: Aditya Hari Sharma

CLASS: XII-A

ROLL NO.:

SUBMITTED TO:

Mr. Manish Kumar Sharma

(IP teacher)

Certificate

This is to certify that this project on "Tours and Travel Agency" is the bona fide work of "Aditya Hari Sharma" for the subject INFORMATICS PRACTICES for Class 12th and satisfies all criteria to be submitted as the final project requisition for accomplishment of AISSCE 2020-21 through the Central Board of Secondary Education.

Manish Kumar Sharma

IP Teacher

External

ACKNOWLEDGEMENT

Apart from the efforts of myself, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project. I would like to show my greatest appreciation to Mr. Manish Kumar Sharma. I can't say thank you enough for his tremendous support and help. Without his encouragement and guidance this project would not have materialized.

Index Page

-

Advantages of PYTHON

Easy to Use OO Language

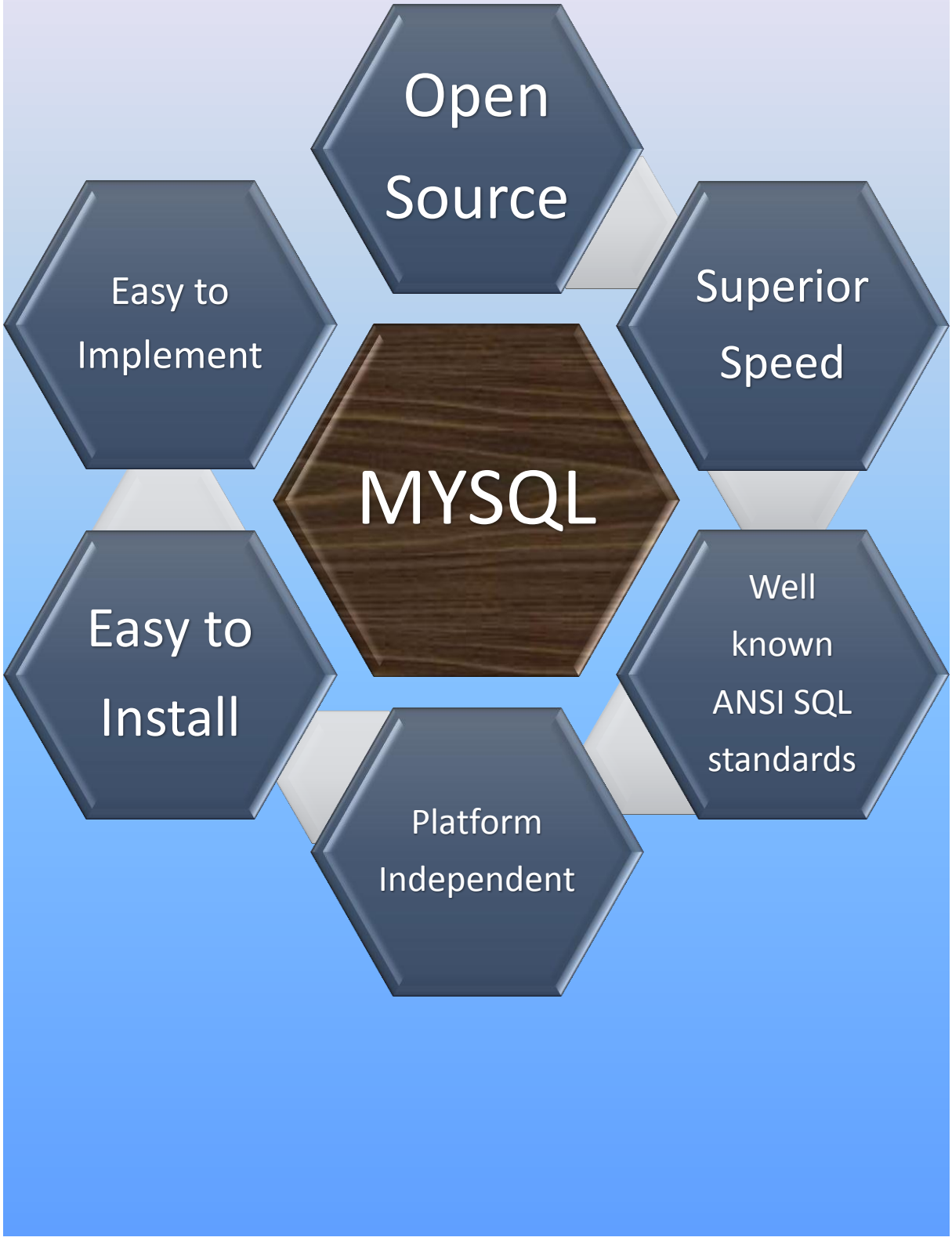
Expressive Language

Interpreted Language

Cross platform Language

Free and Open Source

Advantages of MYSQL

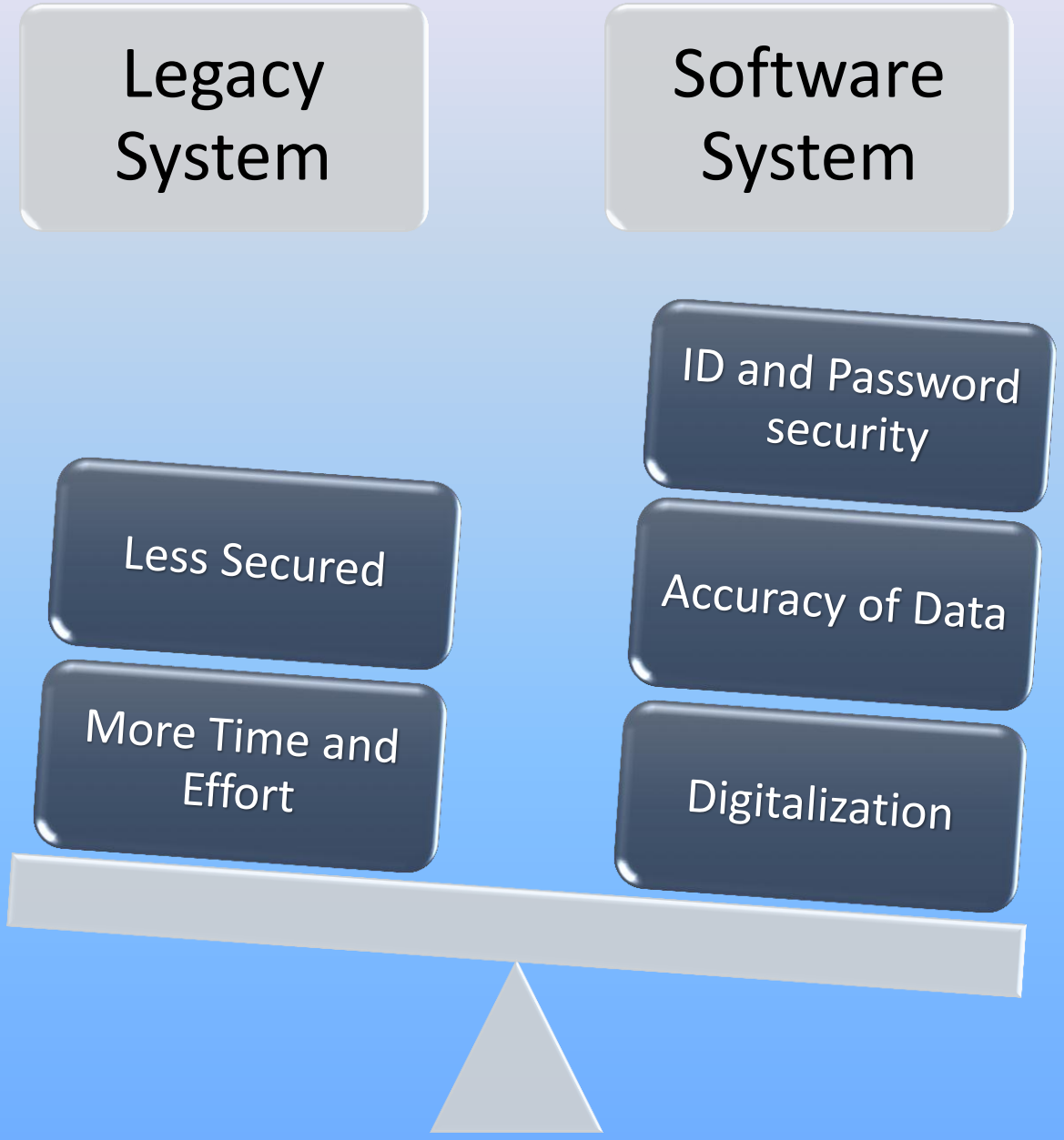


Preface

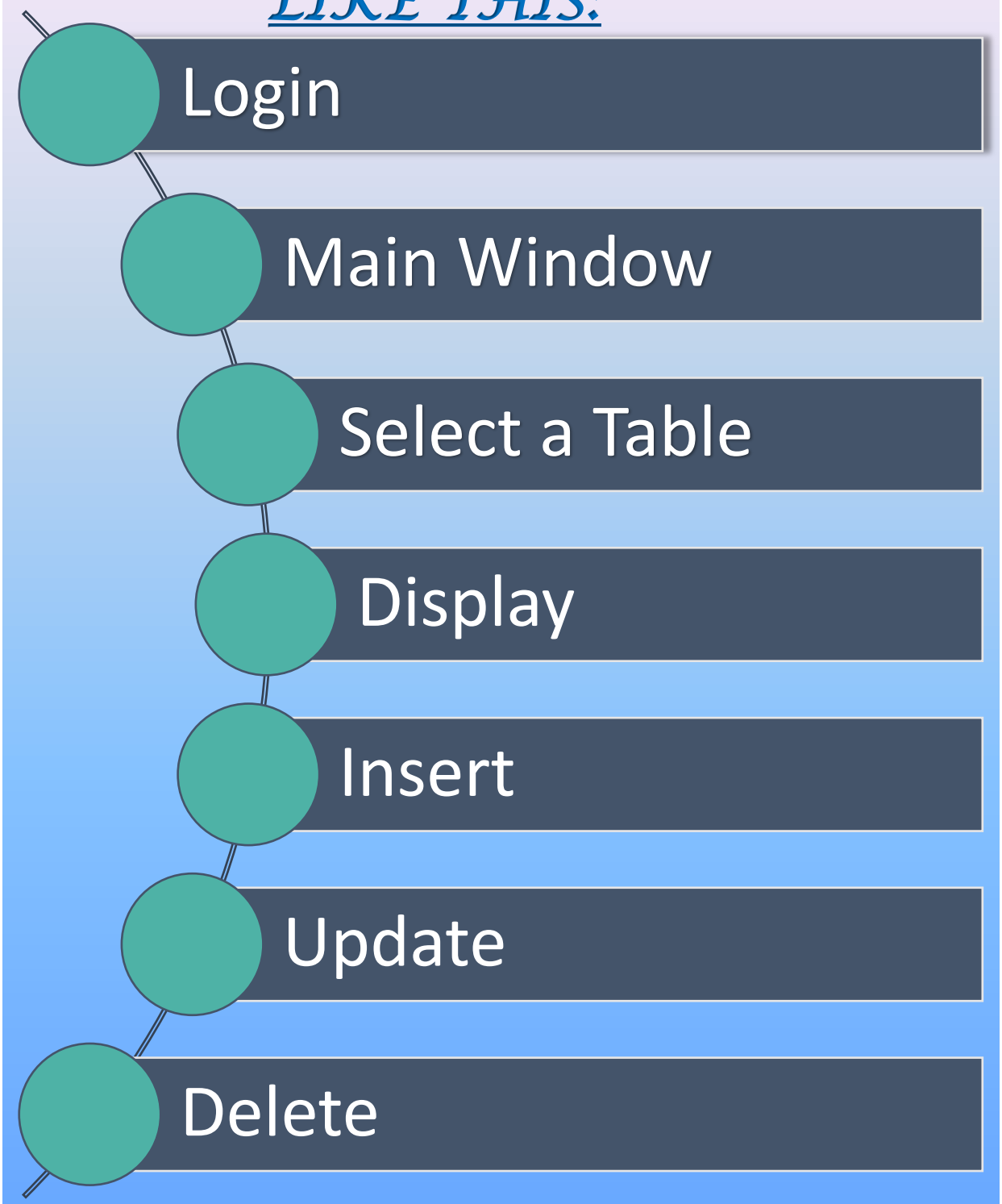
The project titled 'Tours and Travel Agency' is a software for monitoring and controlling the entire transactions of the business regarding details of customers who have purchased holiday packages from the agency , their booking details and the package details. This project is developed in python , which mainly focuses on operations like searching for information regarding a particular booking , or a particular customer's personal details or even details of a package. We can even update , insert and delete the data.

Earlier when transactions were maintained through paperwork , there were many problems like data was unsafe as there is always a possibility of loss of papers. Too much written work is hectic and there is no backup available etc. But with our software , all the transactions are maintained through "Tours and Travel Agency Management System". It is easy to use, has an attractive user interface where data is safe, existing data can be viewed in seconds, changes can be made more easily, backup is available and there is no burden of handwritten information.

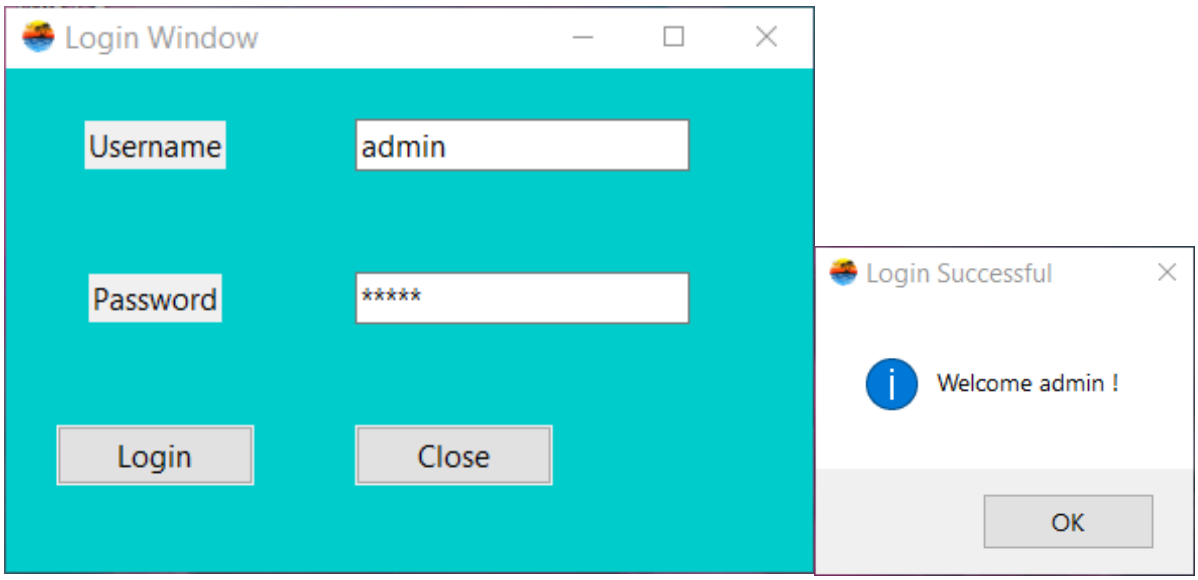
Advantage of Software



THEY CAN PROCEED LIKE THIS:



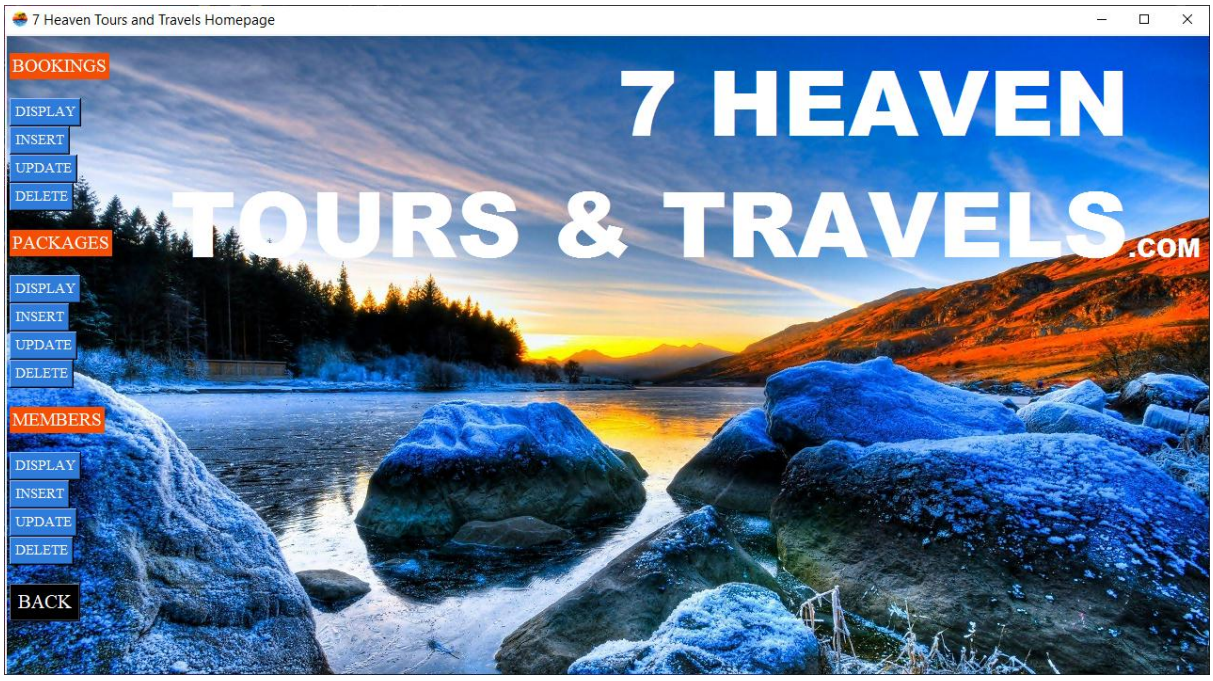
LOGIN WINDOW




```
import tkinter as tk
from tkinter import ttk
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image

win=tk.Tk()
win.title('Login Window')
win.minsize(400,250)
win.configure(bg='#00cccc')
win.iconbitmap(r'logo.ico')

luser=ttk.Label(win,text='Username')
luser.grid(column=0,row=0)

user=tk.StringVar()
tuser=ttk.Entry(win,width=20,textvariable=user)
tuser.grid(column=1,row=0)

lpass=ttk.Label(win,text='Password')
lpass.grid(column=0,row=1)

pas=tk.StringVar()
tpass=ttk.Entry(win,width=20,textvariable=pas,show='*')
tpass.grid(column=1,row=1)
```

```
def _msgBox():
    try:      # exception handling block

     conn = mysql.connector.connect(host='localhost',
                           database='tours',
                           user='root',
                           password='',
                           charset='utf8')  # will create connection object
     cursor = conn.cursor()   # will provide control for table records
     cursor.execute("SELECT * FROM login where user_id='"+user.get()
                    +"' and password='"+pas.get()+"'")
     ls = cursor.fetchall()
     ls=list(ls[0])
     if(ls):
         mBox.showinfo('Login Successful', 'Welcome '+str(ls[0])+ ' !')
         user.set('')    # make control blank
         pas.set('')
         win.quit()    # to make current window invisible
         win.destroy()   # to remove reference of current window from
memory
         import mainwindow       # to open next window

    except Error as e :     # statements to execute in case of some error /
exception
     print("Error while connecting to MySQL\n", e)

    finally:
        print("MySQL connection is closed")
#closing database connection.
def back():
         win.quit()
         win.destroy()

action=ttk.Button(win,text='Login',command=_msgBox)
action.grid(column=0,row=4)
backbt = ttk.Button(win, text="Close",command=back)

backbt.grid(column=1, row=4,sticky=tk.W)

for child in win.winfo_children():

    child.grid_configure(padx=25,pady=25)

win.mainloop()
```

MAIN WINDOW




```
import tkinter as tk

from tkinter import ttk

from tkinter import Menu

from tkinter import messagebox as mBox

from tkinter.font import Font

from PIL import ImageTk, Image


win = tk.Tk()

win.title("7 Heaven Tours and Travels Homepage")

win.configure(background='light blue')

win.iconbitmap(r'logo.ico')
```

```
win.geometry("1280x680+10+10")


bg = ImageTk.PhotoImage(file = "mainwindowbgimg.png")
label1 = tk.Label( win, image = bg)
label1.place(x = 0, y = 0)



def a():

    win.quit()

    win.destroy()

    import insertbooking


def b():

    win.quit()

    win.destroy()

    import select_booking


def c():

    win.quit()

    win.destroy()

    import updatebooking


def d():

    win.quit()
```

```
    win.destroy()

    import deletebooking


def back():
    win.quit()
    win.destroy()
    import login



todord = tk.Label(win,
text="BOOKINGS",font=('Times',15),bg='#f64f04',fg='white')

todord.grid(column=0, row=2,padx=5,sticky=tk.W,pady=20)

todord1 = tk.Button(win,
text="DISPLAY",font=('Times',11),bg='#307cdb',fg='white',command=b)

todord1.grid(column=0, row=3,padx=5,sticky=tk.W)

todord2 = tk.Button(win, text="INSERT",
font=('Times',11),bg='#307cdb',fg='white',command=a)

todord2.grid(column=0, row=4,padx=5,sticky=tk.W)

todord3 = tk.Button(win, text="UPDATE",
font=('Times',11),bg='#307cdb',fg='white',command=c)

todord3.grid(column=0, row=5,padx=5,sticky=tk.W)

todord4 = tk.Button(win,
text="DELETE",font=('Times',11),bg='#307cdb',fg='white',command=d)

todord4.grid(column=0, row=6,padx=5,sticky=tk.W)
```

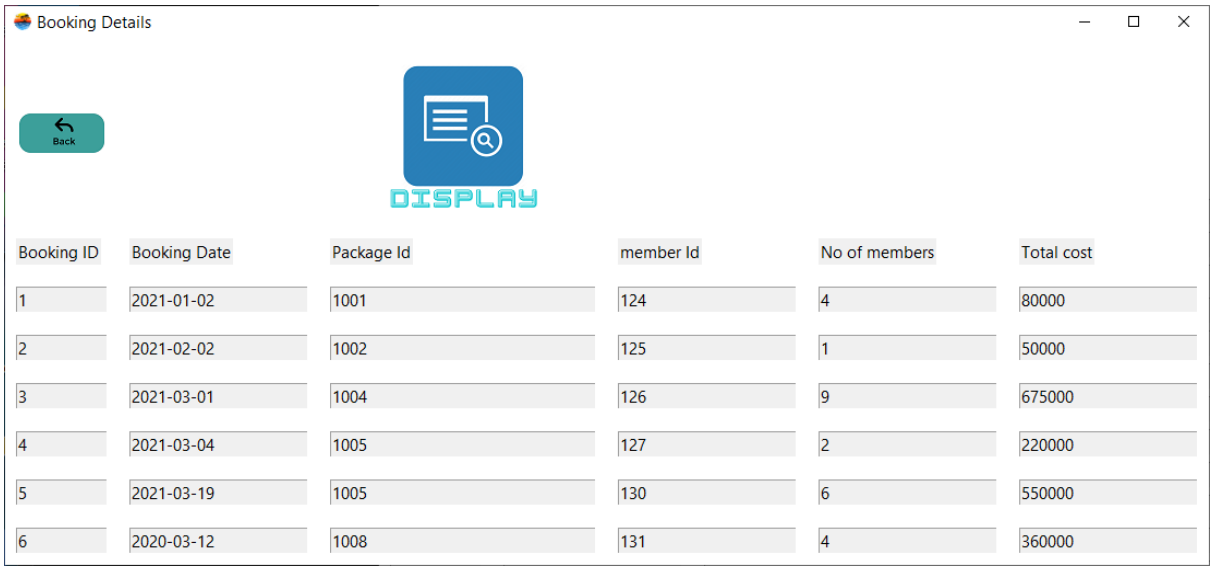
```
def e():
    win.quit()
    win.destroy()
    import insertpackage


def f():
    win.quit()
    win.destroy()
    import select_packages


def g():
    win.quit()
    win.destroy()
    import updatepackages


def h():
    win.quit()
    win.destroy()
    import deletepackages



odrd = tk.Label(win,
text="PACKAGES",font=('Times',15),bg='#f64f04',fg='white')

odrd.grid(column=0, row=7,padx=5,sticky=tk.W,pady=20)
```

```
odrd1 = tk.Button(win,
text="DISPLAY",font=('Times',11),bg='#307cdb',fg='white',command=f)

odrd1.grid(column=0, row=8,padx=5,sticky=tk.W)

odrd2 = tk.Button(win, text="INSERT",
font=('Times',11),bg='#307cdb',fg='white',command=e)

odrd2.grid(column=0, row=9,padx=5,sticky=tk.W)

odrd3 = tk.Button(win, text="UPDATE",
font=('Times',11),bg='#307cdb',fg='white',command=g)

odrd3.grid(column=0, row=10,padx=5,sticky=tk.W)

odrd4 = tk.Button(win,
text="DELETE",font=('Times',11),bg='#307cdb',fg='white',command=h)

odrd4.grid(column=0, row=11,padx=5,sticky=tk.W)




def i():

    win.quit()

    win.destroy()

    import insertmembers


def j():

    win.quit()

    win.destroy()

    import select_members
```

```
def k():
    win.quit()
    win.destroy()
    import updatemembers


def l():
    win.quit()
    win.destroy()
    import deletemembers



todrd = tk.Label(win,
text="MEMBERS",font=('Times',15),bg='#f64f04',fg='white')

todrd.grid(column=0, row=12,padx=5,sticky=tk.W,pady=20)

todrd1 = tk.Button(win,
text="DISPLAY",font=('Times',11),bg='#307cdb',fg='white',command=j)

todrd1.grid(column=0, row=13,padx=5,sticky=tk.W)

todrd2 = tk.Button(win, text="INSERT",
font=('Times',11),bg='#307cdb',fg='white',command=i)

todrd2.grid(column=0, row=14,padx=5,sticky=tk.W)

todrd3 = tk.Button(win, text="UPDATE",
font=('Times',11),bg='#307cdb',fg='white',command=k)

todrd3.grid(column=0, row=15,padx=5,sticky=tk.W)

todrd4 = tk.Button(win,
text="DELETE",font=('Times',11),bg='#307cdb',fg='white',command=l)
```

```
todrd4.grid(column=0, row=16,padx=5,sticky=tk.W)


todrd5 = tk.Button(win,
text="BACK",font=('Times',16),bg='black',fg='white',command=back)

todrd5.grid(column=0, row=17,padx=5,sticky=tk.W,pady=20)


win.mainloop()
```

DISPLAY BOOKING DETAILS

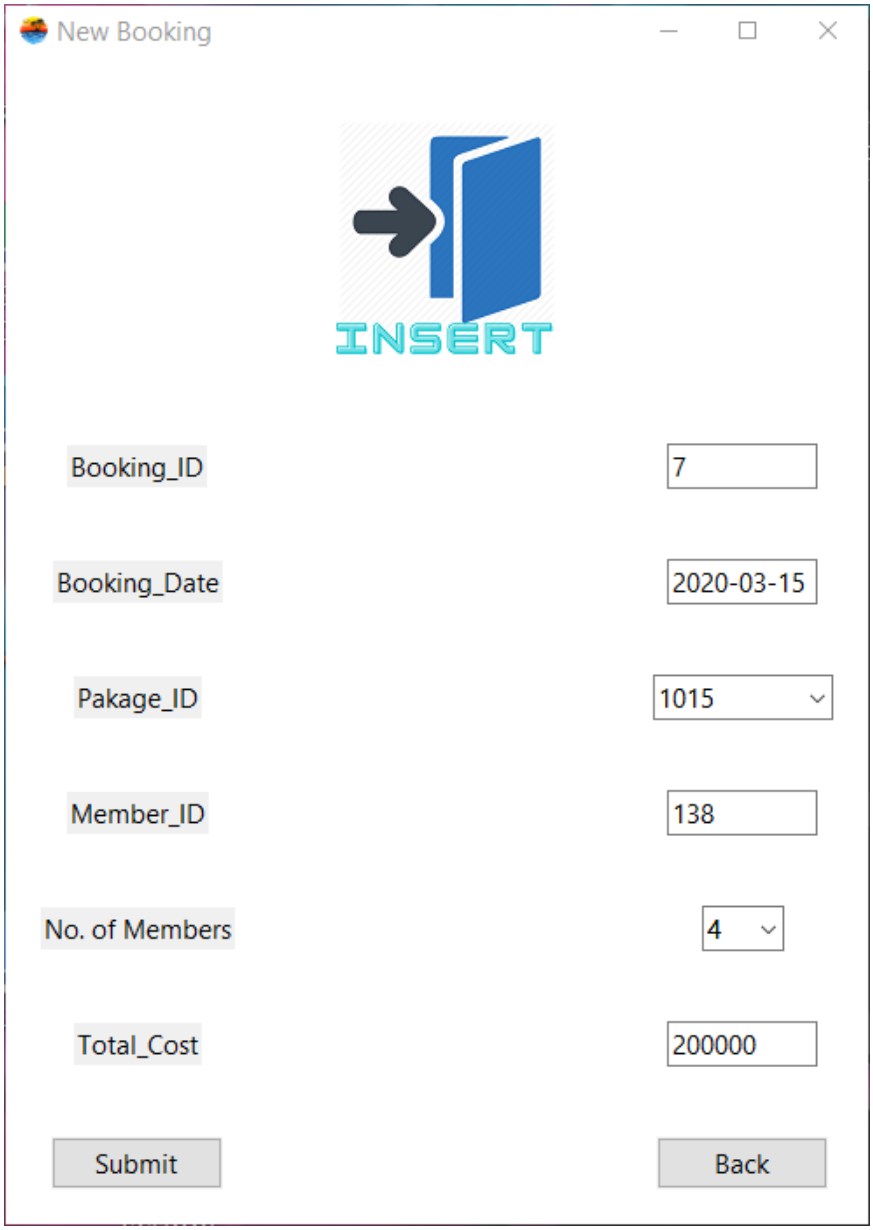


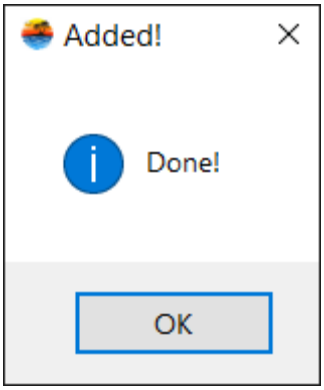

```
import pandas as pd
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image

win = tk.Tk()
win.title("Booking Details")
win.iconbitmap(r'logo.ico')
win.configure(background='white')

bg = ImageTk.PhotoImage(file = "selectimage.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=2,row=0)

def back():
```

```
        win.quit()
        win.destroy()
        import mainwindow


backimg=tk.PhotoImage(file='backbuttonimageresized.png')
submit1 = tk.Button(win, text="Back",
command=back,image=backimg,highlightthickness=0,bd =0)

submit1.grid(column=0, row=0,columnspan=1)

try:

    conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
    cursor = conn.cursor()
    cursor.execute("select * from booking;")
    ls = pd.DataFrame(cursor.fetchall())

    ttk.Label(win, text="Booking ID").grid(column=0, row=1, sticky=tk.W,
columnspan=10)
    ttk.Label(win, text="Booking Date").grid(column=1, row=1,
sticky=tk.W, columnspan=10)
    ttk.Label(win, text="Package Id").grid(column=2, row=1, sticky=tk.W,
columnspan=10)
    ttk.Label(win, text="member Id").grid(column=3, row=1, sticky=tk.W,
columnspan=10)
    ttk.Label(win, text="No of members").grid(column=4, row=1,
sticky=tk.W, columnspan=10)
    ttk.Label(win, text="Total cost").grid(column=5, row=1, sticky=tk.W,
columnspan=10)

    for i in range(0, len(ls.index)):
        for j in range(0, len(ls.columns)):
            b = tk.Entry(win)
            b.insert(0, ls.iloc[i][j])
            b.grid(row=i+2, column=j)
            if(j==0):
                b.config(state = "readonly", width = 10)
            elif(j==2):
                b.config(state = 'readonly', width=30)
            elif(j==4):
                b.config(state = "readonly", width = 20)
            else:
```

```
                b.config(state = "readonly", width = 20)
    conn.commit()
    conn.close()
except Error as e :
    print("Error while connecting to MySQL", e)
finally:
    print("MySQL connection is closed")

for child in win.winfo_children():
    child.grid_configure(padx=10, pady=10)

win.mainloop()
```

INSERT BOOKING DETAILS





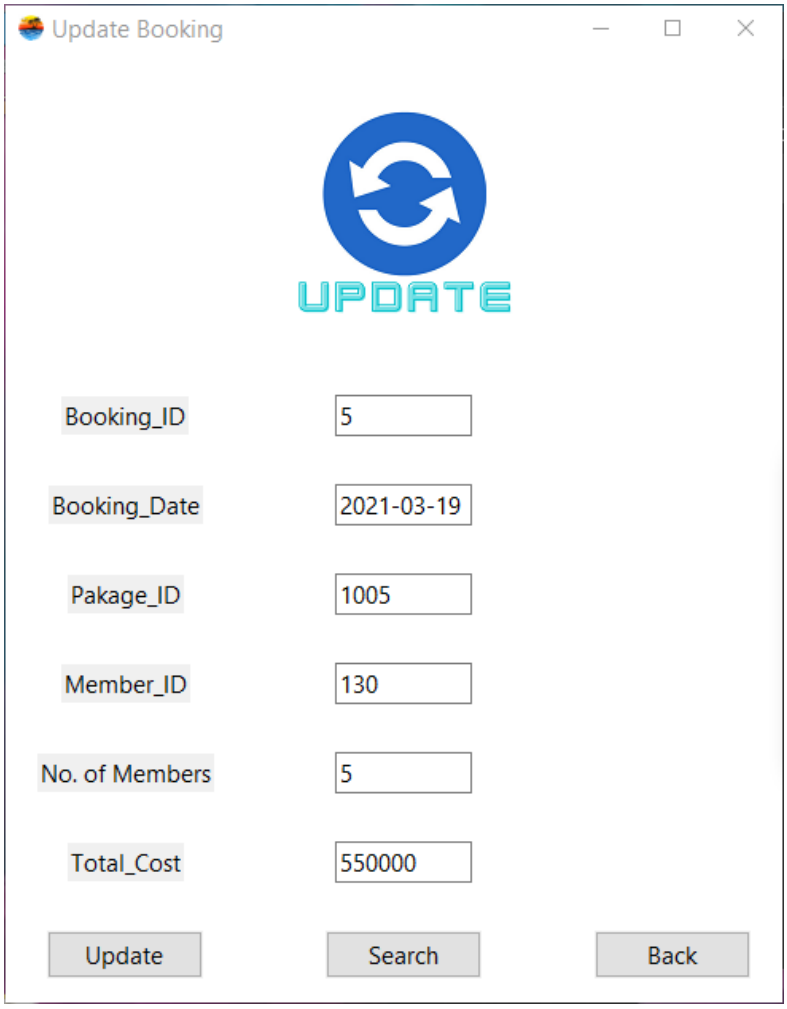

```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image
```

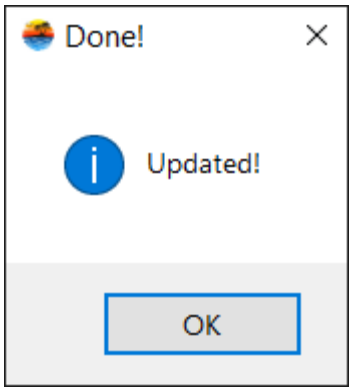
```
win=tk.Tk()
win.title('New Booking')
win.iconbitmap(r'logo.ico')
win.geometry(''+10+10'')
win.configure(bg='white')


bg = ImageTk.PhotoImage(file = "insertpic.png")
label1 = tk.Label( win, image = bg, highlightthickness=0,bd =0)
label1.grid(column=1,row=0)


lbid=ttk.Label(win,text='Booking_ID')
lbid.grid(column=0,row=1)

bid=tk.StringVar()
tbid=ttk.Entry(win,width=10,textvariable=bid)
tbid.grid(column=2,row=1)

lbd=ttk.Label(win,text='Booking_Date')
lbd.grid(column=0,row=2)

bd=tk.StringVar()
tbd=ttk.Entry(win,width=10,textvariable=bd)
tbd.grid(column=2,row=2)

lpid=ttk.Label(win,text='Pakage_ID')
lpid.grid(column=0,row=3)

pid=tk.StringVar()
tpid=ttk.Combobox(win,width=10,textvariable=pid)
tpid['values']=(1001,1002,1003,1004,1005,1006,1007,1008,1009,1010,1
011,1012,1013,1014,1015)
tpid.grid(column=2,row=3)

lmid=ttk.Label(win,text='Member_ID')
lmid.grid(column=0,row=4)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=10,textvariable=mid)
tmid.grid(column=2,row=4)

lnom=ttk.Label(win,text='No. of Members')
lnom.grid(column=0,row=5)
```

```
nom=tk.StringVar()
tnom=ttk.Combobox(win,width=3,textvariable=nom)
tnom['values']=(1,2,3,4,5,6,7,8,9,10)
tnom.grid(column=2,row=5)
tnom.current(0)

ltcost=ttk.Label(win,text='Total_Cost')
ltcost.grid(column=0,row=6)

tcost=tk.StringVar()
ttcost=ttk.Entry(win,width=10,textvariable=tcost)
ttcost.grid(column=2,row=6)

conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
cursor = conn.cursor()
row = cursor.execute("select max(Booking_ID) from booking")
ls=cursor.fetchall()
bid.set(str(ls[0][0]+1))

def _msgBox():
    try:

     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("insert into booking values(" + str(bid.get()) + ",
'"+bd.get()+"', "+str(pid.get())+", '"+mid.get()+"',
"+str(nom.get())+","+str(tcost.get())+");")
     if(cursor.rowcount>0):   # help to ensure that something
changed/added into table
         mBox.showinfo('Added!','Done!')
         bid.set('')
         bd.set('')
         pid.set('')
         mid.set('')
         tcost.set('')
         conn.commit()
     else:
         print('Not Done!')
    except Error as e :
     print("Error while connecting to MySQL", e)
    finally:
```

```
        print("MySQL connection is closed")



def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=2,row=7)

ins = ttk.Button(win, text="Submit",command=_msgBox)
ins.grid(column=0,row=7)

for child in win.winfo_children():
    child.grid_configure(padx=20, pady=20)


win.mainloop()
```

UPDATE BOOKING DETAILS






```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu

from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
```

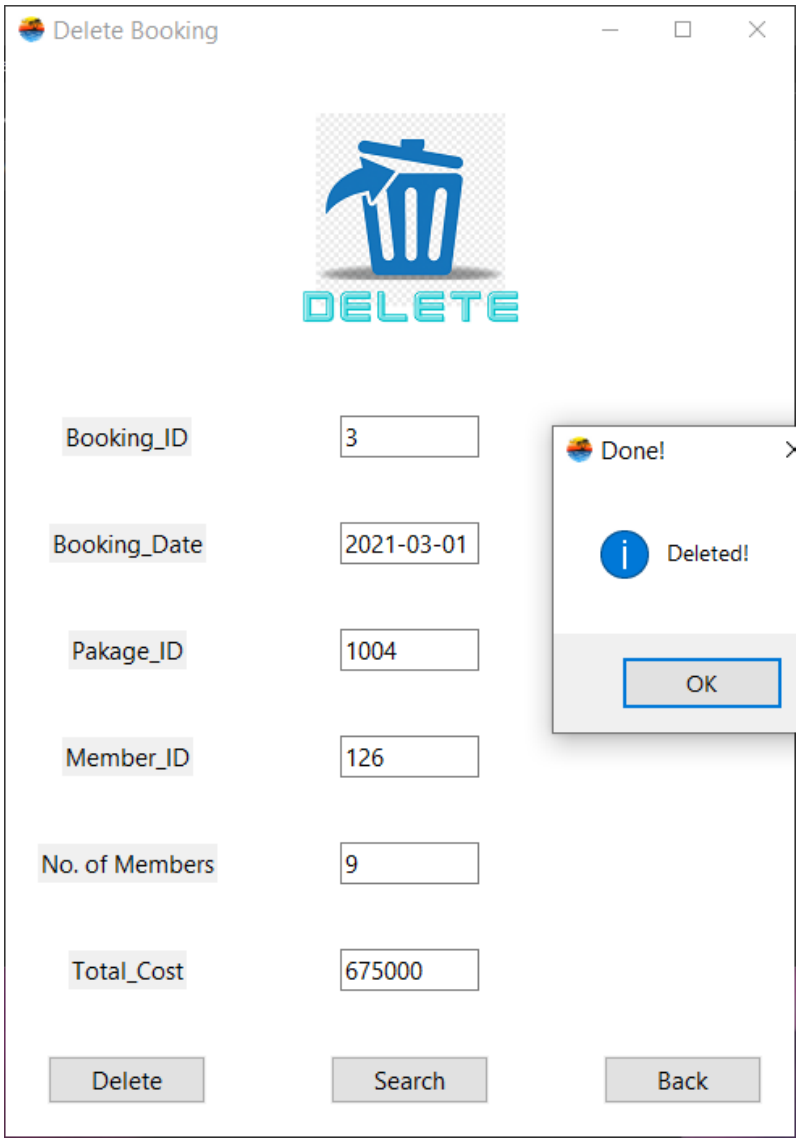
```
import pandas as pd

from PIL import ImageTk, Image


win=tk.Tk()
win.title("Update Booking")
win.iconbitmap(r'logo.ico')
win.configure(background='white')
win.geometry('+100+10')

bg = ImageTk.PhotoImage(file = "updatepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lbid=ttk.Label(win,text='Booking_ID')
lbid.grid(column=0,row=1)

bid=tk.IntVar()
tbid=ttk.Entry(win,width=10,textvariable=bid)
tbid.grid(column=1,row=1)

lbd=ttk.Label(win,text='Booking_Date')
lbd.grid(column=0,row=2)

bd=tk.StringVar()
tbd=ttk.Entry(win,width=10,textvariable=bd)
tbd.grid(column=1,row=2)

lpid=ttk.Label(win,text='Pakage_ID')
lpid.grid(column=0,row=3)

pid=tk.IntVar()
tpid=ttk.Entry(win,width=10,textvariable=pid)
tpid.grid(column=1,row=3)

lmid=ttk.Label(win,text='Member_ID')
lmid.grid(column=0,row=4)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=10,textvariable=mid)
tmid.grid(column=1,row=4)

lnom=ttk.Label(win,text='No. of Members')
lnom.grid(column=0,row=5)
```

```
nom=tk.IntVar()
tnom=ttk.Entry(win,width=10,textvariable=nom)
tnom.grid(column=1,row=5)

ltcost=ttk.Label(win,text='Total_Cost')
ltcost.grid(column=0,row=6)

tcost=tk.IntVar()
ttcost=ttk.Entry(win,width=10,textvariable=tcost)
ttcost.grid(column=1,row=6)

def _msgBox():
    try:
     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("update booking set
Booking_Date='"+str(bd.get())+"',package_id="+str(pid.get())+",
member_id='"+str(mid.get())+"', numberofmembers="+str(nom.get())+",
total_cost="+str(tcost.get())+" where Booking_Id="+str(bid.get())+";")
     if(cursor.rowcount>0):
         mBox.showinfo('Done!', 'Updated!')
         bid.set('')
         bd.set('')
         pid.set('')
         mid.set('')
         nom.set('')
         tcost.set('')

         tbd.config(state='disable')
         tpid.config(state='disable')
         tmid.config(state='disable')
         tnom.config(state='disable')
         ttcost.config(state='disable')
         conn.commit()
     else:
         print('Not Done!')
    except Error as e :
     print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")
```

```
def _fill():
    try:

        conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
        cursor = conn.cursor()
        cursor.execute("select * from booking where
booking_id="+str(bid.get()))
        ls = pd.DataFrame(cursor.fetchall())
        if(len(ls.index)>0):
            bid.set(ls.iloc[0][0])
            bd.set(ls.iloc[0][1])
            pid.set(ls.iloc[0][2])
            mid.set(ls.iloc[0][3])
            nom.set(ls.iloc[0][4])
            tcost.set(ls.iloc[0][5])
            tbid.config(state='enable')
            tbd.config(state='enable')
            tpid.config(state='enable')
            tmid.config(state='enable')
            tnom.config(state='enable')
            ttcost.config(state='enable')

        else:
            mBox.showinfo('Error', 'Booking Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")
        conn.commit()
def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=2,row=7)

ins = ttk.Button(win, text="Search",command=_fill)
ins.grid(column=1,row=7)

updt = ttk.Button(win, text="Update",command=_msgBox)
updt.grid(column=0,row=7)
```

```
for child in win.winfo_children():
    child.grid_configure(padx=20, pady=15)


win.mainloop()
```

DELETE BOOKING DETAILS



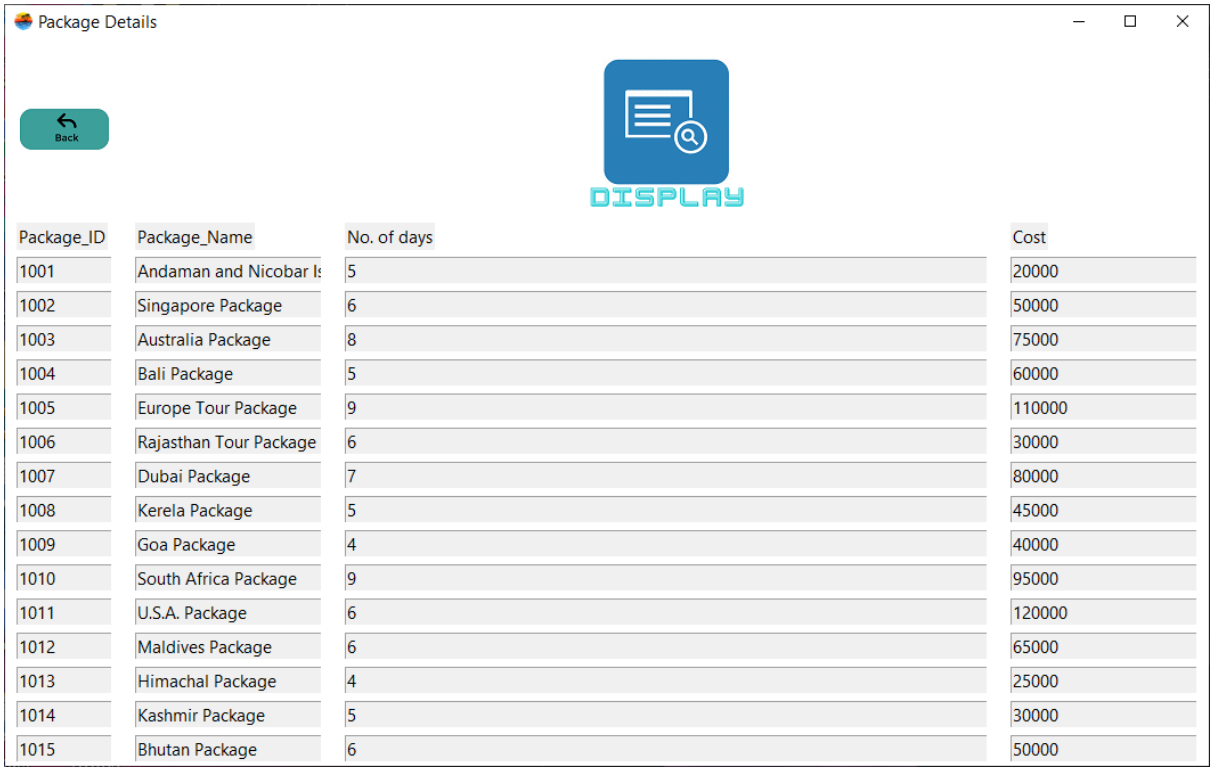

```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
import pandas as pd
from PIL import ImageTk, Image
```

```
win=tk.Tk()
win.title("Delete Booking")
win.iconbitmap(r'logo.ico')
win.configure(background='white')
win.geometry('+200+20')

bg = ImageTk.PhotoImage(file = "deletepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lbid=ttk.Label(win,text='Booking_ID')
lbid.grid(column=0,row=1)

bid=tk.IntVar()
tbid=ttk.Entry(win,width=10,textvariable=bid)
tbid.grid(column=1,row=1)

lbd=ttk.Label(win,text='Booking_Date')
lbd.grid(column=0,row=2)

bd=tk.StringVar()
tbd=ttk.Entry(win,width=10,textvariable=bd)
tbd.grid(column=1,row=2)

lpid=ttk.Label(win,text='Pakage_ID')
lpid.grid(column=0,row=3)

pid=tk.IntVar()
tpid=ttk.Entry(win,width=10,textvariable=pid)
tpid.grid(column=1,row=3)

lmid=ttk.Label(win,text='Member_ID')
lmid.grid(column=0,row=4)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=10,textvariable=mid)
tmid.grid(column=1,row=4)

lnom=ttk.Label(win,text='No. of Members')
lnom.grid(column=0,row=5)

nom=tk.IntVar()
tnom=ttk.Entry(win,width=10,textvariable=nom)
tnom.grid(column=1,row=5)
```

```
ltcost=ttk.Label(win,text='Total_Cost')
ltcost.grid(column=0,row=6)

tcost=tk.IntVar()
ttcost=ttk.Entry(win,width=10,textvariable=tcost)
ttcost.grid(column=1,row=6)

def _msgBox():
    try:
     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("delete from booking where
Booking_Id="+str(bid.get())+";")
     if(cursor.rowcount>0):
         mBox.showinfo('Done!', 'Deleted!')
         bid.set('')
         bd.set('')
         pid.set('')
         mid.set('')
         nom.set('')
         tcost.set('')


         conn.commit()
     else:
         print('Not Done!')
    except Error as e :
     print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")


def _fill():
    try:

        conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
        cursor = conn.cursor()
        cursor.execute("select * from booking where
booking_id="+str(bid.get()))
        ls = pd.DataFrame(cursor.fetchall())
        if(len(ls.index)>0):
```

```
            bid.set(ls.iloc[0][0])
            bd.set(ls.iloc[0][1])
            pid.set(ls.iloc[0][2])
            mid.set(ls.iloc[0][3])
            nom.set(ls.iloc[0][4])
            tcost.set(ls.iloc[0][5])


        else:
            mBox.showinfo('Error', 'Booking Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")
        conn.commit()
def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=2,row=7)

ins = ttk.Button(win, text="Search",command=_fill)
ins.grid(column=1,row=7)

dele = ttk.Button(win, text="Delete",command=_msgBox)
dele.grid(column=0,row=7)



for child in win.winfo_children():
    child.grid_configure(padx=20, pady=20)


win.mainloop()
```

DISPLAY PACKAGE DETAILS



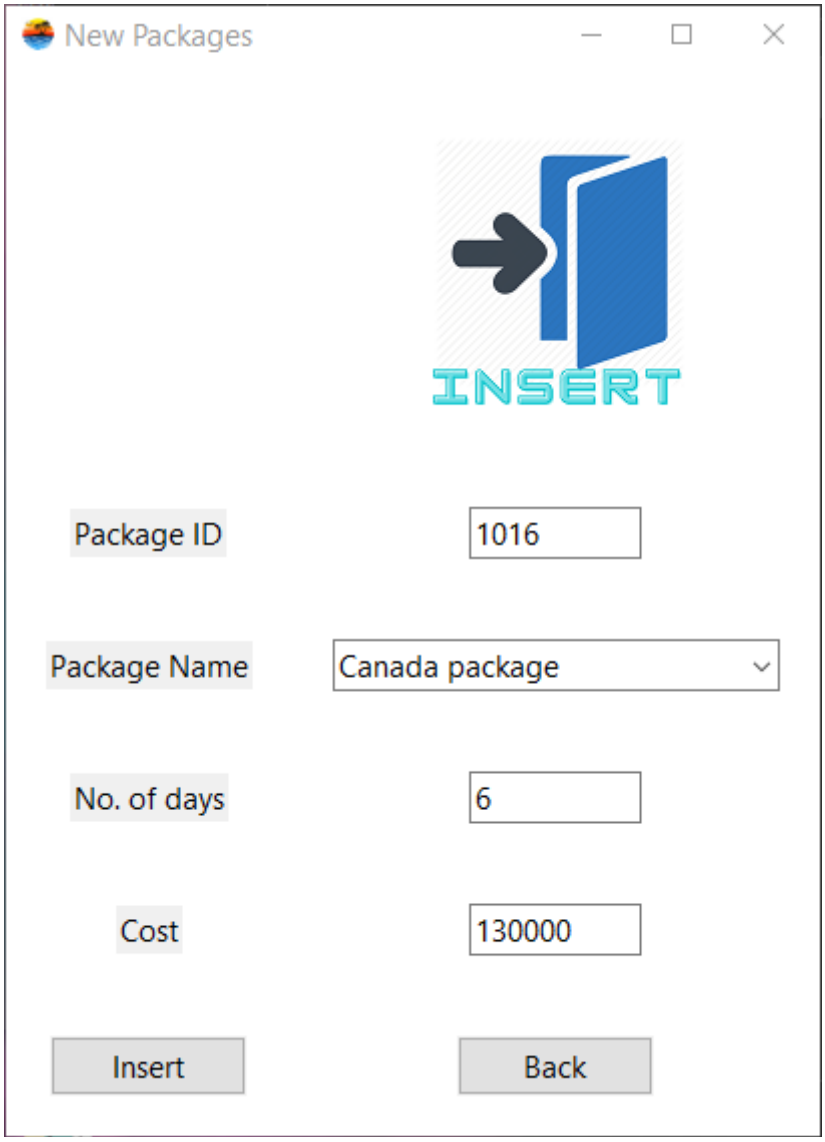
Package_ID	Package_Name	No. of days	Cost
1001	Andaman and Nicobar Is	5	20000
1002	Singapore Package	6	50000
1003	Australia Package	8	75000
1004	Bali Package	5	60000
1005	Europe Tour Package	9	110000
1006	Rajasthan Tour Package	6	30000
1007	Dubai Package	7	80000
1008	Kerela Package	5	45000
1009	Goa Package	4	40000
1010	South Africa Package	9	95000
1011	U.S.A. Package	6	120000
1012	Maldives Package	6	65000
1013	Himachal Package	4	25000
1014	Kashmir Package	5	30000
1015	Bhutan Package	6	50000

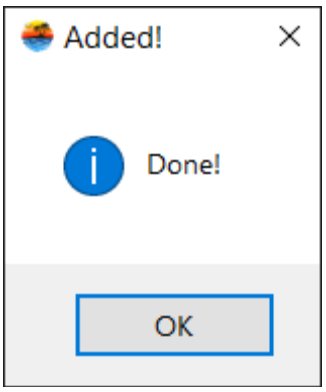
```
import pandas as pd
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image


win = tk.Tk()
win.title("Package Details")
win.configure(background='white')
```

```
win.iconbitmap(r'logo.ico')


bg = ImageTk.PhotoImage(file = "selectimage.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=2,row=0)

def back():
        win.quit()
        win.destroy()
        import mainwindow

backimg=tk.PhotoImage(file='backbuttonimageresized.png')
submit1 = tk.Button(win, text="Back",
command=back,image=backimg,highlightthickness=0,bd =0)
submit1.grid(column=0, row=0,columnspan=1)
try:

    conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
    cursor = conn.cursor()
    cursor.execute("select * from Packages;")
    ls = pd.DataFrame(cursor.fetchall())

    ttk.Label(win, text="Package_ID").grid(column=0, row=1, sticky=tk.W,
columnspan=20)
    ttk.Label(win, text="Package_Name").grid(column=1, row=1,
sticky=tk.W, columnspan=20)
    ttk.Label(win, text="No. of days").grid(column=2, row=1, sticky=tk.W,
columnspan=21)
    ttk.Label(win, text="Cost").grid(column=3, row=1, sticky=tk.W,
columnspan=25)

    for i in range(0, len(ls.index)):
        for j in range(0, len(ls.columns)):
            b = tk.Entry(win)
            b.insert(0, ls.iloc[i][j])
            b.grid(row=i+2, column=j)
            if(j==0):
                b.config(state = "readonly", width = 10)
            elif(j==2):
                b.config(state = 'readonly', width=70)
            elif(j==4):
                b.config(state = "readonly", width = 25)
```

```
            else:
                b.config(state = "readonly", width = 20)
    conn.commit()
    conn.close()
except Error as e :
    print("Error while connecting to MySQL", e)
finally:
    print("MySQL connection is closed")

for child in win.winfo_children():
    child.grid_configure(padx=10, pady=3)

win.mainloop()
```

INSERT PACKAGE DETAILS



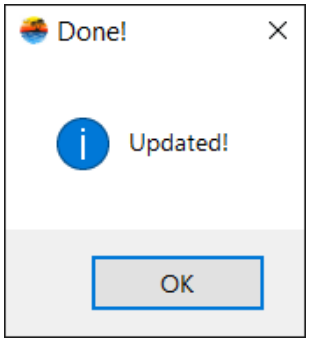



```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image
```

```
win=tk.Tk()
win.title('New Packages')
win.iconbitmap(r'logo.ico')
win.configure(bg='white')
bg = ImageTk.PhotoImage(file = "insertpic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd=0)
label1.grid(column=1,row=0)

lpid=ttk.Label(win,text='Package ID')
lpid.grid(column=0,row=1)

pid=tk.StringVar()
cpid=ttk.Entry(win,width=10,textvariable=pid)
cpid.grid(column=1,row=1)

lpname=ttk.Label(win,text='Package Name')
lpname.grid(column=0,row=2)

pname=tk.StringVar()
cpname=ttk.Combobox(win,width=25,textvariable=pname)
cpname['values']=("Andaman and Nicobar Islands Package","Singapore
Package","Australia Package","Bali Package","Europe Tour
Package","Rajasthan Tour Package","Dubai Package","Kerela
Package","Goa Package","South Africa Package","U.S.A.
Package","Maldives Package","Himachal Package","Kashmir
Package","Bhutan Package")
cpname.grid(column=1,row=2)

lnod=ttk.Label(win,text='No. of days')
lnod.grid(column=0,row=3)

nod=tk.IntVar()
tnod=ttk.Entry(win,width=10,textvariable=nod)
tnod.grid(column=1,row=3)

lcost=ttk.Label(win,text='Cost')
lcost.grid(column=0,row=5)

cost=tk.IntVar()
tcost=ttk.Entry(win,width=10,textvariable=cost)
tcost.grid(column=1,row=5)

conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
```

```
cursor = conn.cursor()
row = cursor.execute("select max(package_id) from packages")
ls=cursor.fetchall()
pid.set(str(ls[0][0]+1))

def _msgBox():
    try:

     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("insert into packages values(" + str(pid.get()) +
", '"+pname.get()+"', "+str(nod.get())+","+str(cost.get())+");")
     if(cursor.rowcount>0):   # help to ensure that something
changed/added into table
         mBox.showinfo('Added!','Done!')
         pid.set('')
         pname.set('')
         nod.set('')
         cost.set('')
         conn.commit()
     else:
         print('Not Done!')
    except Error as e :
     print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")


def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=1,row=6)

ins = ttk.Button(win, text="Insert",command=_msgBox)
ins.grid(column=0,row=6)

for child in win.winfo_children():
    child.grid_configure(padx=20, pady=20)


win.mainloop()
```

UPDATE PACKAGE DETAILS

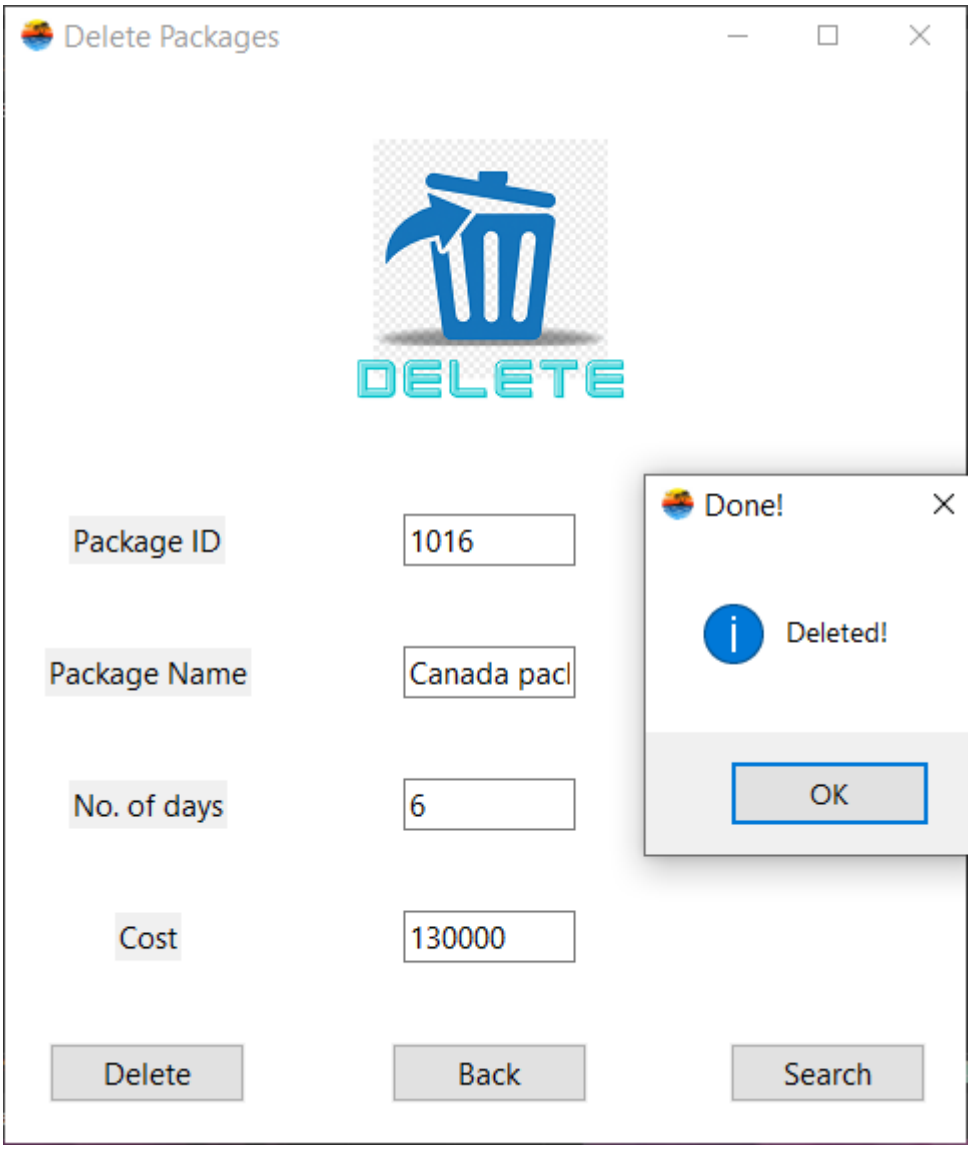


```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
import pandas as pd
from PIL import ImageTk, Image

win=tk.Tk()
win.title('Update Packages')
win.iconbitmap(r'logo.ico')
win.configure(background='white')

bg = ImageTk.PhotoImage(file = "updatepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lpid=ttk.Label(win,text='Package ID')
lpid.grid(column=0,row=1)

pid=tk.StringVar()
tpid=ttk.Entry(win,width=10,textvariable=pid)
tpid.grid(column=1,row=1)

lpname=ttk.Label(win,text='Package Name')
lpname.grid(column=0,row=2)

pname=tk.StringVar()
tpname=ttk.Entry(win,width=10,textvariable=pname)
tpname.grid(column=1,row=2)

lnom=ttk.Label(win,text='No. of days')
lnom.grid(column=0,row=4)

nom=tk.StringVar()
tnom=ttk.Entry(win,width=10,textvariable=nom)
tnom.grid(column=1,row=4)

lcost=ttk.Label(win,text='Cost')
lcost.grid(column=0,row=3)

cost=tk.StringVar()
```

```
tcost=ttk.Entry(win,width=10,textvariable=cost)
tcost.grid(column=1,row=3)

def _msgBox():
    try:
     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("update packages set
package_id='"+str(pid.get())+"',package_name='"+pname.get()+"',
totaldays="+str(nom.get())+",cost="+str(cost.get())+" where
package_id="+str(pid.get())+";")
     if(cursor.rowcount>0):
         mBox.showinfo('Done!', 'Updated!')
         pid.set('')
         pname.set('')
         nom.set('')
         cost.set('')

         tpid.config(state='disable')
         tpname.config(state='disable')
         tnom.config(state='disable')
         tcost.config(state='disable')

         conn.commit()
     else:
         print('Not Done!')
    except Error as e :
     print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")


def _fill():
    try:

        conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
        cursor = conn.cursor()
        cursor.execute("select * from packages where
package_id="+str(pid.get()))
        ls = pd.DataFrame(cursor.fetchall())
        if(len(ls.index)>0):
```

```
            pid.set(ls.iloc[0][0])
            pname.set(ls.iloc[0][1])
            nom.set(ls.iloc[0][2])
            cost.set(ls.iloc[0][3])
            tpid.config(state='enable')
            tpname.config(state='enable')
            tnom.config(state='enable')
            tcost.config(state='active')
            conn.commit()

        else:
            mBox.showinfo('Error', 'Package Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")

def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=1,row=6)

ins = ttk.Button(win, text="Update",command=_msgBox)
ins.grid(column=0,row=6)

srch=ttk.Button(win,text='Search',command=_fill)
srch.grid(column=2,row=6)

for child in win.winfo_children():
    child.grid_configure(padx=20, pady=20)


win.mainloop()
```

DELETE PACKAGE DETAILS



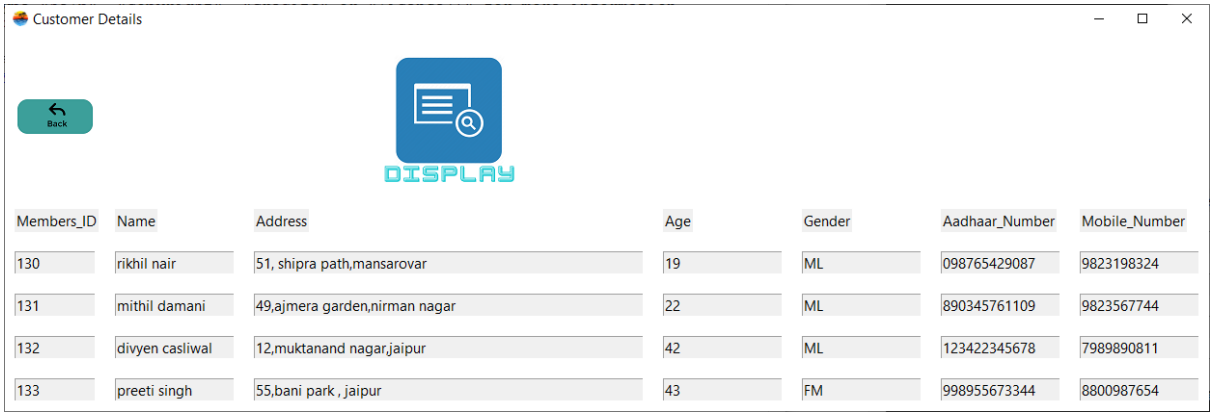

```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
```

```
import pandas as pd
from PIL import ImageTk, Image

win=tk.Tk()
win.title('Delete Packages')
win.iconbitmap(r'logo.ico')
win.configure(background='white')
win.geometry('+200+40')

bg = ImageTk.PhotoImage(file = "deletepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lpid=ttk.Label(win,text='Package ID')
lpid.grid(column=0,row=1)

pid=tk.StringVar()
tpid=ttk.Entry(win,width=10,textvariable=pid)
tpid.grid(column=1,row=1)

lpname=ttk.Label(win,text='Package Name')
lpname.grid(column=0,row=2)

pname=tk.StringVar()
tpname=ttk.Entry(win,width=10,textvariable=pname)
tpname.grid(column=1,row=2)

lnom=ttk.Label(win,text='No. of days')
lnom.grid(column=0,row=3)

nom=tk.StringVar()
tnom=ttk.Entry(win,width=10,textvariable=nom)
tnom.grid(column=1,row=3)

lcost=ttk.Label(win,text='Cost')
lcost.grid(column=0,row=5)

cost=tk.StringVar()
tcost=ttk.Entry(win,width=10,textvariable=cost)
tcost.grid(column=1,row=5)

def _msgBox():
    try:
```

```
    conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
    cursor = conn.cursor()
    row = cursor.execute("delete from packages where
package_id="+str(pid.get())+";")
    if(cursor.rowcount>0):
        mBox.showinfo('Done!', 'Deleted!')
        pid.set('')
        pname.set('')
        nom.set('')
        cost.set('')

        tpid.config(state='disable')
        tpname.config(state='disable')
        tnom.config(state='disable')
        tcost.config(state='disable')

        conn.commit()
    else:
        print('Not Done!')
   except Error as e :
    print("Error while connecting to MySQL", e)
   finally:
       print("MySQL connection is closed")


def _fill():
   try:

       conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
       cursor = conn.cursor()
       cursor.execute("select * from packages where
package_id="+str(pid.get()))
       ls = pd.DataFrame(cursor.fetchall())
       if(len(ls.index)>0):
           pid.set(ls.iloc[0][0])
           pname.set(ls.iloc[0][1])
           nom.set(ls.iloc[0][2])
           cost.set(ls.iloc[0][3])
           tpid.config(state='enable')
           tpname.config(state='enable')
           tnom.config(state='enable')
```

```
            tcost.config(state='active')
            conn.commit()

        else:
            mBox.showinfo('Error', 'Package Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")

def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=1,row=6)

ins = ttk.Button(win, text="Delete",command=_msgBox)
ins.grid(column=0,row=6)

srch=ttk.Button(win,text='Search',command=_fill)
srch.grid(column=2,row=6)
for child in win.winfo_children():
    child.grid_configure(padx=20, pady=20)


win.mainloop()
```

DISPLAY MEMBER DETAILS



Members_ID	Name	Address	Age	Gender	Aadhaar_Number	Mobile_Number
130	rikhil nair	51, shipra path,mansarovar	19	ML	098765429087	9823198324
131	mithil damani	49,ajmera garden,nirman nagar	22	ML	890345761109	9823567744
132	divyen casliwal	12,muktanand nagar,jaipur	42	ML	123422345678	7989890811
133	preeti singh	55,bani park , jaipur	43	FM	998955673344	8800987654

```
import pandas as pd
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image

win = tk.Tk()
win.title("Customer Details")
win.iconbitmap(r'logo.ico')
win.configure(background='white')

bg = ImageTk.PhotoImage(file = "selectimage.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=2,row=0)

def back():
        win.quit()
        win.destroy()
        import mainwindow


backimg=tk.PhotoImage(file='backbuttonimageresized.png')
```

```
submit1 = tk.Button(win, text=”Back”,
command=back,image=backimg,highlightthickness=0,bd =0)
submit1.grid(column=0, row=0,columnspan=1)


try:

    conn =
mysql.connector.connect(host=’localhost’,database=’tours’,user=’root’,p
assword=’’,charset=’utf8’)
    cursor = conn.cursor()
    cursor.execute(“select * from members;”)
    ls = pd.DataFrame(cursor.fetchall())

    ttk.Label(win, text=”Members_ID”).grid(column=0, row=1, sticky=tk.W,
columnspan=20)
    ttk.Label(win, text=”Name”).grid(column=1, row=1, sticky=tk.W,
columnspan=20)
    ttk.Label(win, text=”Address”).grid(column=2, row=1, sticky=tk.W,
columnspan=21)
    ttk.Label(win, text=”Age”).grid(column=3, row=1, sticky=tk.W,
columnspan=20)
    ttk.Label(win, text=”Gender”).grid(column=4, row=1, sticky=tk.W,
columnspan=25)
    ttk.Label(win, text=”Aadhaar_Number”).grid(column=5, row=1,
sticky=tk.W, columnspan=20)
    ttk.Label(win, text=”Mobile_Number”).grid(column=6, row=1,
sticky=tk.W, columnspan=20)



    for I in range(0, len(ls.index)):
        for j in range(0, len(ls.columns)):
            b = tk.Entry(win)
            b.insert(0, ls.iloc[i][j])
            b.grid(row=i+2, column=j)
            if(j==0):
                b.config(state = “readonly”, width = 10)
            elif(j==2):
                b.config(state = ‘readonly’, width=50)
            elif(j==4):
                b.config(state = “readonly”, width = 15)
            else:
                b.config(state = “readonly”, width = 15)
    conn.commit()
```

```
    conn.close()
except Error as e :
    print(“Error while connecting to MySQL”, e)
finally:
    print(“MySQL connection is closed”)

for child in win.winfo_children():
    child.grid_configure(padx=10, pady=10)

win.mainloop()
```

INSERT MEMBER DETAILS

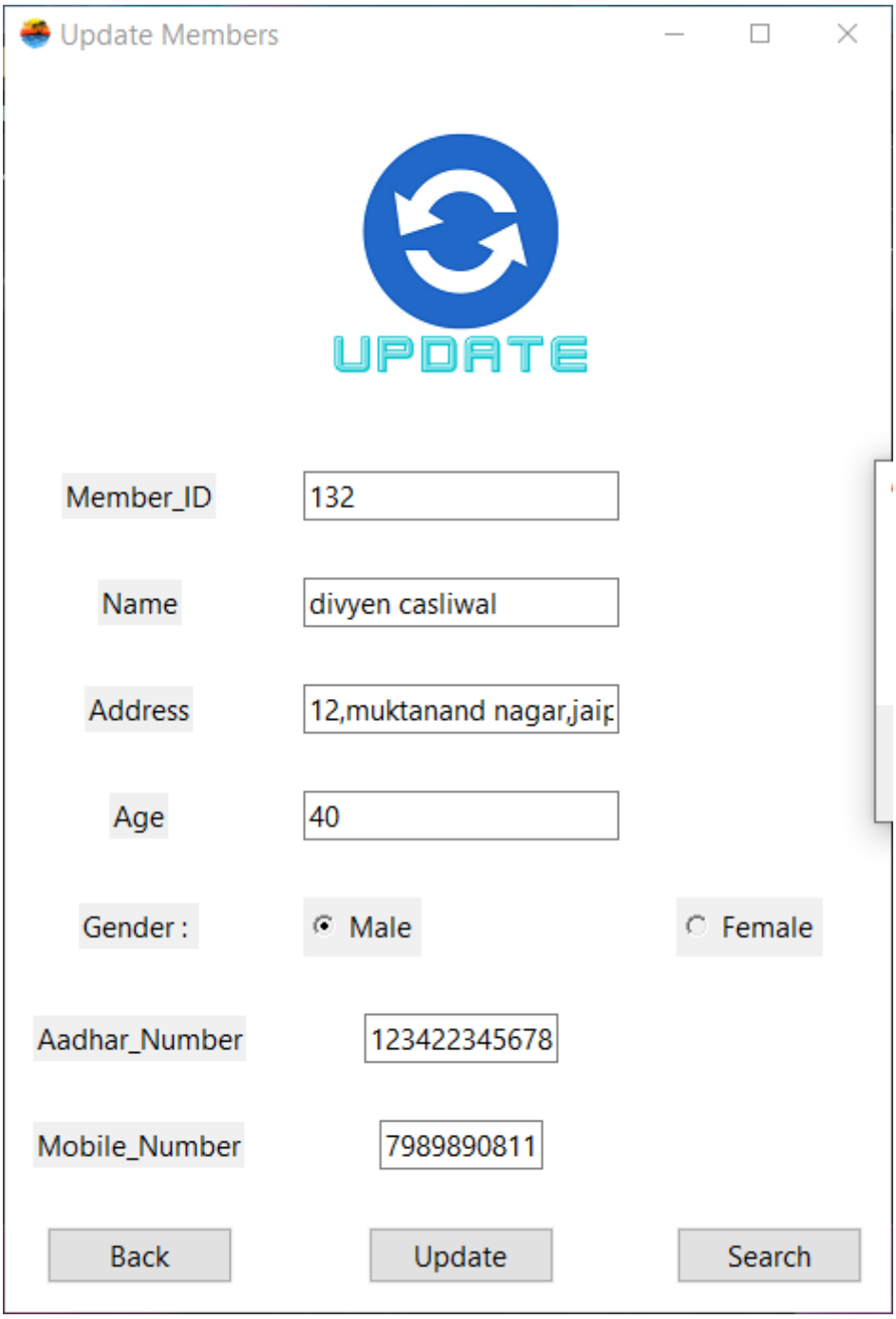

```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
from PIL import ImageTk, Image
```

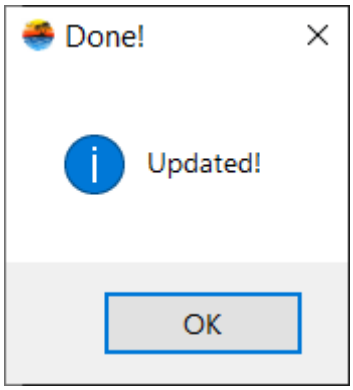
```
win=tk.Tk()
win.title('Insert Members')
win.configure(bg='white')
win.iconbitmap(r'logo.ico')
win.geometry('+200+40')

bg = ImageTk.PhotoImage(file = "insertpic.png")
label1 = tk.Label( win, image = bg , highlightthickness=0 , bd=0)
label1.grid(column=1,row=0)

lmid=ttk.Label(win,text='Member_ID',cursor='x_cursor')
lmid.grid(column=0,row=1)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=20,textvariable=mid)
tmid.grid(column=1,row=1)

lnm=ttk.Label(win,text='Name',cursor='x_cursor')
lnm.grid(column=0,row=2)

nm=tk.StringVar()
tnm=ttk.Entry(win,width=20,textvariable=nm)
tnm.grid(column=1,row=2)

ladd=ttk.Label(win,text='Address',cursor='x_cursor')
ladd.grid(column=0,row=3)

add=tk.StringVar()
tadd=ttk.Entry(win,width=20,textvariable=add)
tadd.grid(column=1,row=3)



lage=ttk.Label(win,text='Age',cursor='x_cursor')
lage.grid(column=0,row=4)

age=tk.StringVar()
tage=ttk.Combobox(win,width=8,textvariable=age)
tage['values']=('Select',1,2,3,4,5,6,7,8,9,10,
                11,12,13,14,15,16,17,18,19,20,
                21,22,23,24,25,26,27,28,29,30,
                31,32,33,34,35,36,37,38,39,40,
                41,42,43,44,45,46,47,48,49,50,
                51,52,53,54,55,56,57,58,59,60,
                61,62,63,64,65,66,67,68,69,70,
```

```
                71,72,73,74,75,76,77,78,79,80,
                81,82,83,84,85,86,87,88,89,90,
                91,92,93,94,95,96,97,98,99,100)
tage.grid(column=1,row=4)
tage.current(0)

lgen = ttk.Label(win, text="Gender : ",cursor='x_cursor')
lgen.grid(column=0, row=5)

gen = tk.StringVar()
ml = tk.Radiobutton(win, text='Male', variable=gen, value='ML')
ml.grid(column=1, row=5, sticky=tk.W)
fm=tk.Radiobutton(win, text='Female', variable=gen, value='FM')
fm.grid(column=2, row=5, sticky=tk.W)
ml.config(state='active')
fm.config(state='active')

ladh = ttk.Label(win, text='Aadhar_Number',cursor='x_cursor')
ladh.grid(column=0, row=6)

adh=tk.StringVar()
tadh=ttk.Entry(win,width=12,textvariable=adh)
tadh.grid(column=1,row=6)

lmob = ttk.Label(win, text='Mobile_Number',cursor='x_cursor')
lmob.grid(column=0, row=7)

mob=tk.StringVar()
tmob=ttk.Entry(win,width=10,textvariable=mob)
tmob.grid(column=1,row=7)

conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
cursor = conn.cursor()
row = cursor.execute("select max(Member_ID) from members")
ls=cursor.fetchall()
#m=ls[0][0]+1
#mid.set(str(m))
#mid.set(str(ls[0][0]+1))
def _msgBox():
    try:
```

```
    conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
    cursor = conn.cursor()
    row = cursor.execute("insert into members values(" + str(mid.get()) +
","'+str(nm.get())+"','"+str(add.get())+"','"+str(age.get())+"','"+str(gen.get())
+"', "'+str(adh.get())+"','"+str(mob.get())+"  );")
    if(cursor.rowcount>0):   # help to ensure that something
changed/added into table
        mBox.showinfo('Added!','Thank You!')
        mid.set('')
        nm.set('')
        add.set('')
        age.set('')
        gen.set('')
        adh.set('')
        mob.set('')


        conn.commit()
    else:
        print('Not Done!')
   except Error as e :
    print("Error while connecting to MySQL", e)
   finally:
       print("MySQL connection is closed")


def back():
   win.quit()
   win.destroy()
   import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=0,row=8)

ins = ttk.Button(win, text="Submit",command=_msgBox)
ins.grid(column=2,row=8)


for child in win.winfo_children():
   child.grid_configure(padx=15, pady=15)


win.mainloop()
```

UPDATE MEMBER DETAILS


Update Members
UPDATE
Member_ID
132
Name
divyen casliwal
Address
12,muktanand nagar,jaip
Age
40
Gender :
Male
Female
Aadhar_Number
123422345678
Mobile_Number
7989890811
Back
Update
Search


Done!
Updated!
OK

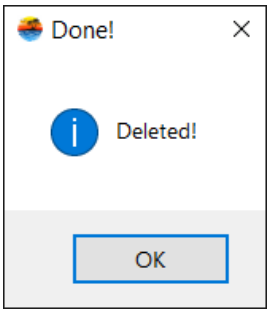
```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
import pandas as pd
from PIL import ImageTk, Image

win=tk.Tk()
win.title('Update Members')
win.iconbitmap(r'logo.ico')
win.configure(background='white')
win.geometry('+200+50')

bg = ImageTk.PhotoImage(file = "updatepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lmid=ttk.Label(win,text='Member_ID',cursor='x_cursor')
lmid.grid(column=0,row=1)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=20,textvariable=mid)
tmid.grid(column=1,row=1)

lnm=ttk.Label(win,text='Name',cursor='x_cursor')
lnm.grid(column=0,row=2)

nm=tk.StringVar()
tnm=ttk.Entry(win,width=20,textvariable=nm)
tnm.grid(column=1,row=2)

ladd=ttk.Label(win,text='Address',cursor='x_cursor')
ladd.grid(column=0,row=3)

add=tk.StringVar()
tadd=ttk.Entry(win,width=20,textvariable=add)
tadd.grid(column=1,row=3)


lage=ttk.Label(win,text='Age',cursor='x_cursor')
lage.grid(column=0,row=4)

age=tk.StringVar()
```

```
tage=ttk.Entry(win,width=20,textvariable=age)
tage.grid(column=1,row=4)

lgen = ttk.Label(win, text="Gender : ",cursor='x_cursor')
lgen.grid(column=0, row=6)

gen = tk.StringVar()
ml = tk.Radiobutton(win, text='Male', variable=gen, value='ML')
ml.grid(column=1, row=6, sticky=tk.W)
fm=tk.Radiobutton(win, text='Female', variable=gen, value='FM')
fm.grid(column=2, row=6, sticky=tk.W)
ml.config(state='active')
fm.config(state='active')

ladh = ttk.Label(win, text='Aadhar_Number',cursor='x_cursor')
ladh.grid(column=0, row=7)

adh=tk.StringVar()
tadh=ttk.Entry(win,width=12,textvariable=adh)
tadh.grid(column=1,row=7)

lmob = ttk.Label(win, text='Mobile_Number',cursor='x_cursor')
lmob.grid(column=0, row=8)

mob=tk.StringVar()
tmob=ttk.Entry(win,width=10,textvariable=mob)
tmob.grid(column=1,row=8)

def _msgBox():
    try:
     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row=cursor.execute("update members set
member_id="+str(mid.get())+",name='"+str(nm.get())+"',
address='"+str(add.get())+"', age="+str(age.get())+",
gender='"+str(gen.get())+"',aadhar_number='"+str(adh.get())+"',mobile_n
umber="+str(mob.get())+" where member_Id='"+str(mid.get())+"';")
     if(cursor.rowcount>0):
         mBox.showinfo('Done!', 'Updated!')
         mid.set('')
         nm.set('')
         add.set('')
         age.set('')
```

```
            gen.set('')
            adh.set('')
            mob.set('')

            tnm.config(state='disable')
            tadd.config(state='disable')
            tage.config(state='disable')
            ml.config(state='disable')
            fm.config(state='disable')
            tadh.config(state='disable')
            tmob.config(state='disable')

            conn.commit()
        else:
            print('Not Done!')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")


def _fill():
    try:

        conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
        cursor = conn.cursor()
        cursor.execute("select * from members where
member_id="+str(mid.get()))
        ls = pd.DataFrame(cursor.fetchall())
        if(len(ls.index)>0):
            mid.set(ls.iloc[0][0])
            nm.set(ls.iloc[0][1])
            age.set(ls.iloc[0][3])
            add.set(ls.iloc[0][2])
            gen.set(ls.iloc[0][4])
            adh.set(ls.iloc[0][5])
            mob.set(ls.iloc[0][6])
            tmid.config(state='enable')
            tnm.config(state='enable')
            tage.config(state='enable')
            tadd.config(state='enable')
            fm.config(state='active')
            tadh.config(state='normal')
```

```
            tmob.config(state='normal')
            conn.commit()

        else:
            mBox.showinfo('Error', 'Booking Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")




def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=0,row=9)

dele= ttk.Button(win, text="Update",command=_msgBox)
dele.grid(column=1,row=9)

srch=ttk.Button(win,text="Search",command=_fill)
srch.grid(column=2,row=9)

for child in win.winfo_children():
    child.grid_configure(padx=15, pady=15)


win.mainloop()
```

DELETE MEMBER DETAILS



```
import tkinter as tk
from tkinter import ttk
from tkinter import Menu
from tkinter import messagebox as mBox
import mysql.connector
from mysql.connector import Error
import pandas as pd
from PIL import ImageTk, Image

win=tk.Tk()
win.title('Delete Members')
win.iconbitmap(r'logo.ico')
win.configure(background='white')
win.geometry('+200+40')
bg = ImageTk.PhotoImage(file = "deletepic.png")
label1 = tk.Label( win, image = bg,highlightthickness=0,bd =0)
label1.grid(column=1,row=0)

lmid=ttk.Label(win,text='Member_ID',cursor='x_cursor')
lmid.grid(column=0,row=1)

mid=tk.StringVar()
tmid=ttk.Entry(win,width=20,textvariable=mid)
tmid.grid(column=1,row=1)

lnm=ttk.Label(win,text='Name',cursor='x_cursor')
lnm.grid(column=0,row=2)
nm=tk.StringVar()
tnm=ttk.Entry(win,width=20,textvariable=nm)
tnm.grid(column=1,row=2)

ladd=ttk.Label(win,text='Address',cursor='x_cursor')
ladd.grid(column=0,row=3)

add=tk.StringVar()
tadd=ttk.Entry(win,width=20,textvariable=add)
tadd.grid(column=1,row=3)


lage=ttk.Label(win,text='Age',cursor='x_cursor')
lage.grid(column=0,row=5)

age=tk.StringVar()
tage=ttk.Entry(win,width=20,textvariable=age)
tage.grid(column=1,row=5)
```

```
lgen = ttk.Label(win, text="Gender : ",cursor='x_cursor')
lgen.grid(column=0, row=6)

gen = tk.StringVar()
ml = tk.Radiobutton(win, text='Male', variable=gen, value='ML')
ml.grid(column=1, row=6, sticky=tk.W)
fm=tk.Radiobutton(win, text='Female', variable=gen, value='FM')
fm.grid(column=2, row=6, sticky=tk.W)
ml.config(state='active')
fm.config(state='active')

ladh = ttk.Label(win, text='Aadhar_Number',cursor='x_cursor')
ladh.grid(column=0, row=7)

adh=tk.StringVar()
tadh=ttk.Entry(win,width=12,textvariable=adh)
tadh.grid(column=1,row=7)

lmob = ttk.Label(win, text='Mobile_Number',cursor='x_cursor')
lmob.grid(column=0, row=8)

mob=tk.StringVar()
tmob=ttk.Entry(win,width=10,textvariable=mob)
tmob.grid(column=1,row=8)

def _msgBox():
    try:
     conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
     cursor = conn.cursor()
     row = cursor.execute("delete from members where
member_id="+str(mid.get())+";")
     if(cursor.rowcount>0):
         mBox.showinfo('Done!', 'Deleted!')
         mid.set('')
         nm.set('')
         add.set('')
         age.set('')
         gen.set('')
         adh.set('')
         mob.set('')

         tnm.config(state='disable')
```

```
            tadd.config(state='disable')
            tage.config(state='disable')
            ml.config(state='disable')
            fm.config(state='disable')
            tadh.config(state='disable')
            tmob.config(state='disable')

            conn.commit()
        else:
            print('Not Done!')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")


def _fill():
    try:

        conn =
mysql.connector.connect(host='localhost',database='tours',user='root',pa
ssword='',charset='utf8')
        cursor = conn.cursor()
        cursor.execute("select * from members where
member_id="+str(mid.get()))
        ls = pd.DataFrame(cursor.fetchall())
        if(len(ls.index)>0):
            mid.set(ls.iloc[0][0])
            nm.set(ls.iloc[0][1])
            age.set(ls.iloc[0][3])
            add.set(ls.iloc[0][2])
            gen.set(ls.iloc[0][4])
            adh.set(ls.iloc[0][5])
            mob.set(ls.iloc[0][6])
            tmid.config(state='enable')
            tnm.config(state='enable')
            tage.config(state='enable')
            tadd.config(state='enable')
            ml.config(state='active')
            fm.config(state='active')
            tadh.config(state='normal')
            tmob.config(state='normal')
            conn.commit()

        else:
```

```
            mBox.showinfo('Error', 'Booking Doesnt Exist')
    except Error as e :
        print("Error while connecting to MySQL", e)
    finally:
        print("MySQL connection is closed")




def back():
    win.quit()
    win.destroy()
    import mainwindow

bck = ttk.Button(win, text="Back", command=back)
bck.grid(column=0,row=9)

dele= ttk.Button(win, text="Delete",command=_msgBox)
dele.grid(column=1,row=9)

srch=ttk.Button(win,text="Search",command=_fill)
srch.grid(column=2,row=9)

for child in win.winfo_children():
    child.grid_configure(padx=15, pady=15)


win.mainloop()
```

Database Tables

```
mysql> show tables;
+----------------+
| Tables_in_tours |
+----------------+
| booking        |
| login          |
| members        |
| packages       |
+----------------+
```

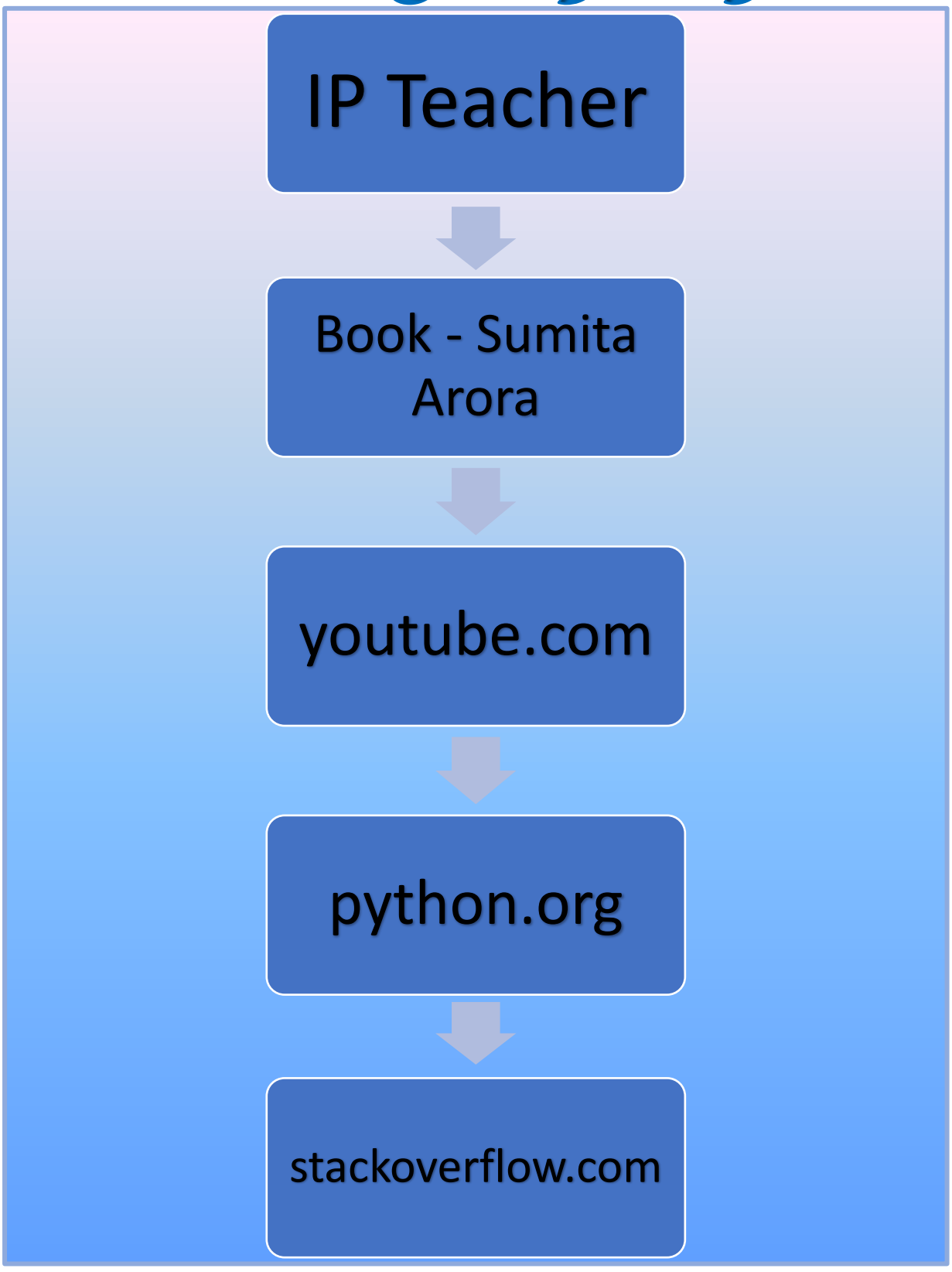
```
mysql> desc login;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| User_iD  | varchar(20) | NO   | PRI | NULL    |       |
| Password | varchar(20) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
```

```
mysql> desc booking;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| Booking_ID      | int(11)     | NO   | PRI | 0       |       |
| Booking_Date    | date        | YES  |     | NULL    |       |
| Package_Id      | int(11)     | YES  |     | NULL    |       |
| Member_ID       | varchar(20) | YES  |     | NULL    |       |
| NumberOfMembers | int(11)     | YES  |     | NULL    |       |
| Total_Cost      | int(11)     | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
6 rows in set (4.26 sec)
```

```
mysql> desc packages;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| package_id   | int(11)     | NO   | PRI | 0       |       |
| package_name | varchar(40) | YES  |     | NULL    |       |
| totaldays    | int(11)     | YES  |     | NULL    |       |
| cost         | int(11)     | YES  |     | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.05 sec)
```

```
mysql> desc members;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| member_id     | int(11)     | NO   | PRI | 0       |       |
| Name          | varchar(20) | YES  |     | NULL    |       |
| Address       | varchar(40) | YES  |     | NULL    |       |
| Age           | int(11)     | YES  |     | NULL    |       |
| Gender        | varchar(6)  | YES  |     | NULL    |       |
| Aadhar_Number | char(12)    | YES  |     | NULL    |       |
| Mobile_Number | bigint(20)  | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
```

Bibliography



Remarks