```
#MODULES USED
import mysql.connector
from decimal import Decimal
import matplotlib.pyplot as plt
import numpy as np
import tkinter as tk
from tkinter import ttk
from tkinter import *
from tkinter import messagebox
import tkinter.messagebox
from tkcalendar import Calendar, DateEntry
from PIL import Image, ImageTk

#NEXT BUTTON ON EXPENSES PAGE

def Next ():
  try:
    FO = int(food.get())
    C = int(clothing.get())
    T = int(transport.get())
    FE = int(fees.get())
    O = int(others.get())
    global anc

    db = mysql.connector.connect(host="localhost", user = "root",
password = "Root", database = "Auxilium")
    cursor = db.cursor()
    val = [('Food', FO, anc), ('Clothing', C, anc),
           ('Transport', T, anc), ('Fees', FE, anc), ('Others', O,
anc)]
    print("set anc", anc)

    cursor.executemany(
```

```
        "INSERT INTO AUXILIUM.EXP VALUES (curdate(),%s, %s, %s)",
(val))


    db.commit()

    cursor.close()


    tkinter.messagebox.showinfo(

        "CONGRATULATIONS", "Your details have been inserted
successfully    ")

    print("You have inserted ", {FO}, {C}, {T}, {FE}, {O})

    print(cursor.rowcount, "record inserted.")


  except:

           tkinter.messagebox.showerror("ERROR", "Please enter all
the details. Click enter to add it.")

           print("Error occurred in addition of data")


def expense_graph():

   try:

        def calendar_view():

            def print_sel():

              global start_date

              start_date = cal.selection_get()

              print(start_date)

            top = tk.Toplevel(root6)

            cal = Calendar(top, font = "Arial 14", selectmode =
'day', cursor = "hand1", year = 2018, month = 2, day = 5)

            cal.pack(fill = "both", expand = True)

            ttk.Button(top, text = "ok", command = print_sel).pack()

            #ttk.Button(top, text = 'Exit', command =
tk.destroy).pack()

            #exit_button = t.Button(frame, text= 'Exit', command =
root.destroy).grid(row = 9, column = 0)
```

```
        def dateentry_view():
            def print_sel():
                global end_date
                end_date = cal.get_date()
                print (end_date)
            top = tk.Toplevel(root6)
            ttk.Label(top, text = 'Choose date').pack(padx = 10,
pady = 10)
            cal = DateEntry(top, width = 12, background =
'darkblue',
                            foreground = 'white', borderwidth = 2)
            cal.pack(padx = 10, pady = 10)
            ttk.Button(top, text = "ok", command = print_sel).pack()
        root6 = tk.Tk()
        s = ttk.Style(root6)
        s.theme_use('clam')


        def graph ():
          try:
            #TOTAL EXPENSES
                global anc
                db1 = mysql.connector.connect(
                    host="localhost", user="root", password="Root",
database="Auxilium")
                cursor1 = db1.cursor()
                date = start_date
                date11 = end_date
                k = int(anc)
                food_sum = "select sum(expenses) from Auxilium.EXP
WHERE UserId = %s and cat = 'Food' and DOE between %s and %s "
                for result in cursor1.execute(food_sum, (anc, date,
date11), multi = True):
                    if result.with_rows:
                        #print("Rows produced by statement
'{}':".format(result.statement))
```

```
                        row = result.fetchall()
                        results = [tuple(str(item) for item in t)
for t in row]
                        food_total = int(results[0][0])
                        #print(type(food_total))
                    else:
                        print("Number of rows affected by statement
'{}': {}".format(result.statement, result.rowcount))
                cursor1.close()


                db2 = mysql.connector.connect(host="localhost",
user="root", password="Root", database="Auxilium")
                cursor2 = db2.cursor()
                date2 = start_date
                date22 = end_date
                clothing_sum = "select sum(expenses) from
Auxilium.EXP WHERE userid = %s and CAT = 'Clothing' and DOE between
%s and %s"
                for result2 in cursor2.execute(clothing_sum,
(anc,date2, date22), multi=True):
                    if result2.with_rows:
                        #print("Rows produced by statement
'{}':".format(result2.statement))
                        row2 = result2.fetchall()
                        results2 = [tuple(str(item) for item in t)
                                    for t in row2]
                        cloth_total = int(results2[0][0])
                    else:
                        print("Number of rows affected by statement
'{}': {}".format(result2.statement, result2.rowcount))
                cursor2.close()


                db3 = mysql.connector.connect(host="localhost",
user="root", password="Root", database="Auxilium")
                cursor3 = db3.cursor()
                date3 = start_date
```

```
                date33 = end_date

                transport_sum = "select sum(expenses) from
Auxilium.EXP WHERE userid= %s and CAT = 'Transport' and DOE between
%s and %s"

                for result3 in cursor3.execute(transport_sum,
(anc,date3, date33), multi=True):

                    if result3.with_rows:

                        #print("Rows produced by statement
'{}':".format(result3.statement))

                        row3 = result3.fetchall()

                        results3 = [tuple(str(item) for item in t)

                                    for t in row3]

                        transport_total = int(results3[0][0])

                    else:

                        print("Number of rows affected by statement
'{}': {}".format(result3.statement, result3.rowcount))

                cursor3.close()


                db4 = mysql.connector.connect(host="localhost",
user="root", password="Root", database="Auxilium")

                cursor4 = db4.cursor()

                date4 = start_date

                date44 = end_date

                fees_sum = "select sum(expenses) from Auxilium.EXP
WHERE UserId= %s and cat = 'Fees' and DOE between %s and %s"

                for result4 in cursor4.execute(fees_sum, (anc,date4,
date44), multi=True):

                    if result4.with_rows:

                        #print("Rows produced by statement
'{}':".format(result4.statement))

                        row4 = result4.fetchall()

                        results4 = [tuple(str(item) for item in t)

                                    for t in row4]

                        fees_total = int(results4[0][0])

                    else:
```

```
                        print("Number of rows affected by statement
'{}': {}".format(result4.statement, result4.rowcount))
                cursor4.close()


                db5 = mysql.connector.connect(
                    host="localhost", user="root", password="Root",
database="Auxilium")
                cursor5 = db5.cursor()
                date5 = start_date
                date55 = end_date
                others_sum = "select sum(expenses) from Auxilium.EXP
WHERE userid= %s and CAT = 'Others' and DOE between %s and %s"
                for result5 in cursor5.execute(others_sum,
(anc,date5, date55), multi=True):
                    if result5.with_rows:
                        #print("Rows produced by statement
'{}':".format(result5.statement))
                        row5 = result5.fetchall()
                        results5 = [tuple(str(item) for item in t)
                                    for t in row5]
                        others_total = int(results5[0][0])
                    else:
                        print("Number of rows affected by statement
'{}': {}".format(result5.statement, result5.rowcount))
                cursor5.close()


                l = [int(results[0][0]), int(results2[0][0]), int(
                    results3[0][0]), int(results4[0][0]),
int(results5[0][0])]
                #print('Your expenditure: ', l)


            #TOTAL EXPENSES BAR GRAPH
                data = {'FOOD': food_total, 'CLOTHING': cloth_total,
'TRANSPORT':transport_total,'FEES':fees_total,'OTHERS':others_total}
                courses = list(data.keys())
```

```
                values = list(data.values())
                fig = plt.figure(figsize = (10, 5))
                plt.bar(courses, values, color  = 'maroon', width =
0.4)
                plt.xlabel("Category")
                plt.ylabel("Total Expenditure")
                plt.title("Your Total Expenditure")
                plt.show()
          except:
              tkinter.messagebox.showerror("ERROR", "No data for
graph found in the chosen time period Please choose the time
period.")
              print ("No data for graph found in the chosen time
period :(")


        ttk.Button(root6, text = 'Starting Date', command =
calendar_view).pack(padx = 10, pady = 10)
        ttk.Button(root6, text = 'End Date', command =
dateentry_view).pack(padx = 10, pady = 10)
        ttk.Button(root6, text = 'Next', command = graph).pack(padx
= 10, pady = 10)
        root6.mainloop()


   except:
        tkinter.messagebox.showerror("ERROR", "Please choose the
time period.")


#NET WORTH WINDOW
def net_worth():
    try:
        dbI = mysql.connector.connect(host = "localhost", user =
"root", password = "Root", database = "Auxilium")
        cursorI = dbI.cursor()
        income_sum = "select sum(income) from INC"


        for resultI in cursorI.execute(income_sum, multi = True):
```

```
          if resultI.with_rows:

            #print("Rows produced by statement
'{}':".format(resultI.statement))

            rowI = resultI.fetchall()

            resultsI = [tuple(str(item) for item in t) for t in
rowI]

            income_total = int(resultsI[0][0])


          else:

            print("Number of rows affected by statement '{}':
{}".format(resultI.statement, resultI.rowcount))

        cursorI.close()


        #TOTAL EXPENSES

        dbN = mysql.connector.connect(host = "localhost", user =
"root", password = "Root", database = "Auxilium")

        cursorN = dbN.cursor()

        net_sum = "select sum(expenses) from EXP"


        for resultN in cursorN.execute(net_sum, multi = True):

          if resultN.with_rows:

            #print("Rows produced by statement
'{}':".format(resultN.statement))

            rowN = resultN.fetchall()

            resultsN = [tuple(str(item) for item in t) for t in
rowN]

            net_total = int(resultsN[0][0])

            #print ("Your net worth is: ",(income_total-net_total))

          else:

            print("Number of rows affected by statement '{}':
{}".format(resultN.statement, resultN.rowcount))

        cursorN.close()


    except:
```

```
        tkinter.messagebox.showerror("ERROR", "Could not display
your net worth :(")


    rootN = Toplevel()

    rootN.geometry("500x350")

    rootN.title('NET WORTH PAGE')

    rootN.tk.call('wm', 'iconphoto', rootN._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

    #rootN.iconphoto(False, tk.PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

    bg2 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Desktop Background.png")

    bg2_label = Label(rootN, image = bg2)

    bg2_label.place(x = 0, y = 0)

    rootN.bg2 = bg2


    label_0 = tk.Label(rootN, text = "Total Income (in ₹): ", width
= 20,font = ("bold",14),fg = 'blue')

    label_0.place(x = 60, y = 60)

    label_inc = tk.Label(rootN, text = income_total, width = 15,
font = ("bold", 14), fg = 'red')

    label_inc.place(x = 300, y = 60)


    label_1 = tk.Label(rootN, text = "Total Expenditure (in ₹): ",
width = 20, font = ("bold", 14), fg = 'blue')

    label_1.place(x = 60, y = 130)

    label_exp = tk.Label(rootN, text = net_total, width = 15, font =
("bold",14), fg = 'red')

    label_exp.place(x = 300, y = 130)


    label_3 = tk.Label(rootN, text =  "Net Worth (in ₹):", width =
20, font = ("bold", 14), fg = 'blue')

    label_3.place(x = 60, y = 200)
```

```
    label_net = tk.Label(rootN, text = income_total-net_total, width
= 15, font = ("bold", 14), fg = 'red')
    label_net.place(x = 300, y = 200)


    Button(rootN, text = 'BACK' , width = 35, bg = "Red", font =
("bold", 12), fg = 'white', command = main_page1).place(x = 100, y =
280)
    rootN.mainloop()


def expenses():
    global food
    global clothing
    global transport
    global fees
    global others
    global root3
    root3 = Toplevel()
    #root1.wm_state('iconic')
    root3.geometry("800x500")
    root3.title("EXPENSES")
    root3.tk.call('wm', 'iconphoto', root3._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))
    bg = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Desktop Background.png")
    bg_label = Label(root3, image = bg)
    bg_label.place(x = 0, y = 0)
    root3.bg = bg


    def ex(win):
        win.destroy()
        #root1.wm_state('zoomed')


    def open_img2():
```

```
        x = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Expenses.jpg"
        img = Image.open(x)
        img = img.resize((280,250), Image.ANTIALIAS)
        img = ImageTk.PhotoImage(img)
        panel = Label(root3, image = img).place(x = 500, y = 120)
        root3.img = img
    open_img2()


    label_0 = tk.Label(root3,text = "Enter the following: ", width =
20, font = ("bold", 22))
    label_0.place(x = 100, y = 50)


    label_1 = tk.Label(root3, text = "Food: ", width = 20, fg =
'blue', font = ('bold',14))
    label_1.place(x = 80,y = 130)
    food = tk.Entry(root3)
    food.place(x = 330,y = 130)


    label_2  = tk.Label(root3,text = "Clothing: ", width = 20,fg =
"blue", font = ("bold", 14))
    label_2.place(x = 80,y = 170)
    clothing = tk.Entry(root3)
    clothing.place(x = 330,y = 170)


    label_3  = tk.Label(root3,text = "Transport: ", width = 20,fg =
"blue", font = ("bold",14))
    label_3.place(x = 80,y = 210)
    transport = tk.Entry(root3)
    transport.place(x = 330,y = 210)


    label_4  = tk.Label(root3,text = "Fees: ", width = 20,fg =
"blue", font = ("bold",14))
    label_4.place(x = 80,y = 250)
    fees = tk.Entry(root3)
```

```
    fees.place(x = 330,y = 250)


    label_5  = tk.Label(root3,text = "Others: ", width = 20,fg =
"blue", font = ("bold",14))

    label_5.place(x = 80,y = 290)

    others = tk.Entry(root3)

    others.place(x = 330,y = 290)


    Button(root3, text = 'SUBMIT' , width = 20,bg = "blue", fg =
'white', font = ("bold",12),command = Next).place(x = 80,y = 350)

    Button(root3, text = 'NEXT' , width = 20,bg = "blue", fg =
'white', font = ("bold",12),command = budgeting).place(x = 310,y =
350)

    Button(root3, text = 'BACK' , width = 20,bg = "red", fg =
'white', font = ("bold",12),command = Next).place(x = 80,y = 420)

    Button(root3, text = "EXIT", width = 20, bg = "red", font =
("bold", 12), fg = 'white', command = lambda: ex(root3)).place(x =
310, y = 420)


    root3.mainloop()


#INCOME DETAILS OF THE USER

def option_1 ():

    global source1

    global monthly_income

    rootN = Toplevel()

    #root1.wm_state('iconic')

    rootN.geometry("400x500")

    rootN.maxsize(400,430)

    rootN.title('INCOME PAGE')

    bgs = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Desktop Background.png")

    bgs_label = Label(rootN, image = bgs)

    bgs_label.place(x = 0,y = 0)
```

```
    rootN.tk.call('wm', 'iconphoto', rootN._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))


    def ex(win):
        win.destroy()
        #root1.wm_state('zoomed')


    def open_inc():
        x = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Income Details.png"
        img = Image.open(x)
        img = img.resize((300, 150), Image.ANTIALIAS)
        img = ImageTk.PhotoImage(img)
        panel = Label(rootN, image = img).place(x = 50,y = 290)
        rootN.img = img
    open_inc()


    label_0  = tk.Label(rootN, text = "ENTER THE FOLLOWING: ", width
= 22,font = ("bold",16),fg="red")
    label_0.place(x = 60,y = 40)
    label_income  = tk.Label(rootN, text = "Source of Income:",
width = 20,font = ("bold",12),fg = "blue")
    label_income.place(x = 20,y = 100)
    source1 = tk.Entry(rootN)
    source1.place(x = 260,y = 100)
    label_month  = tk.Label(rootN, text = "Monthly Income:", width =
20,font = ("bold",12),fg = "blue")
    label_month.place(x = 20,y = 150)
    monthly_income = tk.Entry(rootN)
    monthly_income.place(x = 260,y = 150)


    Button(rootN, text = 'EXIT' , width = 25, bg = "Red", font =
("bold", 12), fg = 'white', command = lambda: ex(rootN)).place(x =
100, y = 250)
```

```
    Button(rootN, text = "NEXT", width = 25, bg = "blue", font =
("bold", 12), fg = 'white', command = income).place(x = 80, y = 200)

    mylabel = tk.Label(rootN, text = '').grid(row = 8, column = 0)

    rootN.mainloop()


#INCOME DETAILS SAVED INTO THE DATABASE

def income ():

    global income_category

    global Income

    income_category = source1.get()

    Income = monthly_income.get()

    global anc


    #try:

    db = mysql.connector.connect(host="localhost", user="root",
password="Root", database="Auxilium")

    cursor = db.cursor()

    print("value of ANC in Income", anc)

    print(456)


    insert_query = "insert into inc values (curdate(),%s, %s, %s)"

    print('abc')

    val = (income_category, Income,anc)

    print(789)

    cursor.execute(insert_query, val)#fail

    print('def')

    db.commit()

    cursor.close()

    print(123)

    tkinter.messagebox.showinfo(

        "CONGRATULATIONS", "Your details have been inserted
successfully   ")

    print("You have inserted ", {income_category}, {Income})
```

```
    expenses()


    #except:
        #tkinter.messagebox.showerror("ERROR", "Could not insert
details.")
        #print("Error occurred in inserting values")



#LOAN DETAILS SAVED INTO THE DATABASE
def details():
    mydb = mysql.connector.connect(host = 'localhost', user =
'root', password = 'Root', database = 'auxilium')
    mycursor = mydb.cursor()
    N = name.get()
    A = int(age.get())
    E = emp_choice.get()
    I = inc_choice.get()
    L = loan_choice.get()
    P = int(pin.get())
    query = "Insert into details2 values
('{}',{},'{}','{}','{}',{})".format(N, A, E, I, L, P)
    mycursor.execute(query)
    mydb.commit()
    messagebox.showinfo("CONGRATULATIONS!!","""Your details have
been inserted into
    the database successfully.""")


#ASKING THE DETAILS
def loan_details():
    global name
    global age
    global emp_choice
    global inc_choice
    global loan_choice
```

```
    global pin


    root4 = Toplevel()

    root4.geometry("480x570")

    #root1.wm_state('iconic')

    root4.title('DETAILS FOR LOAN')

    root4.tk.call('wm', 'iconphoto', root4._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

    bg = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\picsgreens.png")

    bg_label = Label(root4,image = bg)

    bg_label.place(x = 0,y = 0)

    l1 = tk.Label(root4,text = 'ENTER THE FOLLOWING DETAILS:', width
= 30,font = ('bold',18))

    l1.place(x = 30,y = 50)


    def ex(win):

        win.destroy()

        #root1.wm_state('zoomed')


    def open_loan():

        x = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Details for loan.jpg"

        img = Image.open(x)

        img = img.resize((210, 140), Image.ANTIALIAS)

        img = ImageTk.PhotoImage(img)

        panel = Label(root4, image = img).place(x = 260,y = 420)

        root4.img = img

    open_loan()


    l2  = tk.Label(root4,text = "Full name (PAN card):", width =
20,font = ("bold",14),fg = "blue")

    l2.place(x = 20,y = 130)
```

```
    name =  tk.Entry(root4)
    name.place(x = 260,y = 130)

    l3  = tk.Label(root4,text = "Your age :", width = 20,font =
("bold",14),fg = "blue")
    l3.place(x = 20,y = 170)
    age =  tk.Entry(root4)
    age.place(x = 260,y = 170)

    l4  = tk.Label(root4,text = "Gender :", width = 20,font =
("bold",14),fg = "blue")
    l4.place(x = 20,y = 210)
    rad1 = Radiobutton(root4, text = 'Male', value = 1)
    rad2 = Radiobutton(root4, text = 'Female', value = 2)
    rad3 = Radiobutton(root4, text = 'Other', value = 3)
    rad1.grid(column = 1, row = 0)
    rad1.place(x = 260,y = 210)
    rad2.grid(column = 2, row = 0)
    rad2.place(x = 330,y = 210)
    rad3.grid(column = 3, row = 0)
    rad3.place(x = 400,y = 210)

    l6  = tk.Label(root4,text = "Employer type:", width = 20,font =
("bold",14),fg = "blue")
    l6.place(x = 20,y = 250)
    a = tk.StringVar()
    emp_choice = ttk.Combobox(root4,width = 27,
                          textvariable = a)
    emp_choice['values'] = ('N-O-N-E',
                        'Pvt/MNC job',
                        'Government job',
                        'Proprietorship/Partnership',)
    emp_choice.grid(column = 1,row = 4)
    emp_choice.place(x = 260,y = 250)
```

```
    emp_choice.current(1)


    l7 = tk.Label(root4,text = "Income per month :",width = 20,font
= ("bold",14),fg = "blue")
    l7.place(x = 20,y = 290)
    c = tk.StringVar()
    inc_choice = ttk.Combobox(root4, width = 27,
                              textvariable = c)
    inc_choice['values'] = ('0-2,50,000',
                            ' 2,50,001-5,00,000',
                            ' 5,00,001-7,50,000',
                            ' 7,50,001-10,00,000',
                            ' 10,00,001-12,50,000',
                            ' 12,50,001-15,00,000', )
    inc_choice.grid(column = 1, row = 6)
    inc_choice.place(x = 260,y = 290)
    inc_choice.current(1)


    l8 = tk.Label(root4,text = "Type of loan :",width = 20,font =
("bold",14),fg = "blue")
    l8.place(x = 20,y = 330)
    d = tk.StringVar()
    loan_choice = ttk.Combobox(root4, width = 27,
                              textvariable = d)
    loan_choice['values'] = ('Personal loan',
                            'Education loan',
                            'Home loan',
                            'Business loan',
                            'Car loan',
                            'Gold loan', )
    loan_choice.grid(column = 1, row = 6)
    loan_choice.place(x = 260,y = 330)
    loan_choice.current(1)
```

```
    l9 = tk.Label(root4,text = "Pincode :", width = 20,font =
("bold",14),fg = "blue")

    l9.place(x = 20,y = 370)

    pin = tk.Entry(root4)

    pin.place(x = 260,y = 370)


#DETAILS WILL DISPLAY FOR LOAN CHOSEN

    def clicked():

        x = loan_choice.get()

        if x == 'Personal loan':

            messagebox.showinfo("DOCUMENTS REQUIRED", "Identity
proof , Address proof , Bank statement of previous 3 months , Two
latest salary slip/current dated salary certificate with Form 16")

            root5 = Toplevel()

            canvas1 = Canvas(root5, width = 1000, height = 800)

            canvas1.pack(expand = YES, fill = BOTH)

            img1 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Personal.PNG")

            root5.img1 = img1

            canvas1.create_image(50, 50, anchor = tk.NW, image =
img1)


        elif x == 'Education loan':

            messagebox.showinfo("DOCUMENTS REQUIRED","KYC documents
, Bank Statement / Pass Book of last 6 months , Copy of admission
letter of the Institute along with fees schedule , Mark sheets /
passing certificates of S.S.C , H.S.C , Degree courses")

            root5 = Toplevel()

            canvas2 = Canvas(root5, width = 900, height = 450)

            canvas2.pack(expand = YES, fill = BOTH)

            img2 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Education.PNG")

            root5.img2 = img2
```

```
            canvas2.create_image(50,50, anchor = tk.NW, image =
img2)


        elif x == 'Home loan':

            messagebox.showinfo("DOCUMENTS REQUIRED", "Identity
proof , Address proof , Proof of income ,Other documents , Property
documents")

            root5 = Toplevel()

            canvas3 = Canvas(root5, width = 900, height = 600)

            canvas3.pack(expand = YES, fill = BOTH)

            img3 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Home.PNG")

            root5.img3 = img3

            canvas3.create_image(50,50, anchor = tk.NW, image =
img3)


        elif x == 'Business loan':

            messagebox.showinfo("DOCUMENTS REQUIRED", "Identity
proof , Address proof , Income proof, Financial documents, Business
Ownership proof")

            root5 = Toplevel()

            canvas4 = Canvas(root5, width = 900, height = 400)

            canvas4.pack(expand = YES, fill = BOTH)

            img4 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Business.PNG")

            root5.img4 = img4

            canvas4.create_image(50,50, anchor = tk.NW, image =
img4)


        elif x == 'Car loan':

            messagebox.showinfo("DOCUMENTS REQUIRED", "Photo ID with
age proof , Signed application with 3 photos , Residence proof, Bank
statement for last 6 months , Form 16/Income tax returns")

            root5 = Toplevel()

            canvas5 = Canvas(root5, width = 900, height = 400)
```

```
            canvas5.pack(expand = YES, fill = BOTH)

            img5 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Car.PNG")

            root5.img5 = img5

            canvas5.create_image(50,50, anchor = tk.NW, image =
img5)


        elif x == 'Gold loan':

            messagebox.showinfo("DOCUMENTS REQUIRED", "Identity
proof , Rent agreement/Passport/Driving license/Aadhar card/Voter's
ID for address proof")

            root5 = Toplevel()

            canvas6 = Canvas(root5, width = 900, height = 450)

            canvas6.pack(expand = YES, fill = BOTH)

            img6 = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Gold.PNG")

            root5.img6 = img6

            canvas6.create_image(50,50, anchor = tk.NW, image =
img6)


        else:

            messagebox.showinfo("N-O-T-H-I-N-G", "Kindly select a
valid option")


    Button(root4, text = 'SUBMIT DETAILS', width = 25,font =
('bold',13),bg = 'blue', fg = 'white', command = details).place(x =
20,y = 440)

    Button(root4, text = 'NEXT', width = 25,bg = "blue", font =
('bold',13),fg = 'white', command = clicked).place(x = 20,y = 480)
#TRANSITION HERE

    Button(root4, text = "EXIT", width = 25, bg = 'red', font =
("bold", 12), fg = 'white', command = lambda: ex(root4)).place(x =
20, y = 520)  #TRANSITION HERE

    root4.mainloop()


def banking():
```

```
    mydb = mysql.connector.connect(host = 'localhost', user =
'root', password = 'Root', database = 'Auxilium')
    mycursor = mydb.cursor()


    def ex(win):
        win.destroy()
        root.wm_state('zoomed')


    def writeAccount():
        from tkinter import messagebox
        def bclick():
            a = acc_no.get()
            n = ahn.get()
            at = atype.get()
            i = initial.get()
            p = auth.get()
            if (a == '' or n == '' or at == '' or i == '' or p ==
''):
                messagebox.showwarning('Error', 'All fields are
required')
                new_window.destroy()
                root.wm_state('zoomed')
            else:
                mycursor.execute('use Auxilium')
                mycursor.execute('insert into accounts values ("'+ a
+'", "'+ n +'", "'+ at +'", "'+ i +'", "'+ p +'")')
                mycursor.execute('commit')
                messagebox.showinfo('Account Created', 'Thank you
for submitting. Please click X on the to exit.')
                new_window.destroy()
                root.wm_state('zoomed')


        new_window = tk.Toplevel()
        new_window.configure(bg='lightblue')
```

```
        new_window.resizable(width = False, height = False)

        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))

        new_window.title('Create Account')

        mylabel = tk.Label(new_window, text = 'Create an account\n',
font = ('Arial', 15), bg = 'lightblue').grid(sticky = 'W', row = 0,
column = 0)

        mylabel = tk.Label(new_window, text = 'PLEASE TYPE IN
UPPERCASE LETTERS\n', bg = 'lightblue').grid(sticky = 'W', row = 1,
column = 0)


        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg = 'lightblue').grid(sticky = 'W', row = 2, column = 0)

        acc_no = tk.Entry(new_window)

        acc_no.grid(sticky = 'W', row = 3, column = 0)


        lab2 = tk.Label(new_window, text = 'Enter the account holder
name : \n', bg = 'lightblue').grid(sticky = 'W', row = 4, column =
0)

        ahn = tk.Entry(new_window)

        ahn.grid(sticky = 'W', row = 5, column = 0)


        lab4 = tk.Label(new_window,text ='Enter the type of account
[enter C or S]: \n', bg = 'lightblue').grid(sticky = 'W', row = 6,
column = 0)

        atype = tk.Entry(new_window)

        atype.grid(sticky = 'W', row = 7, column = 0)


        lab5 = tk.Label(new_window,text = 'Enter the initial
amount(>=500 for saving and >=1000 for checking): \n', bg =
'lightblue').grid(sticky = 'W', row = 8, column = 0)

        initial = tk.Entry(new_window)

        initial.grid(sticky = 'W', row = 9, column = 0)


        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(sticky = 'W', row = 10, column=0)

        pin = tk.StringVar()
```

```
        auth = tk.Entry(new_window, textvariable = pin, show = '•')
        auth.grid(sticky = 'W', row = 11, column = 0)
        mylabel = tk.Label(frame, text = '',
bg='lightblue').grid(row = 12, column = 0)


        tk.Button(new_window, text = 'Submit', command = lambda:
bclick()).grid(row = 13, column = 1)
        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 14, column = 1)
        new_window.mainloop()


    def Deposit():
        from tkinter import messagebox
        def bclick():
            a = acc_no.get()
            d = dep.get()
            p = auth.get()
            if (a == '' or d == '' or p == ''):
                messagebox.showwarning('Error', 'All fields are
required')
                new_window.destroy()
                root.wm_state('zoomed')
            else:
                mycursor.execute('select * from accounts where
Account_No = "'+ a +'"')
                detail = []
                for i in mycursor:
                    for j in i:
                        detail.append(j)
                if int(p) == detail[-1]:
                        mycursor.execute('use Auxilium')
                        mycursor.execute('set @d = cast("'+ d +'" as
unsigned)')
                        mycursor.execute('set @a = cast("'+ a +'" as
unsigned)')
```

```
                        mycursor.execute('set @p = cast("'+ p +'" as
unsigned)')
                        mycursor.execute('update accounts set Amount
= Amount + @d where Account_No = @a and PIN = @p')
                        mycursor.execute('commit')
                        messagebox.showinfo('Amount deposited',
'Thank you for submitting. Please click X on the to exit.')
                        new_window.destroy()
                        root.wm_state('zoomed')
                elif int(p) != detail[-1]:
                    messagebox.showwarning('Error', 'Invalid PIN')
                    new_window.destroy()
                    root.wm_state('zoomed')


        new_window = tk.Toplevel()
        new_window.configure(bg='lightblue')
        new_window.resizable(width = False, height = False)
        new_window.title('Cash Deposit')
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))
        mylabel = tk.Label(new_window, text = 'Cash Deposit\n', font
= ('Arial', 15), bg='lightblue').grid(sticky = 'W', row = 0, column
= 0)


        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg ='lightblue').grid(sticky = 'W', row = 2, column = 0)
        acc_no = tk.Entry(new_window)
        acc_no.grid(sticky = 'W', row = 3, column = 0)


        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(stick = 'W', row = 4, column=0)
        pin = tk.StringVar()
        auth = tk.Entry(new_window, textvariable = pin, show = '•')
        auth.grid(sticky = 'W', row = 5, column = 0)
```

```
        lab2 = tk.Label(new_window, text = 'Amount of money to be
deposited: \n', bg = 'lightblue').grid(sticky = 'W', row = 6, column
= 0)

        dep = tk.Entry(new_window)

        dep.grid(sticky = 'W', row = 7, column = 0)


        tk.Button(new_window, text = 'Submit', command = lambda:
bclick()).grid(row = 8, column = 1)

        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 9, column = 1)

        new_window.mainloop()


    def Withdraw():

        from tkinter import messagebox

        def bclick():

            a = acc_no.get()

            w = wit.get()

            p = auth.get()

            if a == '' or w == '' or p == '':

                messagebox.showwarning('Error', 'All fields are
required')

                new_window.destroy()

                root.wm_state('zoomed')

            else:

                mycursor.execute('select * from accounts where
Account_No = "'+ a +'"')

                detail = []

                for i in mycursor:

                    for j in i:

                        detail.append(j)

                if int(p) == detail[-1]:

                    mycursor.execute('use Auxilium')

                    mycursor.execute('set @w = cast("'+ w +'" as
unsigned)')
```

```
                    mycursor.execute('set @a = cast("'+ a +'" as
unsigned)')
                    mycursor.execute('set @p = cast("'+ p +'" as
unsigned)')
                    mycursor.execute('update accounts set Amount =
Amount - @w where Account_No = @a and PIN = @p')
                    mycursor.execute('commit')
                    messagebox.showinfo('Amount withdrawn', 'Thank
you for submitting. Please click X on the to exit.')
                    new_window.destroy()
                    root.wm_state('zoomed')
                elif int(p) != detail[-1]:
                    messagebox.showwarning('Error', 'Invalid PIN')
                    new_window.destroy()
                    root.wm_state('zoomed')


        new_window = tk.Toplevel()
        new_window.configure(bg='lightblue')
        new_window.resizable(width = False, height = False)
        new_window.title('Cash Withdrawal')
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))
        mylabel = tk.Label(new_window, text = 'Cash Withdrawal\n',
font = ('Arial', 15), bg = 'lightblue').grid(sticky = 'W', row = 0,
column = 0)


        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg = 'lightblue').grid(sticky = 'W', row = 2, column = 0)
        acc_no = tk.Entry(new_window)
        acc_no.grid(sticky = 'W', row = 3, column = 0)


        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(stick = 'W', row = 4, column=0)
        pin = tk.StringVar()
        auth = tk.Entry(new_window, textvariable = pin, show = '•')
```

```
        auth.grid(sticky = 'W', row = 5, column = 0)


        lab2 = tk.Label(new_window, text = 'Amount of money to be
withdrawn: \n', bg = 'lightblue').grid(sticky = 'W', row = 6, column
= 0)
        wit = tk.Entry(new_window)
        wit.grid(sticky = 'W', row = 7, column = 0)


        tk.Button(new_window, text = 'Submit', command = lambda:
bclick()).grid(row = 8, column = 1)
        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 9, column = 1)
        new_window.mainloop()


    def balance():
        from tkinter import messagebox
        def bclick(a, p):
            if a == '' or p == '':
                messagebox.showwarning('Error', 'All fields are
required')
                new_window.destroy()
                root.wm_state('zoomed')
            else:
                mycursor.execute('select * from accounts where
Account_No = "'+ a +'"')
                detail = []
                for i in mycursor:
                    for j in i:
                        detail.append(j)
                if int(p) == detail[-1]:
                    mycursor.execute('use Auxilium')
                    mycursor.execute('set @a = cast("'+ a +'" as
unsigned)')
                    mycursor.execute('select Amount from accounts
where Account_No = @a')
```

```
                    for i in mycursor:
                        for j in range(0, len(i)):
                            lab2 = tk.Label(new_window, text =
'Account Balance:', bg = 'lightblue').grid(sticky = 'W', row = 6,
column = 0)
                            lab3 = tk.Label(new_window, text = '',
bg = 'lightblue').grid(sticky = 'W', row = 7, column = 0)
                            e = tk.Entry(new_window)
                            e.grid(sticky = 'W', row = 7, column =
0)
                            e.insert(0, i[j])
                elif int(p) != detail[-1]:
                    messagebox.showwarning('Error', 'Invalid PIN')
                    new_window.destroy()
                    root.wm_state('zoomed')


        new_window = tk.Toplevel()
        new_window.configure(bg='lightblue')
        new_window.resizable(width = False, height = False)
        new_window.title('Balance Inquiry')
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))
        mylabel = tk.Label(new_window, text = 'Balance Inquiry\n',
font = ('Arial', 15), bg = 'lightblue').grid(sticky = 'W', row = 0,
column = 0)


        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg='lightblue').grid(sticky = 'W', row = 2, column = 0)
        acc_no = tk.Entry(new_window)
        acc_no.grid(sticky = 'W', row = 3, column = 0)


        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(stick = 'W', row = 4, column=0)
        pin = tk.StringVar()
        auth = tk.Entry(new_window, textvariable = pin, show = '•')
```

```
        auth.grid(sticky = 'W', row = 5, column = 0)


        tk.Button(new_window, text = 'Submit', command = lambda:
bclick(acc_no.get(), auth.get())).grid(row = 8, column = 1)
        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 9, column = 1)
        new_window.mainloop()


    def displayAll():
        from tkinter import messagebox
        from tkinter import ttk


        new_window = tk.Toplevel()
        new_window.geometry('300x300')
        new_window.configure(bg='lightblue')
        new_window.resizable(width = False, height = False)
        new_window.title('Account Holders List')


        mylabel = tk.Label(new_window, text = 'Account Holders
List\n', font = ('Arial', 15), bg = 'lightblue').grid(sticky = 'W',
row = 0, column = 0)


        mycursor.execute('use Auxilium')
        mycursor.execute('select Account_Holder_Name, Type from
accounts')
        tree = ttk.Treeview(new_window)
        tree['show'] = 'headings'
        tree['columns'] = ('Account_Holder_Name', 'Type')


        tree.heading('Account_Holder_Name', text = 'Account Holder
Name', anchor = tk.W)
        tree.heading('Type', text = 'Account Type', anchor = tk.W)


        i = 0
        for j in mycursor:
```

```
            tree.insert('', i, text = '', values = (j[0], j[1]))
            i += 1
        tree.grid(sticky = 'W', row = 2, column = 0)


    def modifyAccount():
        from tkinter import messagebox
        def bclick(a, n, t, p):
            if (a == '' or n == '' or t == '' or p == ''):
                messagebox.showwarning('Insert Status', 'All fields
are required')
                new_window.destroy()
                root.wm_state('zoomed')
            else:
                mycursor.execute('select * from accounts where
Account_No = "'+ a +'"')
                detail = []
                for i in mycursor:
                    for j in i:
                        detail.append(j)
                if int(p) == detail[-1]:
                    mycursor.execute('use Auxilium')
                    mycursor.execute('set @a = cast("'+ a +'" as
unsigned)')
                    mycursor.execute('set @n = "'+ n +'"')
                    mycursor.execute('set @t = "'+ t +'"')
                    mycursor.execute('update accounts set
Account_Holder_Name = @n where Account_No = @a')
                    mycursor.execute('update accounts set Type = @t
where Account_No = @a')
                    mycursor.execute('commit')
                    messagebox.showinfo('Account Updated', 'Thank
you for submitting. Please note that if you have submitted the
incorrect account number, your changes will not be saved. Please
click X on the to exit.')
                    new_window.destroy()
```

```
                    root.wm_state('zoomed')
                elif int(p) != detail[-1]:
                    messagebox.showwarning('Error', 'Invalid PIN')
                    new_window.destroy()
                    root.wm_state('zoomed')

        new_window = tk.Toplevel()
        new_window.configure(bg='lightblue')
        new_window.resizable(width = False, height = False)
        new_window.title('Create Account')
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))
        mylabel = tk.Label(new_window, text = 'Update Account\n',
font = ('Arial', 15), bg='lightblue').grid(sticky = 'W', row = 0,
column = 0)
        mylabel = tk.Label(new_window, text = 'You can only update
AHN & Type of Account\n', bg ='lightblue').grid(sticky = 'W', row =
1, column = 0)
        mylabel = tk.Label(new_window, text = 'PLEASE TYPE IN
UPPERCASE LETTERS\n', bg ='lightblue').grid(sticky = 'W', row = 2,
column = 0)

        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg='lightblue').grid(sticky = 'W', row = 3, column = 0)
        acc_no = tk.Entry(new_window)
        acc_no.grid(sticky = 'W', row = 4, column = 0)

        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(stick = 'W', row = 5, column=0)
        pin = tk.StringVar()
        auth = tk.Entry(new_window, textvariable = pin, show = '•')
        auth.grid(sticky = 'W', row = 6, column = 0)

        lab2 = tk.Label(new_window,text = 'Enter the account holder
name : \n', bg ='lightblue').grid(sticky = 'W', row = 7, column = 0)
        ahn = tk.Entry(new_window)
```

```
        ahn.grid(sticky = 'W', row = 8, column = 0)


        lab3 = tk.Label(new_window,text ='Enter the type of account
[enter C or S]: \n', bg ='lightblue').grid(sticky = 'W', row = 9,
column = 0)

        atype = tk.Entry(new_window)

        atype.grid(sticky = 'W', row = 10, column = 0)


        tk.Button(new_window, text = 'Submit', command = lambda:
bclick(acc_no.get(), ahn.get(), atype.get(), auth.get())).grid(row =
11, column = 1)

        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 12, column = 1)

        new_window.mainloop()


    def deleteAccount():

        from tkinter import messagebox

        def bclick(a, r, p):

            if (a == '' or r == '' or p == ''):

                messagebox.showwarning('Error', 'All fields are
required')

                new_window.destroy()

                root.wm_state('zoomed')

            else:

                mycursor.execute('select * from accounts where
Account_No = "'+ a +'"')

                detail = []

                for i in mycursor:

                    for j in i:

                        detail.append(j)

                if int(p) == detail[-1]:

                    mycursor.execute('use Auxilium')

                    mycursor.execute('set @a = cast("'+ a +'" as
unsigned)')

                    mycursor.execute('delete from accounts where
Account_No = @a')
```

```
                    mycursor.execute('insert into Feedback values
("'+ r +'")')

                    mycursor.execute('commit')

                    messagebox.showinfo('Account Updated', 'Thank
you for submitting. Please note that if you have submitted the
incorrect account number, your changes will not be saved. Please
click X on the to exit.')

                    new_window.destroy()

                    root.wm_state('zoomed')

                elif int(p) != detail[-1]:

                    messagebox.showwarning('Error', 'Invalid PIN')

                    new_window.destroy()

                    root.wm_state('zoomed')


        new_window = tk.Toplevel()

        new_window.configure(bg='lightblue')

        new_window.resizable(width = False, height = False)

        new_window.title('Create Account')

        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))

        mylabel = tk.Label(new_window, text = 'Close Account\n',
font = ('Arial', 15), bg='lightblue').grid(sticky = 'W', row = 0,
column = 0)


        lab1 = tk.Label(new_window, text = 'Enter the account no:
\n', bg='lightblue').grid(sticky = 'W', row = 1, column = 0)

        acc_no = tk.Entry(new_window)

        acc_no.grid(sticky = 'W', row = 2, column = 0)


        lab6 = tk.Label(new_window, text= 'Enter your PIN: \n', bg =
'lightblue').grid(stick = 'W', row = 3, column=0)

        pin = tk.StringVar()

        auth = tk.Entry(new_window, textvariable = pin, show = '•')

        auth.grid(sticky = 'W', row = 4, column = 0)
```

```
        lab2 = tk.Label(new_window, text = 'Why do you want to close
your account?\n', bg='lightblue').grid(sticky = 'W', row = 5, column
= 0)

        reason = tk.Entry(new_window)

        reason.grid(sticky = 'W', row = 6, column = 0)

        newlab = tk.Label(new_window, text = '', bg =
'lightblue').grid(row = 7, column = 0)

        lab4 = tk.Label(new_window, text = 'Once you click "Submit",
your account will only be removed from our e-banking database. Thank
you for using Auxilium Banking Services',
bg='lightblue').grid(sticky = 'W', row = 8, column = 0)

        lab3 = tk.Label(new_window, text = 'You can walk into
closest bank branch near you anytime between 12:00PM & 6:00PM to
completely close your account\n', bg='lightblue').grid(sticky = 'W',
row = 9, column = 0)



        tk.Button(new_window, text = 'Submit', command = lambda:
bclick(acc_no.get(), reason.get(), pin.get())).grid(row = 10, column
= 1)

        tk.Button(new_window, text = '   Exit   ', command = lambda:
ex(new_window)).grid(row = 11, column = 1)

        new_window.mainloop()


    img = "C:\\Users\\Vijay\\Desktop\\Sahaana\\School\\Comp
Sci\\GR12 - Project\\Bank Management.png"

    w, h = 1200, 1080


    root = tk.Toplevel()

    root.geometry('900x500')

    root.maxsize(900, 500)

    root.title('Auxilium Banking Services')

    root.resizable(width = False, height = False)

    root.tk.call('wm', 'iconphoto', root._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))


    canvas = tk.Canvas(root, width = w, height = h)

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

```
    img = ImageTk.PhotoImage(Image.open(img).resize((900, 500),
Image.ANTIALIAS))
    canvas.background = img
    bg = canvas.create_image(0, 0, anchor = tk.NW, image = img)


    frame = tk.LabelFrame(root, padx = 25, pady = 25)
    frame.grid(row = 0, column = 0, padx = 10, pady = 10)


    mylabel = tk.Label(frame, text = 'Hey there! What would you like
to do today? \n', font = ('Arial', 15)).grid(row = 0, column = 0)


    l = (
        ['New Account', 'New Account'],
        ['Cash Deposit', 'Cash Deposit'],
        ['Cash Withdrawal', 'Cash Withdrawal'],
        ['Balance Inquiry', 'Balance Inquiry'],
        ['Account Holders List', 'Account Holders List'],
        ['Modify an Account', 'Modify an Account'],
        ['Close an Account', 'Close an Account'],
        )
    var = tk.StringVar()
    var.set('New account')
    i = 0
    for text, mode in l:
        r = tk.Radiobutton(frame, text = text, variable = var, value
= mode, command = lambda: clicked(var.get()), font = ('Arial', 10))
        r.grid(row = 1 + i, column = 0)
        i += 1


    window = canvas.create_window(10, 10, anchor = tk.NW, window =
frame)


    def clicked(value):
```

```
        if value == 'New Account':
            root.wm_state('iconic')
            writeAccount()

        elif value == 'Cash Deposit':
            root.wm_state('iconic')
            Deposit()

        elif value == 'Cash Withdrawal':
            root.wm_state('iconic')
            Withdraw()

        elif value == 'Balance Inquiry':
            root.wm_state('iconic')
            balance()

        elif value == 'Account Holders List':
            root.wm_state('iconic')
            displayAll()

        elif value == 'Modify an Account':
            root.wm_state('iconic')
            modifyAccount()

        elif value == 'Close an Account':
            root.wm_state('iconic')
            deleteAccount()

    mylabel = tk.Label(frame, text = '').grid(row = 8, column = 0)
    exit_button = tk.Button(frame, text= 'Exit', command =
root.destroy).grid(row = 9, column = 0)
    root.mainloop()
```

```
def vfa():
    import tkinter as t
    from PIL import Image, ImageTk

    def ex(win):
        win.destroy()
        root.wm_state('zoomed')

    def retirement():
        from tkinter import scrolledtext as st
        def bclick():
            ad1 = '''Great! You have come to the right place. Retirement planning is a long but useful
process that would help you have a comfortable and secure retirement. I am only here to give you
advice on how to go about it but I'm afraid I am not programmed to help you with respect to your
situation. I hope that's alright with you. So, shall we begin our journey?'''

            ad2 = '''The first step to planning your retirement is to assess your situation. An honest
assessment of where you are financially is pivotal to planning your retirement. Begin by counting
how much you have accumulated in accounts earmarked for retirement. This includes balances in IRAs
(individual retirement accounts) and workplace retirement plans.
\nA tip: Eliminate debt, especially high-interest debts such as credit cards. It is crucial to get
your finances under control. This should be done at least 10 years before retiring.'''

            ad3 = '''Yes, you can include taxable accounts if you would be using them specifically for retirement.'''
```

ad4 = '''The next step is to identify your sources of income. Additional income can come from a many

places outside of savings. You should consider that money as well. If you are covered by a pension plan, great!

You can add the monthly income from that.'''

ad5 = '''What are your retirement goals? Would you like to travel around the world or would you like to

live a peaceful life in a modest house in the hills? Do consider your retirement goals while planning.

Here's a tip: Prepare a budget every month to estimate regular expenditures such as housing, food and

dining out, and leisure activities. Do consider health and medical expenses such as life insurance,

medicines, and doctor’s visits into your budget.'''

ad6 = '''If you are at least 8-10 years before retirement, try to increase your savings rate and cut back

on unnecessary spending. It is necessary. If you have many debts and you are considering retirement

now, I'm sorry but you are not financially good for retirement.'''

ad7 = '''You are not good to go for retirement if you are struggling to pay current bills, nor accounting

for inflation in expenses, have high level of debts, no plans for future major expenses, no monthly and

long-term financial plan.'''

ad8 = '''Finally, you also need to assess your risk tolerance for investments. A bear market (when a market

experiences prolonged price declines) with only a few years left for retirement can hurt you chances of retiring at

the right time. Retirement portfolios at this stage should focus primarily on high-quality, dividend-paying

stocks and investment-grade bonds to produce both conservative growth and income.

\nOne guideline suggests that investors should subtract their age from 110 to determine how much to

```
invest in stocks. A 70-year-old, for example, would target an
allocation of 40% stocks and 60% bonds.

\nBe aware that if you’re behind on your savings, do not try to
increase up your portfolio risk in

order to try to produce above-average portfolio returns. While this
strategy may be successful on

occasion, it often delivers mixed results.

\nI hope I have helped you! If you are not satisfied, please meet a
financial advisor in our nearest Auxilium

branch who can help you more.

If you need me, I'm here!'''


            if b1['text'] == 'Hi! Yes, I am looking at retirement
now':

                T.delete(1.0, t.END)

                T.insert(t.END, ad1)

                b1['text'] = 'Sure!'


            elif b1['text'] == 'Sure!':

                T.delete(1.0, t.END)

                T.insert(t.END, ad2)

                b1['text'] = 'Can I add taxable accounts as well?'


            elif b1['text'] == 'Can I add taxable accounts as
well?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad3)

                b1['text'] = "What's the next step?"


            elif b1['text'] == "What's the next step?":

                T.delete(1.0, t.END)

                T.insert(t.END, ad4)

                b1['text'] = 'Tell me more!'


            elif b1['text'] == 'Tell me more!':
```

```
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'When am I not good to go for
retiring?'

            elif b1['text'] == 'When am I not good to go for
retiring?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'Okay... What next?'

            elif b1['text'] == 'Okay... What next?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'What other tip do you have?'

            elif b1['text'] == 'What other tip do you have?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad8)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'

            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, '''No problem. See you soon!


    Credits: Investopedia''')
                b1['text'] = ''


        def backclick():
            ad1 = '''Great! You have come to the right place.
Retirement planning is a long but useful

process that would help you have a comfortable and secure
retirement. I am only here to give you
```

advice on how to go about it but I'm afraid I am not programmed to help you with respect to your

situation. I hope that's alright with you. So, shall we begin our journey?'''

ad2 = '''The first step to planning your retirement is to assess your situation. An honest

assessment of where you are financially is pivotal to planning your retirement. Begin by counting

how much you have accumulated in accounts earmarked for retirement. This includes balances in IRAs

(individual retirement accounts) and workplace retirement plans.

\nA tip: Eliminate debt, especially high-interest debts such as credit cards. It is crucial to get

your finances under control. This should be done at least 10 years before retiring.'''

ad3 = '''Yes, you can include taxable accounts if you would be using them specifically for retirement.'''

ad4 = '''The next step is to identify your sources of income. Additional income can come from a many

places outside of savings. You should consider that money as well. If you are covered by a pension plan, great!

You can add the monthly income from that.'''

ad5 = '''What are your retirement goals? Would you like to travel around the world or would you like to

live a peaceful life in a modest house in the hills? Do consider your retirement goals while planning.

Here's a tip: Prepare a budget every month to estimate regular expenditures such as housing, food and

dining out, and leisure activities. Do consider health and medical expenses such as life insurance,

medicines, and doctor’s visits into your budget.'''

ad6 = '''If you are at least 8-10 years before retirement, try to increase your savings rate and cut back

```
on unnecessary spending. It is necessary. If you have many debts and
you are considering retirement

now, I'm sorry but you are not financially good for retirement.'''


            ad7 = '''You are not good to go for retirement if you
are struggling to pay current bills, nor accounting

for inflation in expenses, have high level of debts, no plans for
future major expenses, no monthly and

long-term financial plan.'''


            ad8 = '''Finally, you also need to assess your risk
tolerance for investments. A bear market (when a market

experiences prolonged price declines) with only a few years left for
retirement can hurt you chances of retiring at

the right time. Retirement portfolios at this stage should focus
primarily on high-quality, dividend-paying

stocks and investment-grade bonds to produce both conservative
growth and income.

\nOne guideline suggests that investors should subtract their age
from 110 to determine how much to

invest in stocks. A 70-year-old, for example, would target an
allocation of 40% stocks and 60% bonds.

\nBe aware that if you’re behind on your savings, do not try to
increase up your portfolio risk in

order to try to produce above-average portfolio returns. While this
strategy may be successful on

occasion, it often delivers mixed results.

\nI hope I have helped you! If you are not satisfied, please meet a
financial advisor in our nearest Auxilium

branch who can help you more.

If you need me, I'm here!'''


            if b1['text'] == 'Hi! Yes, I am looking at retirement
now':

                new_window.destroy()


            elif b1['text'] == 'Sure!':
```

```
                T.delete(1.0, t.END)
                T.insert(t.END, 'Hey there! You needed help?')
                b1['text'] = 'Hi! Yes, I am looking at retirement
now'

            elif b1['text'] == 'Can I add taxable accounts as
well?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'

            elif b1['text'] == "What's the next step?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = 'Can I add taxable accounts as well?'

            elif b1['text'] == 'Tell me more!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'Can I add taxable accounts as well?'

            elif b1['text'] == 'When am I not good to go for
retiring?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'Tell me more!'

            elif b1['text'] == 'Okay... What next?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'When am I not good to go for
retiring?'

            elif b1['text'] == 'What other tip do you have?':
```

```
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'Okay... What next?'

            elif b1['text'] == 'What other tip do you have?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'

            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'What other tip do you have?'

            elif b1['text'] == '':
                T.delete(1.0, t.END)
                T.insert(t.END, ad8)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'

        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
        new_window.resizable(width = False, height = False)
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))

        l = t.Label(new_window, text = 'Retirement', bg =
'lightblue')
        l.grid(row = 0, column = 0)
        l.config(font = ('Comic Sans', 14))
        T = st.ScrolledText(new_window)
        T.grid(row = 1, column = 0)
```

```
        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'

        b1 = t.Button(new_window, text = 'Hi! Yes, I am looking at
retirement now', command = lambda: bclick(), font = ('Comic Sans',
12))

        b1.grid(row = 3, column = 0, sticky = 'nesw')

        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))

        b3.grid(row = 4, column = 0, sticky = 'nesw')

        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))

        b2.grid(row = 5, column = 0, sticky = 'nesw')

        T.insert(-1.-1, ad)

        new_window.mainloop()


    def college():

        from tkinter import scrolledtext as st

        def bclick():

            ad1 = '''Awesome! You have come to the right place.
College is indeed expensive but it would be cakewalk

if you set certain boundaries and rules for yourself, and start
planning for college early. I am only here to

give you advice on how to go about it but I'm afraid I am not
programmed to help you with respect to your situation.

I hope that's alright with you. So, shall we begin our journey?'''


            ad2 = '''Firstly, Make a list of all the colleges you
want to attend, create two lists - one list containing

colleges you want to apply to but have some or little chance of
getting in (this includes

highly competitive schools) and the other list containing schools
where you have a high chance of getting into.

Choose your colleges wisely, considering financial constraints.'''


            ad3 = '''Next, you should research scholarships and
grants. You can find good scholarships on
```

https://www.scholarshipportal.com/

\nA money-saving strategy that does not require postponing college is to apply to schools where you have

unique characteristics they seek. For example, your ethnic background or low income.'''

ad4 = '''Keep in mind that housing and other living costs will vary by location. If you choose to live

off-campus, your living expenses are typically much less.'''

ad5 = '''Another option is to work and study. Make your after-school and summer jobs count by going after

high-paying work. To find high-paying work—particularly in the summer when you may be free during

business hours, seek out office jobs through temp agencies. If you cannot get a high-paying job, get a job

that will keep your living expenses down.

\nIf you are still in high school, start working now and save all your pay checks for college. Presumably,

you are still living at home, which is low cost. You probably do not have high living expenses eating into

your earnings as you will later on. Check if your high school has a program that will allow you to leave

school at noon every day to go to work during your senior year. '''

ad6 = '''Great question! I also have a few tips for managing finances while in college. If you are graduating early

i.e., before 18 years of age, your parents will take care of your finances till you turn 18. If you are

graduating when you are 18 or older, you would have to do that yourself. My first tip is to create a monthly budget.

In fact, Auxilium provides a budgeting and expenses tracking feature which would be very handy while in college.'''

ad7 = '''Tip #2: Minimise Student Debt. There are many ways in which you can minimise debt so that you don't

graduate from college broke.

1. Do not use your financial aid to fund things like a party in your dorm. You should spend on the right

things. You should be aware of what you should use your money for. tuition, books, housing, and maybe

food plans – not social outings, parties & new clothes.

\n2. Borrow Only What's Required. Not every student heads off to school with a fully-funded college trust.

If you need to take out student loans, remind yourself that the amount borrowed should be commensurate

with the type of salary available once a degree is obtained. Even if your you do choose to borrow money

for school, it should be for school.

\n3. Fund Extras with a Job. Work-study positions usually offer the flexibility a student needs with the

convenience of location, while off-campus positions frequently pay more.

\n4. Funnel Extra Earnings to Loan Payments. While loans technically aren't due until after graduation,

paying them off while in school can help you save serious money when it comes to long-term interest.

You should be an expert at exploring the ways your educational status can save money. Vendors, local

venues, restaurants, and services near college campuses often offer student discounts that could save

you big money during the first year.'''

ad8 = '''Credit Card Companies expect freshmen to be careless with credit cards, racking up late

fees and high interest payments. They often lure students in with college-centric offers, such as the

promise of free concert tickets or free college swag. Freshmen should never sign up for a student credit

card on a whim. Instead, you can talk about the pros and cons of different cards, set a reasonably low

spending limit, and look for cards with points or cash back rewards.'''

ad9 = '''My final advice is - avoid full-price textbooks. Textbooks are the budget breakers of college

students everywhere, Your school’s bookstore is basically like a convenience store. It’s there, it’s easy to

get to if you live on campus, and it has what you need. The trouble is you might not be

getting the best price if you exclusively buy books, both used and new, from the bookstore. The books

can be expensive. Look beyond the bookstore.

Tools like Bookfinder.com or 'Capital One Shopping' search the web for the titles you need and

show you which store has them for a better price.

With the exception of rare books, used books are pretty much always going to be cheaper than new

ones. Unless you need to buy the most recent edition of a book and the most recent edition was only

published a month ago, you can get by with a second-hand copy. Many websites sell used textbooks.

Some popular examples include:

1. Amazon

2. Chegg

3. eCampus.com

4. ValoreBooks

5. AbeBooks.com

6. CampusBooks

7. Barnes & Noble

8. sellbackyourBook.com

If you’re shopping online for your books, pay attention to the notes included by the seller.

Some used books have a lot of highlighting, underlining, or notes scribbled in the margins. I advise you to rent instead of buying.'''

ad10 = '''The other option is to use the college library or go digital. You can join the 'Open Textbook

Library' (https://open.umn.edu/opentextbooks) The library consists of nearly 800 titles that have been

```
faculty-reviewed and are available to students for free. Some
professors even

assign books exclusively from the library. But even if they don’t,
you might find the book you need in

the library without any effort from your instructor.

\nI hope I have helped you! If you are not satisfied, please meet a
financial advisor in our nearest Auxilium

branch who can help you more. If you need me, I'm here!'''



            if b1['text'] == 'Hi! Yes, I need help with funding
college':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'


            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = "Okay, what's the next step?"


            elif b1['text'] == "Okay, what's the next step?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'Do you have a tip?'


            elif b1['text'] == 'Do you have a tip?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = "What's the other option?"


            elif b1['text'] == "What's the other option?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
```

```
                b1['text'] = 'How do I manage finances while in
college?'


            elif b1['text'] == 'How do I manage finances while in
college?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text']= 'Tell me more!'


            elif b1['text'] == 'Tell me more!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Okay... What about using credit
cards?'


            elif b1['text'] == 'Okay... What about using credit
cards?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad8)
                b1['text'] = 'What other advice do you have?'


            elif b1['text'] == 'What other advice do you have?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad9)
                b1['text'] = 'Another option?'


            elif b1['text'] == 'Another option?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad10)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
```

```
                T.insert(t.END, '''No problem. See you soon!


    Credits: Investopedia, Money Crashers''')
                b1['text'] = ''


        def backclick():
            ad1 = '''Awesome! You have come to the right place.
College is indeed expensive but it would be cakewalk

if you set certain boundaries and rules for yourself, and start
planning for college early. I am only here to

give you advice on how to go about it but I'm afraid I am not
programmed to help you with respect to your situation.

I hope that's alright with you. So, shall we begin our journey?'''


            ad2 = '''Firstly, Make a list of all the colleges you
want to attend, create two lists - one list containing

colleges you want to apply to but have some or little chance of
getting in (this includes

highly competitive schools) and the other list containing schools
where you have a high chance of getting into.

Choose your colleges wisely, considering financial constraints.'''


            ad3 = '''Next, you should research scholarships and
grants. You can find good scholarships on

https://www.scholarshipportal.com/

\nA money-saving strategy that does not require postponing college
is to apply to schools where you have

unique characteristics they seek. For example, your ethnic
background or low income.'''


            ad4 = '''Keep in mind that housing and other living
costs will vary by location. If you choose to live

off-campus, your living expenses are typically much less.'''


            ad5 = '''Another option is to work and study. Make your
after-school and summer jobs count by going after
```

high-paying work. To find high-paying work—particularly in the summer when you may be free during

business hours, seek out office jobs through temp agencies. If you cannot get a high-paying job, get a job

that will keep your living expenses down.

\nIf you are still in high school, start working now and save all your pay checks for college. Presumably,

you are still living at home, which is low cost. You probably do not have high living expenses eating into

your earnings as you will later on. Check if your high school has a program that will allow you to leave

school at noon every day to go to work during your senior year. '''

ad6 = '''Great question! I also have a few tips for managing finances while in college. If you are graduating early

i.e., before 18 years of age, your parents will take care of your finances till you turn 18. If you are

graduating when you are 18 or older, you would have to do that yourself. My first tip is to create a monthly budget.

In fact, Auxilium provides a budgeting and expenses tracking feature which would be very handy while in college.'''

ad7 = '''Tip #2: Minimise Student Debt. There are many ways in which you can minimise debt so that you don't

graduate from college broke.

1. Do not use your financial aid to fund things like a party in your dorm. You should spend on the right

things. You should be aware of what you should use your money for. tuition, books, housing, and maybe

food plans – not social outings, parties & new clothes.

\n2. Borrow Only What's Required. Not every student heads off to school with a fully-funded college trust.

If you need to take out student loans, remind yourself that the amount borrowed should be commensurate

with the type of salary available once a degree is obtained. Even if your you do choose to borrow money

for school, it should be for school.

\n3. Fund Extras with a Job. Work-study positions usually offer the flexibility a student needs with the

convenience of location, while off-campus positions frequently pay more.

\n4. Funnel Extra Earnings to Loan Payments. While loans technically aren’t due until after graduation,

paying them off while in school can help you save serious money when it comes to long-term interest.

You should be an expert at exploring the ways your educational status can save money. Vendors, local

venues, restaurants, and services near college campuses often offer student discounts that could save

you big money during the first year.'''

ad8 = '''Credit Card Companies expect freshmen to be careless with credit cards, racking up late

fees and high interest payments. They often lure students in with college-centric offers, such as the

promise of free concert tickets or free college swag. Freshmen should never sign up for a student credit

card on a whim. Instead, you can talk about the pros and cons of different cards, set a reasonably low

spending limit, and look for cards with points or cash back rewards.'''

ad9 = '''My final advice is - avoid full-price textbooks. Textbooks are the budget breakers of college

students everywhere, Your school’s bookstore is basically like a convenience store. It’s there, it’s easy to

get to if you live on campus, and it has what you need. The trouble is you might not be

getting the best price if you exclusively buy books, both used and new, from the bookstore. The books

can be expensive. Look beyond the bookstore.

Tools like Bookfinder.com or 'Capital One Shopping' search the web for the titles you need and

show you which store has them for a better price.

With the exception of rare books, used books are pretty much always going to be cheaper than new

ones. Unless you need to buy the most recent edition of a book and the most recent edition was only

published a month ago, you can get by with a second-hand copy. Many websites sell used textbooks.

Some popular examples include:

1. Amazon

2. Chegg

3. eCampus.com

4. ValoreBooks

5. AbeBooks.com

6. CampusBooks

7. Barnes & Noble

8. sellbackyourBook.com

If you're shopping online for your books, pay attention to the notes included by the seller.

Some used books have a lot of highlighting, underlining, or notes scribbled in the margins. I

advise you to rent instead of buying.'''

ad10 = '''The other option is to use the college library or go digital. You can join the 'Open Textbook

Library' (https://open.umn.edu/opentextbooks) The library consists of nearly 800 titles that have been

faculty-reviewed and are available to students for free. Some professors even

assign books exclusively from the library. But even if they don't, you might find the book you need in

the library without any effort from your instructor.

\nI hope I have helped you! If you are not satisfied, please meet a financial advisor in our nearest Auxilium

branch who can help you more. If you need me, I'm here!'''

```
            if b1['text'] == 'Hi! Yes, I need help with funding
college':
                new_window.destroy()

            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
                T.insert(t.END, 'Hey there! You needed help?')
                b1['text'] = 'Hi! Yes, I need help with funding
college'

            elif b1['text'] == "Okay, what's the next step?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'

            elif b1['text'] == 'Do you have a tip?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = "Okay, what's the next step?"

            elif b1['text'] == "What's the other option?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'Do you have a tip?'

            elif b1['text'] == 'How do I manage finances while in
college?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = "What's the other option?"

            elif b1['text'] == 'Tell me more!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
```

```
                b1['text'] = 'How do I manage finances while in
college?'


            elif b1['text'] == 'Okay... What about using credit
cards?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'Tell me more!'


            elif b1['text'] == 'What other advice do you have?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Okay... What about using credit
cards?'


            elif b1['text'] == 'Another Option?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad8)
                b1['text'] = 'What other advice do you have?'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad9)
                b1['text'] = 'Another Option?'


            elif b1['text'] == '':
                T.delete(1.0, t.END)
                T.insert(t.END, ad10)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
```

```
        new_window.resizable(width = False, height = False)

        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))


        l = t.Label(new_window, text = 'College Funding',
bg='lightblue')

        l.grid(row = 0, column = 0)

        l.config(font = ('Comic Sans', 14))

        T = st.ScrolledText(new_window)

        T.grid(row = 1, column = 0)

        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'

        b1 = t.Button(new_window, text = 'Hi! Yes, I need help with
funding college', command = lambda: bclick(), font = ('Comic Sans',
12))

        b1.grid(row = 3, column = 0, sticky = 'nesw')

        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))

        b3.grid(row = 4, column = 0, sticky = 'nesw')

        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))

        b2.grid(row = 5, column = 0, sticky = 'nesw')

        T.insert(-1.-1, ad)

        new_window.mainloop()


    def debt():

        from tkinter import scrolledtext as st

        def bclick():

            ad1 = '''So, you need help with managing debts. Debts
can indeed be a burden. But do

not worry, you are in the right place. I am only here to give you
advice on how to go about

it but I'm afraid I am not programmed to help you with respect to
your situation. I hope
```

that's alright with you. So, shall we begin our journey?'''

ad2 = '''To get out of debt you need to have a solid plan and it should be executed

properly. The first step to get out is to collect your data. Here's what you need to get:

> Your most recent bill statements for all credit cards and loans, including student loans.

> Your credit reports, so you can check for accuracy and identify all recorded debts.

> Your credit score to find out whether you're eligible to lower your interest rates or for a debt

consolidation loan.

Once you have your data in hand, make a list of all your debts. Be sure to

include the creditor's name, balance, minimum monthly payment and the interest rate'''

ad3 = '''Next, list how much you need to pay in order to zero-out the debt's balance within

three years or whatever your target time frame is. Remember to include items that are not listed on

your credit reports, such as family loans, medical bills and recurring bills, such as groceries and

utilities.'''

ad4 = '''At this point, you have to start creating your plan. A good way to approach a debt

pay-off plan is to take the total payoff number you calculated in the 2nd step and use it as a goal

to work towards by:

> Totalling the three-year or your chosen timeframe pay-off amount for all your credit cards.

> Adding the monthly payments for all other debts.

> Writing down the result as "Your Total Monthly Payment."

You do not have to pay the exact minimum amount. I advise you to pay more than the

monthly minimum payment on your credit card or loan.

Once you have your plan:

Determine if you can afford to pay the Total Monthly Payment until your debt is paid off.

- If not, contact a credit counselling agency and/or bankruptcy attorney for advice. Remember though, bankruptcy has a huge impact on your credit score, and if you're able to work out a payment plan with your creditors, it can be avoided.

- If doable, pick one debt to pay off first. Start with paying off the debt with the highest interest rate or lowest balance. That's your "target debt." Paying your target debt off first is known as the "debt snowball" or "avalanche" method.

Pay as much as possible toward target debt until that debt is paid off. Then choose a new target debt and pay extra toward that one, and so on.'''

ad5 = '''That is a good question! A few signs of a debt trap are:

1. Your EMIs exceed 50% of your income

2. Your fixed expenses are more than 70% of income

3. You have exhausted your credit card limit

4. You cannot afford to put aside money for savings'''

ad6 = '''Here are a few steps to get rid of a debt-trap:

1. Determine your problem and analyse it.

2. Make a budget, prioritise, and stick to the budget.

- Create a priority list of all your needs.

- Make debt repayment your first priority as that can have a positive and long-term effect on your financial situation.

- Refrain from indulging in non-essential or even on semi-essential items at least till you are back on track.

3. Opt for debt consolidation

- When you consolidate your debt, you are combining multiple debts into a single debt.

Consolidating your debt also allows you to opt for favourable payoff terms, lower interest

rates and lower EMIs.

4. Pay of expensive loans first.

5. Do not take any extra loans. This would push you deeper into the trap.

6. Increase your income

- You can increase your income by taking up part-time jobs or freelance jobs.

'''

ad7 = '''To prevent falling into a debt trap in future, here are a few things you should

keep in mind:

1. Lower your interest rates. You can do that by getting a credit card with a lower interest

rate. The other option is to get a bank transfer credit card with a low interest rate or 0%

intro rate. Third option is to get a loan with low interest rate, and finally, consolidate

loans.

2. Create an emergency fund. The suggested amount to have in an emergency fund is three to six

months' worth of expenses. If that amount isn't possible, aim for one months' worth, which is

still a good starting point.

3. Budget out your monthly expenses you can better track where your money is going and where you

can afford to spend it. In fact, Auxilium has a feature to track expenses and prepare a budget.

It would be helpful here.

4. The best way to keep your spending under control is to pay your credit card balance as you go.

So, if you make a purchase with your credit card, say to earn rewards, send your payment the

```
   very next day!
I hope I have helped you! If you are not satisfied, please meet a
financial advisor in our nearest
Auxilium branch who can help you more. If you need me, I'm here!'''


            if b1['text'] == 'Hi! Yes, I need help with managing
debts':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'


            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = 'What next?'


            elif b1['text'] == 'What next?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'Tell me more!'


            elif b1['text'] == 'Tell me more!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'How do I know if I am in a debt-trap?'


            elif b1['text'] == 'How do I know if I am in a debt-
trap?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'How do I get out of a debt trap?'


            elif b1['text'] == 'How do I get out of a debt trap?':
                T.delete(1.0, t.END)
```

```
                T.insert(t.END, ad6)
                b1['text'] = 'How can I prevent a debt-trap in
future?'


            elif b1['text'] == 'How can I prevent a debt-trap in
future?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, '''No problem. See you soon!


    Credits: Money Tap, credit.org, HDFC, Central Bank''')
                b1['text'] = ''


        def backclick():
            ad1 = '''So, you need help with managing debts. Debts
can indeed be a burden. But do
not worry, you are in the right place. I am only here to give you
advice on how to go about
it but I'm afraid I am not programmed to help you with respect to
your situation. I hope
that's alright with you. So, shall we begin our journey?'''


            ad2 = '''To get out of debt you need to have a solid
plan and it should be executed
properly. The first step to get out is to collect your data. Here's
what you need to get:
> Your most recent bill statements for all credit cards and loans,
including student loans.
> Your credit reports, so you can check for accuracy and identify
all recorded debts.
```

> Your credit score to find out whether you’re eligible to lower your interest rates or for a debt

consolidation loan.

Once you have your data in hand, make a list of all your debts. Be sure to

include the creditor’s name, balance, minimum monthly payment and the interest rate'''

ad3 = '''Next, list how much you need to pay in order to zero-out the debt’s balance within

three years or whatever your target time frame is. Remember to include items that are not listed on

your credit reports, such as family loans, medical bills and recurring bills, such as groceries and

utilities.'''

ad4 = '''At this point, you have to start creating your plan. A good way to approach a debt

pay-off plan is to take the total payoff number you calculated in the 2nd step and use it as a goal

to work towards by:

> Totalling the three-year or your chosen timeframe pay-off amount for all your credit cards.

> Adding the monthly payments for all other debts.

> Writing down the result as "Your Total Monthly Payment."

You do not have to pay the exact minimum amount. I advise you to pay more than the

monthly minimum payment on your credit card or loan.

Once you have your plan:

Determine if you can afford to pay the Total Monthly Payment until your debt is paid off.

- If not, contact a credit counselling agency and/or bankruptcy attorney for advice. Remember

though, bankruptcy has a huge impact on your credit score, and if you’re able to work out

a payment plan with your creditors, it can be avoided.

- If doable, pick one debt to pay off first. Start with paying off the debt with the highest

interest rate or lowest balance. That's your "target debt." Paying your target debt off

first is known as the "debt snowball" or "avalanche" method.

Pay as much as possible toward target debt until that debt is paid off. Then choose a new

target debt and pay extra toward that one, and so on.'''

ad5 = '''That is a good question! A few signs of a debt trap are:

1. Your EMIs exceed 50% of your income

2. Your fixed expenses are more than 70% of income

3. You have exhausted your credit card limit

4. You cannot afford to put aside money for savings'''

ad6 = '''Here are a few steps to get rid of a debt-trap:

1. Determine your problem and analyse it.

2. Make a budget, prioritise, and stick to the budget.

- Create a priority list of all your needs.

- Make debt repayment your first priority as that can have a positive and long-term effect

on your financial situation.

- Refrain from indulging in non-essential or even on semi-essential items at least till

you are back on track.

3. Opt for debt consolidation

- When you consolidate your debt, you are combining multiple debts into a single debt.

Consolidating your debt also allows you to opt for favourable payoff terms, lower interest

rates and lower EMIs.

4. Pay of expensive loans first.

5. Do not take any extra loans. This would push you deeper into the trap.

6. Increase your income

- You can increase your income by taking up part-time jobs or freelance jobs.

'''

ad7 = '''To prevent falling into a debt trap in future, here are a few things you should

keep in mind:

1. Lower your interest rates. You can do that by getting a credit card with a lower interest

rate. The other option is to get a bank transfer credit card with a low interest rate or 0%

intro rate. Third option is to get a loan with low interest rate, and finally, consolidate

loans.

2. Create an emergency fund. The suggested amount to have in an emergency fund is three to six

months' worth of expenses. If that amount isn't possible, aim for one months' worth, which is

still a good starting point.

3. Budget out your monthly expenses you can better track where your money is going and where you

can afford to spend it. In fact, Auxilium has a feature to track expenses and prepare a budget.

It would be helpful here.

4. The best way to keep your spending under control is to pay your credit card balance as you go.

So, if you make a purchase with your credit card, say to earn rewards, send your payment the

very next day!

I hope I have helped you! If you are not satisfied, please meet a financial advisor in our nearest

Auxilium branch who can help you more. If you need me, I'm here!'''

if b1['text'] == 'Hi! Yes, I need help with managing debts':

new_window.destroy()

```
            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
                T.insert(t.END, 'Hey there! You needed help?')
                b1['text'] = 'Hi! Yes, I need help with managing
debts'

            elif b1['text'] == 'What next?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'

            elif b1['text'] == 'Tell me more!':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = 'What next?'

            elif b1['text'] == 'How do I know if I am in a debt-
trap?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'Tell me more!'

            elif b1['text'] == 'How do I get out of a debt trap?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'How do I know if I am in a debt-trap?'

            elif b1['text'] == 'How can I prevent a debt-trap in
future?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'How do I get out of a debt trap?'
```

```
            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'How can I prevent a debt-trap in
future?'


            elif b1['text'] == '':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
        new_window.resizable(width = False, height = False)
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))


        l = t.Label(new_window, text = 'Debts', bg = 'lightblue')
        l.grid(row = 0, column = 0)
        l.config(font = ('Comic Sans', 14))
        T = st.ScrolledText(new_window)
        T.grid(row = 1, column = 0)
        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'
        b1 = t.Button(new_window, text = 'Hi! Yes, I need help with
managing debts', command = lambda: bclick(), font = ('Comic Sans',
12))
        b1.grid(row = 3, column = 0, sticky = 'nesw')
        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))
        b3.grid(row = 4, column = 0, sticky = 'nesw')
```

```
        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))
        b2.grid(row = 5, column = 0, sticky = 'nesw')
        T.insert(-1.-1, ad)
        new_window.mainloop()


    def job():
        from tkinter import scrolledtext as st
        def bclick():
            ad1 = '''Congratulations on landing on your first job!
You are probably experiencing a
    steady inflow of money each month and you don’t see any real
liabilities as you are either living
    with your family or friends. With greater earning potential
comes greater financial responsibility,
    from maximizing your employee benefits, such as life insurance
and retirement planning, to paying
    down your student and credit card debts to investing in an
emergency fund.
    So shall we begin our journey?'''


            ad2 = '''The most important thing you must do after
landing on your first job is to
    prepare a saving and investment budget. This means that from the
pay you take home, you must
    set aside a fixed amount that you save and then invest wisely.
Most people prepare a budget for
    expenses.
    It is not about how much you save; it is more about starting to
save and investing those savings
    in an ordered manner. Even it means starting with 1% if your
income, and then gradually scaling
    up to 5-10%, most people can easily save 5-10% of their income
without suffering financial
    problems. You should eventually try saving 25% of your
income.'''
```

ad3 = '''To maintain a consistent savings rate, you can automate the savings process. You

can do this by:

- Direct Deposit.

If your employer allows, send a portion of your pay check deposit to your savings account every payday.

- Recurring Bank Transfer.

Schedule a recurring checking-to-savings transfer every payday or on the same day each month.

- Automated Savings App.

Use an automated savings app such as Acorns or Digit to periodically draw funds from your

checking accounts and deposit them into your savings account. Apps like Digit use

sophisticated algorithms to determine how much you can afford to save each month. If you

prefer, you can manually set and change your savings rate too. Some apps – and some banks

– have round-up-the-change features that round each debit card purchase up to the nearest

____ (1 unit of the currency of your country) and transfer the difference to savings.'''

ad4 = '''You should also set up a direct deposit. If your employer offers free direct deposit,

then set yours up before your official start date. A recurring direct deposit is the easiest way to

avoid monthly maintenance fees on entry-level bank accounts that don’t already waive fees, and the

convenience is second to none.

Building an emergency fund should also be your top savings priority. A robust emergency

fund is sufficient to cover at least three months' expenses at your current spending levels, but the

ideal amount is six months’ expenses. Develop a plan to address any high-interest debt. Check out

the Auxilium interactive manuals for debts for more information.'''

ad5 = '''Yes, whether your employer sponsors a tax-advantaged deferred compensation plan or not,

you can always open an individual retirement account (IRA) on your own through a platform like

Betterment. Most taxpayers choose one of two IRA options:

Traditional IRA

- A traditional IRA allows you to deduct the amount of your investments from your tax return.

Every time you make an investment in a traditional IRA, your taxable income gets reduced by

the amount invested by you. This means that you will able to grow your investments without

having to pay taxes on the earnings from them, until you withdraw them at the time of your

retirement.

Roth IRA

- A Roth IRA allows you to invest your money after deducting the taxes and then earn tax-free

returns till the time of your retirement. Unlike the traditional IRA, you won’t be able to claim

deductions from your taxable income at the time of making the investment in Roth IRA. However, the

advantage is that you don’t have to pay any tax at the time of withdrawal of your investment.

Simple IRA

- The Simple IRA is meant for self-employed individuals or small business owners. Same taxation

rules are applicable for simple IRAs as it is for traditional IRAs. A simple IRA allows the small

business owners to contribute towards creating a retirement corpus for themselves as well as their

employees.'''

ad6 = '''Take full advantage of employee benefits. Depending on the type and size of company

you start working for, you might be offered next to nothing in the way of benefits or you might be

offered everything from health insurance to life insurance to disability income insurance and more.

Here are some things to consider while thinking about them:

1. Life Insurance

With the likelihood that you will change jobs several times in your 20s and 30s, getting life

insurance through your employer may not be your best option. es, it may be your cheapest option

— your employer might offer life insurance with a death benefit of one, two, or three times your

annual salary as a free or inexpensive employee benefit — but if you leave the company, you

usually will not be able to take your policy with you.

While you may not need life insurance in your 20s in the sense that you may not yet have a spouse

or children who depend on your income, buying at least a small, inexpensive term life insurance

policy could be a smart move so that you know you have established a baseline of coverage,

regardless of what happens with your health or your job, if you do end up having dependents.

here are many types of life insurance available at different price points, so even if you do

not have much disposable income, you may be able to find an affordable policy that offers a

good value in terms of coverage, especially if you are a non-smoker with a healthy weight.

2. Disability Income Insurance

Disability income insurance provides can protect a portion of your income if you are too sick

or injured to work. You may not be able to take an employer-sponsored disability insurance

policy with you when you change jobs, and the coverage your employer offers might not provide

as much financial protection as you would like. Private disability insurance may provide more

liberal benefits, and you can customize the policy to your needs. You can choose the amount

of coverage, within limits based on your income, and you can decide how long you would be

willing to wait before benefits would reap — for example, three months versus six months'''

ad7 = '''Most large employers will give you the ability to choose between several types

of health insurance plans. Each plan comes with its own benefits and drawbacks, depending on

your anticipated needs and healthcare costs. Type of plans:

PPO — A PPO, or preferred provider organization, is a health plan where you have the ability

to choose your own healthcare providers. If you choose a doctor in the insurer's

preferred provider network, you will get additional discounts and savings. If you choose

a provider outside of the network, you will get reduced benefits. PPOs give you more

flexibility to see whichever doctor you want, whenever you want, often without a referral.

You will pay less when you choose an in-network provider, but you can choose to see an

out-of-network provider, although you will pay more than if you received your care

in-network.

HMO — An HMO, or health maintenance organisation, is a healthcare provider in which your

insurance company is your primary healthcare provider. Your doctors are employees of the

same company that offers your insurance. You have fewer options when choosing a doctor,

and are generally required to receive all care, outside of emergency situations, through

the HMO in order to receive insurance coverage.

If you choose an HMO, or health maintenance organization, your premiums may be lower than those

```
    for a PPO, but you will lose flexibility: you will have to see
your designated primary care

    doctor first when you need a referral to a specialist. HMOs also
limit you to in-network

    providers, but the price you will pay for a doctor visit is
fixed.

    If you choose a PPO, your employer may offer a choice between a
high deductible plan with

    low premiums, which can save you money if you are healthy and
anticipate few doctor visits, and

    a low deductible plan with higher premiums, which may save you
money if you have a pre-existing

    condition or are a woman planning to get pregnant in the
upcoming year.

    For more information on insurance, refer to the Auxilium
Interactive Financial Advice User Manual

    on insurance.


    I hope I have helped you! If you are not satisfied, please meet
a financial advisor in our nearest

    Auxilium branch who can help you more. If you need me, I'm
here!'''


            if b1['text'] == 'Hi! Yes, I want to know how to go
about managing finances for first job':

                T.delete(1.0, t.END)

                T.insert(t.END, ad1)

                b1['text'] = 'Sure!'


            elif b1['text'] == 'Sure!':

                T.delete(1.0, t.END)

                T.insert(t.END, ad2)

                b1['text'] = 'How do I maintain a consistent savings
rate later?'


            elif b1['text'] == 'How do I maintain a consistent
savings rate later?':
```

```
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'What else should I do?'

            elif b1['text'] == 'What else should I do?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'Should I start saving for retirement?'

            elif b1['text'] == 'Should I start saving for
retirement?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'Do you have any other tips?'

            elif b1['text'] == 'Do you have any other tip?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'What about health insurance?'

            elif b1['text'] == 'What about health insurance?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'

            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, '''No problem. See you soon!

    Credits: Financial Express, World Trips, Mass Mutual, Aditya
Birla Capital''')
                b1['text'] = ''
```

```
        def backclick():
            ad1 = '''Congratulations on landing on your first job! You are probably experiencing a
    steady inflow of money each month and you don’t see any real liabilities as you are either living
    with your family or friends. With greater earning potential comes greater financial responsibility,
    from maximizing your employee benefits, such as life insurance and retirement planning, to paying
    down your student and credit card debts to investing in an emergency fund.
    So shall we begin our journey?'''

            ad2 = '''The most important thing you must do after landing on your first job is to
    prepare a saving and investment budget. This means that from the pay you take home, you must
    set aside a fixed amount that you save and then invest wisely. Most people prepare a budget for
    expenses.
    It is not about how much you save; it is more about starting to save and investing those savings
    in an ordered manner. Even it means starting with 1% if your income, and then gradually scaling
    up to 5-10%, most people can easily save 5-10% of their income without suffering financial
    problems. You should eventually try saving 25% of your income.'''

            ad3 = '''To maintain a consistent savings rate, you can automate the savings process. You
    can do this by:
     - Direct Deposit.
       If your employer allows, send a portion of your pay check deposit to your savings account every payday.
     - Recurring Bank Transfer.
```

Schedule a recurring checking-to-savings transfer every payday or on the same day each month.

- Automated Savings App.

Use an automated savings app such as Acorns or Digit to periodically draw funds from your

checking accounts and deposit them into your savings account. Apps like Digit use

sophisticated algorithms to determine how much you can afford to save each month. If you

prefer, you can manually set and change your savings rate too. Some apps – and some banks

– have round-up-the-change features that round each debit card purchase up to the nearest

____ (1 unit of the currency of your country) and transfer the difference to savings.'''

ad4 = '''You should also set up a direct deposit. If your employer offers free direct deposit,

then set yours up before your official start date. A recurring direct deposit is the easiest way to

avoid monthly maintenance fees on entry-level bank accounts that don’t already waive fees, and the

convenience is second to none.

Building an emergency fund should also be your top savings priority. A robust emergency

fund is sufficient to cover at least three months' expenses at your current spending levels, but the

ideal amount is six months’ expenses. Develop a plan to address any high-interest debt. Check out

the Auxilium interactive manuals for debts for more information.'''

ad5 = '''Yes, whether your employer sponsors a tax-advantaged deferred compensation plan or not,

you can always open an individual retirement account (IRA) on your own through a platform like

Betterment. Most taxpayers choose one of two IRA options:

Traditional IRA

- A traditional IRA allows you to deduct the amount of your investments from your tax return.

Every time you make an investment in a traditional IRA, your taxable income gets reduced by

the amount invested by you. This means that you will able to grow your investments without

having to pay taxes on the earnings from them, until you withdraw them at the time of your

retirement.

Roth IRA

- A Roth IRA allows you to invest your money after deducting the taxes and then earn tax-free

returns till the time of your retirement. Unlike the traditional IRA, you won't be able to claim

deductions from your taxable income at the time of making the investment in Roth IRA. However, the

advantage is that you don't have to pay any tax at the time of withdrawal of your investment.

Simple IRA

- The Simple IRA is meant for self-employed individuals or small business owners. Same taxation

rules are applicable for simple IRAs as it is for traditional IRAs. A simple IRA allows the small

business owners to contribute towards creating a retirement corpus for themselves as well as their

employees.'''

ad6 = '''Take full advantage of employee benefits. Depending on the type and size of company

you start working for, you might be offered next to nothing in the way of benefits or you might be

offered everything from health insurance to life insurance to disability income insurance and more.

Here are some things to consider while thinking about them:

1. Life Insurance

With the likelihood that you will change jobs several times in your 20s and 30s, getting life

insurance through your employer may not be your best option. es, it may be your cheapest option

— your employer might offer life insurance with a death benefit of one, two, or three times your

annual salary as a free or inexpensive employee benefit — but if you leave the company, you

usually will not be able to take your policy with you.

While you may not need life insurance in your 20s in the sense that you may not yet have a spouse

or children who depend on your income, buying at least a small, inexpensive term life insurance

policy could be a smart move so that you know you have established a baseline of coverage,

regardless of what happens with your health or your job, if you do end up having dependents.

here are many types of life insurance available at different price points, so even if you do

not have much disposable income, you may be able to find an affordable policy that offers a

good value in terms of coverage, especially if you are a non-smoker with a healthy weight.

2. Disability Income Insurance

Disability income insurance provides can protect a portion of your income if you are too sick

or injured to work. You may not be able to take an employer-sponsored disability insurance

policy with you when you change jobs, and the coverage your employer offers might not provide

as much financial protection as you would like. Private disability insurance may provide more

liberal benefits, and you can customize the policy to your needs. You can choose the amount

of coverage, within limits based on your income, and you can decide how long you would be

willing to wait before benefits would reap — for example, three months versus six months'''

ad7 = '''Most large employers will give you the ability to choose between several types

of health insurance plans. Each plan comes with its own benefits and drawbacks, depending on

your anticipated needs and healthcare costs. Type of plans:

PPO — A PPO, or preferred provider organization, is a health plan where you have the ability

to choose your own healthcare providers. If you choose a doctor in the insurer's

preferred provider network, you will get additional discounts and savings. If you choose

a provider outside of the network, you will get reduced benefits. PPOs give you more

flexibility to see whichever doctor you want, whenever you want, often without a referral.

You will pay less when you choose an in-network provider, but you can choose to see an

out-of-network provider, although you will pay more than if you received your care

in-network.

HMO — An HMO, or health maintenance organisation, is a healthcare provider in which your

insurance company is your primary healthcare provider. Your doctors are employees of the

same company that offers your insurance. You have fewer options when choosing a doctor,

and are generally required to receive all care, outside of emergency situations, through

the HMO in order to receive insurance coverage.

If you choose an HMO, or health maintenance organization, your premiums may be lower than those

for a PPO, but you will lose flexibility: you will have to see your designated primary care

doctor first when you need a referral to a specialist. HMOs also limit you to in-network

providers, but the price you will pay for a doctor visit is fixed.

```
    If you choose a PPO, your employer may offer a choice between a
high deductible plan with

    low premiums, which can save you money if you are healthy and
anticipate few doctor visits, and

    a low deductible plan with higher premiums, which may save you
money if you have a pre-existing

    condition or are a woman planning to get pregnant in the
upcoming year.

    For more information on insurance, refer to the Auxilium
Interactive Financial Advice User Manual

    on insurance.


    I hope I have helped you! If you are not satisfied, please meet
a financial advisor in our nearest

    Auxilium branch who can help you more. If you need me, I'm
here!'''


            if b1['text'] == 'Hi! Yes, I want to know how to go
about managing finances for first job':

                new_window.destroy()


            elif b1['text'] == 'Sure!':

                T.delete(1.0, t.END)

                T.insert(t.END, 'Hey there! You needed help?')

                b1['text'] = 'Hi! Yes, I want to know how to go
about managing finances for first job'


            elif b1['text'] == 'How do I maintain a consistent
savings rate later?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad1)

                b1['text'] = 'Sure!'


            elif b1['text'] == 'What else should I do?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad2)
```

```
                b1['text'] = 'How do I maintain a consistent savings
rate later?'


            elif b1['text'] == 'Should I start saving for
retirement?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'What else should I do?'


            elif b1['text'] == 'Do you have any other tip?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'Should I start saving for retirement?'


            elif b1['text'] == 'What about health insurance?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'Do you have any other tips?'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'What about health insurance?'


            elif b1['text'] == '':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
        new_window.resizable(width = False, height = False)
```

```
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))


        l = t.Label(new_window, text = 'First Job', bg =
'lightblue')
        l.grid(row = 0, column = 0)
        l.config(font = ('Comic Sans', 14))
        T = st.ScrolledText(new_window)
        T.grid(row = 1, column = 0)
        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'
        b1 = t.Button(new_window, text = 'Hi! Yes, I want to know
how to go about managing finances for first job', command = lambda:
bclick(), font = ('Comic Sans', 12))
        b1.grid(row = 3, column = 0, sticky = 'nesw')
        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))
        b3.grid(row = 4, column = 0, sticky = 'nesw')
        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))
        b2.grid(row = 5, column = 0, sticky = 'nesw')
        T.insert(-1.-1, ad)
        new_window.mainloop()


    def insurance():
        from tkinter import scrolledtext as st
        def bclick():
            ad1 = '''You are in the right place! What is insurance?
    According to Investopedia, insurance is a contract, represented
by a policy, in which an individual
    or entity receives financial protection or reimbursement against
losses from an insurance company.
    There are many types of insurance policies. There are many types
of insurance such as life insurance,
```

motor insurance, Health insurance, travel insurance, property insurance etc.

Why do we need insurance?

1. Insurance ensures family's financial stability

No matter how much you have managed to save or what your monthly income is, an unexpected event

can burn a huge hole in your pocket or can simply jeopardize your family's financial future. For

example, if you do not have adequate life insurance, your family might have to go through

financial hardship if you were to meet with an untimely death. Though no amount of money can

replace the loss of loved ones, having life insurance would save them from going through financial

hardship.

2. Insurance reduces stress during difficult times.

The premium you pay to the insurance company is the price that guarantees that the insurance

company will cover the damage in case of an unforeseen event. And, that guarantee that your risk

is covered brings peace of mind. '''

ad2 = '''Definition of terms:

Policy:

A legal contract between you and the insurer. It details what risks are covered, under what

circumstances the insurer will make a payment to you, how much money and what type of benefit

you will receive if you make a claim.

Policyholder:

The insured or the person covered under the policy.

Coverage:

The amount of protection you have bought. It is also the maximum amount the insurance company

will pay you if you make a claim for loss or event covered by your policy.

Benefit:

The amount the insurer will pay you if the insurer accepts your claim.

Premium:

The amount you pay for the insurance.

Cash value:

This is the amount the insurer pays to the policyholder when a life insurance policy is

cancelled. It can also be an amount added to the death benefit and can be paid upon the

insured's death. This term is used with permanent life insurance policies.

Death benefit:

The amount the insurer will pay the beneficiary or beneficiaries upon the insured's death.

Claim:

This is the official notice to your insurer to be paid for a loss or event covered by your insurance policy.

Beneficiary:

This is the person or entity the insured names or assigns to receive the proceeds of the

policy. A beneficiary can be revocable (can be changed at any time without informing the

beneficiary) or irrevocable (can't be changed without the beneficiary's written permission).

Deductible:

The amount you agree to pay before the insurer pays the rest.

Exclusions:

Things that are not covered by your policy. For example, some health insurance policies may

exclude certain medical conditions you had before you applied for insurance or a travel

insurance policy may exclude claims if you travel to a high-risk country. This is why it

is important to read your policy thoroughly to check what it covers and what it doesn't

cover so that there will be no surprises when the time to claim comes.

Risk:

The probability or likelihood that an insured event, such as loss, injury or death, will

happen while the policy is in effect.

Rider:

It is a clause or term added to your insurance policy to provide protection. This has an additional

cost because it covers risks not covered in the basic policy.

Term:

The time period you are covered by your policy.

Grace period:

10 days immediately following the day you purchase an insurance policy, during which you may

cancel the policy for a full refund of any premiums paid'''

ad3 = '''When people buy insurance, they put their money into a pool with many others. Some of that

pool of money helps the policyholders who suffer a hardship in that period. The hardship can be related

to home, motor or business losses. You are covered only for losses written in your contract, and not for

predictable events. When a hardship occurs, a claim is made. This is an official request for the insurer to

pay for a covered loss. The insured's agent can assist in claiming benefits. Supporting documents/

evidence will be required, depending on the type of loss (for example, pictures of an injury or property

damage for an accident or property insurance claim, or a death certificate for a life insurance claim)

during investigation of claims.'''

ad4 = '''There are 3 important insurances:

1. Life Insurance:

Life insurance financially protects your family in case you die an early death. You pay a regular

premium to the insurance company for a specific number of years. In return the insurance

company pays a sum assured to your family if you die during the policy tenure.

There are two types of life insurance:

> Term – provides coverage for a specific amount of time. If the insured dies within the period

of coverage(and the premiums are paid), the beneficiary receives the death benefit as

stated in the policy. The coverage ends at the specified term. It can be renewed after

the term, however, the premium may increase since premiums may depend on the insured's

age.

> Permanent – it covers the insured throughout their lifetime (unless the insured fails to pay

the premiums).

There are two kinds:

o Whole life insurance – this guarantees that your premiums will not change as you

get older. This type of insurance has a guaranteed minimum

cash value and death benefit amount.

o Universal life insurance – this is a product combining life insurance and

investment.

2.Health Insurance:

Health insurance can help you pay for services that the provincial health care plan does not cover.

Some types can supplement your income if you suffer a major illness or injury. Other types can pay

for medical expenses if you become ill while on vacation.

Health insurance can be tricky to navigate. Managed care insurance plans require policyholders to

receive care from a network of designated healthcare providers for the highest level of coverage.

There are 4 main types of insurance plans:

- Health Maintenance Organization (HMO)

A Health Maintenance Organization, or HMO, provides employers or groups a way to take care of

all their employees' or members' health care needs with reduced costs by negotiating with

specific doctors, hospitals, and clinics. Your doctors are employees of the same company that

offers your insurance. You have fewer options when choosing a doctor, and are generally required

to receive all care, outside of emergency situations, through the HMO in order to receive insurance

coverage. The employee must use these specific providers for the reduced fees to be provided to

their medical insurance plan. In an HMO plan, you have the least flexibility but will likely have

the easiest claims experiences since the network takes care of putting in the claims for you.

- Preferred Provider Organization (PPO)

A PPO, or preferred provider organization, is a health plan where you have the ability to choose

your own healthcare providers. If you choose a doctor in the insurer's preferred provider network,

you will get additional discounts and savings. If you choose a provider outside of the network, you

will get reduced benefits. PPOs give you more flexibility to see whichever doctor you want, whenever

you want, often without a referral. You will pay less when you choose an in-network provider, but you

can choose to see an out-of-network provider, although you will pay more than if you received your

care in-network.

- Point of Service Plan (POS)

With a POS, members can choose their physician that has previously agreed to provide services at a

discounted fee. In a POS, the member uses the chosen physician as a gateway before moving on to a

specialist. Whenever the employee has a medical issue, the POS physician must be contacted first

to obtain the most benefit from the health insurance plan.

- Exclusive Provider Organization (EPO)

With an EPO, you may have a moderate amount of freedom to choose your health care providers -

more than an HMO; you do not have to get a referral from a primary care doctor to see a

specialist. There is no coverage for out-of-network providers; if you see a provider that is

not in your plan's network – other than in an emergency – you will have to pay the full cost

yourself. EPO has a lower premium than a PPO offered by the same insurer.

- High Deductible Health Plan (HDHP)

An HDHP is generally similar to a PPO. However, these plans have a high deductible that you must

meet before your plan will begin to pay a portion of your costs. A high deductible plan typically

has a lower monthly premium but higher out of pocket costs.

3. Liability Insurance

Such insurance is availed to insure properties, cars, businesses, etc. On buying a liability insurance -

like car insurance, home insurance, business insurance, in case of any damage to the insured object or

property during the policy tenure, the insurance company will financially compensate the owner of the

policyholder.

'''

ad5 = '''Insurers will evaluate whether they can issue a policy based on certain criteria such as:

- Age

- Medical history

- Previous claims made

- Amount of coverage you are requesting

Some types of insurance, such as life insurance, would require a medical exam. After which, the insurer

would review your application and access your personal and medical history to assess your risk. After

this assessment, you will know the amount of coverage you are qualified for and the premiums you need

to pay. No matter what type of insurance you are applying for, answer all questions on the application

fully and honestly. If you withhold important information or if you lie on the application, it can be

the basis for cancelling your policy, or worse, refusing your claim in the future.'''

ad6 = '''These conditions apply to most insurance policies:

- Insurance does not cover deliberate damage caused by the insured person.

- Insurance covers only the time period stated in the contract.

- You may not be covered if you do not mitigate the damage—that is, take reasonable steps to prevent

the damage from becoming worse. For example, if your home is being damaged from water leaking

from the roof, and you do not cover the hole, your coverage may be limited.

- Compensation is usually based on the actual cash value of the property, when the damage occurred.

For example, if your ten-year-old car is stolen, you will be compensated for the value of the used

car, not for the value of a new one.

- You must inform the insurer of any factors that might affect the risks that the policy covers.

The insurer could refuse to compensate you if you misrepresent the risks.

- Insurance may become void if you do not pay your premiums on time.

(canada.ca)'''

ad7 = '''Here are a few tips while buying insurance:

1. Purchase life insurance.

Life insurance is essential, no matter how young or old you are. Buying now may be a smart move because

it's cheaper to buy a life insurance policy when you're young and healthy. This kind of insurance can

help your family cover unexpected costs in your absence, including student loan debt or a mortgage,

in addition to end-of-life costs.'

2. When looking for insurance, your top priority should be to find adequate coverage. Price is

important, but you'll want to determine what kind of coverage you need first. You may be tempted

to choose insurance with the lowest price tag, but if you don't have enough coverage

(or the right kind of coverage), you will see less financial benefit when it comes time to file a

claim.

3. An independent insurance agent is an essential resource when purchasing insurance — especially

```
    if this is your first time. working with an independent agent
can help make sure that you are getting

    the best coverage, for the best price. You'll also benefit from
independent agents' insurance knowledge;

    they know how to talk you through your options and actually
explain what each policy includes.

    An independent agent will make sure all of your assets are
covered, help you find discounts or other ways

    to save, and be a valuable resource as your life changes and
your insurance needs change, too.


    I hope I have helped you! If you are not satisfied, please meet
a financial advisor in our nearest

    Auxilium branch who can help you more. If you need me, I'm
here!'''


            if b1['text'] == 'Hi! Yes, I want to know more about
insurance':

                T.delete(1.0, t.END)

                T.insert(t.END, ad1)

                b1['text'] = 'What are some terms I need to know
while opting for insurance?'


            elif b1['text'] == 'What are some terms I need to know
while opting for insurance?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad2)

                b1['text'] = "That's a lot! How does insurance
work?"


            elif b1['text'] == "That's a lot! How does insurance
work?":

                T.delete(1.0, t.END)

                T.insert(t.END, ad3)

                b1['text'] = 'What are the important types of
insurance I should know about?'
```

```
            elif b1['text'] == 'What are the important types of
insurance I should know about?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'Okay... How do I apply for insurance?'


            elif b1['text'] == 'Okay... How do I apply for
insurance?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'What conditions apply to insurance
policies?'


            elif b1['text'] == 'What conditions apply to insurance
policies?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'Anything else?'


            elif b1['text'] == 'Anything else?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, '''No problem. See you soon!


    Credits: Grange Insurance, ETMONEY, World Trips, canada.ca (govt
website), Live&Learn (livelearn.ca),
    Investopedia, WebMD''')
                b1['text'] = ''
```

```
        def backclick():
            ad1 = '''You are in the right place! What is insurance?
    According to Investopedia, insurance is a contract, represented by a policy, in which an individual
    or entity receives financial protection or reimbursement against losses from an insurance company.
    There are many types of insurance policies. There are many types of insurance such as life insurance,
    motor insurance, Health insurance, travel insurance, property insurance etc.
    Why do we need insurance?
    1. Insurance ensures family's financial stability
       No matter how much you have managed to save or what your monthly income is, an unexpected event
       can burn a huge hole in your pocket or can simply jeopardize your family's financial future. For
       example, if you do not have adequate life insurance, your family might have to go through
       financial hardship if you were to meet with an untimely death. Though no amount of money can
       replace the loss of loved ones, having life insurance would save them from going through financial
       hardship.
    2. Insurance reduces stress during difficult times.
       The premium you pay to the insurance company is the price that guarantees that the insurance
       company will cover the damage in case of an unforeseen event. And, that guarantee that your risk
       is covered brings peace of mind. '''

            ad2 = '''Definition of terms:
    Policy:
    A legal contract between you and the insurer. It details what risks are covered, under what
    circumstances the insurer will make a payment to you, how much money and what type of benefit
    you will receive if you make a claim.
```

Policyholder:

The insured or the person covered under the policy.

Coverage:

The amount of protection you have bought. It is also the maximum amount the insurance company

will pay you if you make a claim for loss or event covered by your policy.

Benefit:

The amount the insurer will pay you if the insurer accepts your claim.

Premium:

The amount you pay for the insurance.

Cash value:

This is the amount the insurer pays to the policyholder when a life insurance policy is

cancelled. It can also be an amount added to the death benefit and can be paid upon the

insured's death. This term is used with permanent life insurance policies.

Death benefit:

The amount the insurer will pay the beneficiary or beneficiaries upon the insured's death.

Claim:

This is the official notice to your insurer to be paid for a loss or event covered by your insurance policy.

Beneficiary:

This is the person or entity the insured names or assigns to receive the proceeds of the

policy. A beneficiary can be revocable (can be changed at any time without informing the

beneficiary) or irrevocable (can't be changed without the beneficiary's written permission).

Deductible:

The amount you agree to pay before the insurer pays the rest.

Exclusions:

Things that are not covered by your policy. For example, some health insurance policies may

exclude certain medical conditions you had before you applied for insurance or a travel

insurance policy may exclude claims if you travel to a high-risk country. This is why it

is important to read your policy thoroughly to check what it covers and what it doesn't

cover so that there will be no surprises when the time to claim comes.

Risk:

The probability or likelihood that an insured event, such as loss, injury or death, will

happen while the policy is in effect.

Rider:

It is a clause or term added to your insurance policy to provide protection. This has an additional

cost because it covers risks not covered in the basic policy.

Term:

The time period you are covered by your policy.

Grace period:

10 days immediately following the day you purchase an insurance policy, during which you may

cancel the policy for a full refund of any premiums paid'''

ad3 = '''When people buy insurance, they put their money into a pool with many others. Some of that

pool of money helps the policyholders who suffer a hardship in that period. The hardship can be related

to home, motor or business losses. You are covered only for losses written in your contract, and not for

predictable events. When a hardship occurs, a claim is made. This is an official request for the insurer to

pay for a covered loss. The insured's agent can assist in claiming benefits. Supporting documents/

evidence will be required, depending on the type of loss (for example, pictures of an injury or property

damage for an accident or property insurance claim, or a death certificate for a life insurance claim)

during investigation of claims.'''

ad4 = '''There are 3 important insurances:

1. Life Insurance:

Life insurance financially protects your family in case you die an early death. You pay a regular

premium to the insurance company for a specific number of years. In return the insurance

company pays a sum assured to your family if you die during the policy tenure.

There are two types of life insurance:

> Term – provides coverage for a specific amount of time. If the insured dies within the period

of coverage(and the premiums are paid), the beneficiary receives the death benefit as

stated in the policy. The coverage ends at the specified term. It can be renewed after

the term, however, the premium may increase since premiums may depend on the insured's

age.

> Permanent – it covers the insured throughout their lifetime (unless the insured fails to pay

the premiums).

There are two kinds:

o Whole life insurance – this guarantees that your premiums will not change as you get older. This type of insurance has a guaranteed minimum cash value and death benefit amount.

o Universal life insurance – this is a product combining life insurance and investment.

2.Health Insurance:

Health insurance can help you pay for services that the provincial health care plan does not cover.

Some types can supplement your income if you suffer a major illness or injury. Other types can pay for medical expenses if you become ill while on vacation.

Health insurance can be tricky to navigate. Managed care insurance plans require policyholders to receive care from a network of designated healthcare providers for the highest level of coverage.

There are 4 main types of insurance plans:

- Health Maintenance Organization (HMO)

A Health Maintenance Organization, or HMO, provides employers or groups a way to take care of all their employees' or members' health care needs with reduced costs by negotiating with specific doctors, hospitals, and clinics. Your doctors are employees of the same company that offers your insurance. You have fewer options when choosing a doctor, and are generally required to receive all care, outside of emergency situations, through the HMO in order to receive insurance coverage. The employee must use these specific providers for the reduced fees to be provided to their medical insurance plan. In an HMO plan, you have the least flexibility but will likely have

the easiest claims experiences since the network takes care of putting in the claims for you.

- Preferred Provider Organization (PPO)

A PPO, or preferred provider organization, is a health plan where you have the ability to choose your own healthcare providers. If you choose a doctor in the insurer's preferred provider network, you will get additional discounts and savings. If you choose a provider outside of the network, you will get reduced benefits. PPOs give you more flexibility to see whichever doctor you want, whenever you want, often without a referral. You will pay less when you choose an in-network provider, but you can choose to see an out-of-network provider, although you will pay more than if you received your care in-network.

- Point of Service Plan (POS)

With a POS, members can choose their physician that has previously agreed to provide services at a discounted fee. In a POS, the member uses the chosen physician as a gateway before moving on to a specialist. Whenever the employee has a medical issue, the POS physician must be contacted first to obtain the most benefit from the health insurance plan.

- Exclusive Provider Organization (EPO)

With an EPO, you may have a moderate amount of freedom to choose your health care providers - more than an HMO; you do not have to get a referral from a primary care doctor to see a specialist. There is no coverage for out-of-network providers; if you see a provider that is not in your plan's network – other than in an emergency – you will have to pay the full cost yourself. EPO has a lower premium than a PPO offered by the same insurer.

- High Deductible Health Plan (HDHP)

An HDHP is generally similar to a PPO. However, these plans have a high deductible that you must

meet before your plan will begin to pay a portion of your costs. A high deductible plan typically

has a lower monthly premium but higher out of pocket costs.

3. Liability Insurance

Such insurance is availed to insure properties, cars, businesses, etc. On buying a liability insurance -

like car insurance, home insurance, business insurance, in case of any damage to the insured object or

property during the policy tenure, the insurance company will financially compensate the owner of the

policyholder.

'''

ad5 = '''Insurers will evaluate whether they can issue a policy based on certain criteria such as:

- Age

- Medical history

- Previous claims made

- Amount of coverage you are requesting

Some types of insurance, such as life insurance, would require a medical exam. After which, the insurer

would review your application and access your personal and medical history to assess your risk. After

this assessment, you will know the amount of coverage you are qualified for and the premiums you need

to pay. No matter what type of insurance you are applying for, answer all questions on the application

fully and honestly. If you withhold important information or if you lie on the application, it can be

the basis for cancelling your policy, or worse, refusing your claim in the future.'''

ad6 = '''These conditions apply to most insurance policies:

- Insurance does not cover deliberate damage caused by the insured person.

- Insurance covers only the time period stated in the contract.

- You may not be covered if you do not mitigate the damage—that is, take reasonable steps to prevent

the damage from becoming worse. For example, if your home is being damaged from water leaking

from the roof, and you do not cover the hole, your coverage may be limited.

- Compensation is usually based on the actual cash value of the property, when the damage occurred.

For example, if your ten-year-old car is stolen, you will be compensated for the value of the used

car, not for the value of a new one.

- You must inform the insurer of any factors that might affect the risks that the policy covers.

The insurer could refuse to compensate you if you misrepresent the risks.

- Insurance may become void if you do not pay your premiums on time.

(canada.ca)'''

ad7 = '''Here are a few tips while buying insurance:

1. Purchase life insurance.

Life insurance is essential, no matter how young or old you are. Buying now may be a smart move because

it's cheaper to buy a life insurance policy when you're young and healthy. This kind of insurance can

help your family cover unexpected costs in your absence, including student loan debt or a mortgage,

in addition to end-of-life costs.'

2. When looking for insurance, your top priority should be to find adequate coverage. Price is

important, but you'll want to determine what kind of coverage you need first. You may be tempted

```
    to choose insurance with the lowest price tag, but if you don't
have enough coverage

    (or the right kind of coverage), you will see less financial
benefit when it comes time to file a

    claim.


    3. An independent insurance agent is an essential resource when
purchasing insurance — especially

    if this is your first time. working with an independent agent
can help make sure that you are getting

    the best coverage, for the best price. You'll also benefit from
independent agents' insurance knowledge;

    they know how to talk you through your options and actually
explain what each policy includes.

    An independent agent will make sure all of your assets are
covered, help you find discounts or other ways

    to save, and be a valuable resource as your life changes and
your insurance needs change, too.


    I hope I have helped you! If you are not satisfied, please meet
a financial advisor in our nearest

    Auxilium branch who can help you more. If you need me, I'm
here!'''


            if b1['text'] == 'Hi! Yes, I want to know more about
insurance':

                new_window.destroy()


            elif b1['text'] == 'What are some terms I need to know
while opting for insurance?':

                T.delete(1.0, t.END)

                T.insert(t.END, 'Hey there! You needed help?')

                b1['text'] = 'Hi! Yes, I want to know more about
insurance'


            elif b1['text'] == "That's a lot! How does insurance
work?":

                T.delete(1.0, t.END)
```

```
                T.insert(t.END, ad1)
                b1['text'] = 'What are some terms I need to know
while opting for insurance?'


            elif b1['text'] == 'What are the important types of
insurance I should know about?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = "That's a lot! How does insurance
work?"


            elif b1['text'] == 'Okay... How do I apply for
insurance?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'What are the important types of
insurance I should know about?'


            elif b1['text'] == 'What conditions apply to insurance
policies?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = 'Okay... How do I apply for insurance?'


            elif b1['text'] == 'Anything else?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'What conditions apply to insurance
policies?'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad6)
                b1['text'] = 'Anything else?'
```

```
            elif b1['text'] == '':
                T.delete(1.0, t.END)
                T.insert(t.END, ad7)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
        new_window.resizable(width = False, height = False)
        new_window.tk.call('wm', 'iconphoto', new_window._w,
tk.PhotoImage(file= r"C:\Users\Vijay\Desktop\Sahaana\School\Comp
Sci\GR12 - Project\Auxilium_icon.png"))


        l = t.Label(new_window, text = 'Insurance', bg =
'lightblue')
        l.grid(row = 0, column = 0)
        l.config(font = ('Comic Sans', 14))
        T = st.ScrolledText(new_window)
        T.grid(row = 1, column = 0)
        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'
        b1 = t.Button(new_window, text = 'Hi! Yes, I want to know
more about insurance', command = lambda: bclick(), font = ('Comic
Sans', 12))
        b1.grid(row = 3, column = 0, sticky = 'nesw')
        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))
        b3.grid(row = 4, column = 0, sticky = 'nesw')
        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))
        b2.grid(row = 5, column = 0, sticky = 'nesw')
        T.insert(-1.-1, ad)
        new_window.mainloop()
```

```
    def investment():
        from tkinter import scrolledtext as st
        def bclick():
            ad1 = '''Becoming an investor can seem like a far-off
goal for anyone if they start their
    career. However, they are not required to be rich or even
financially well-off to start investing.
    Anyone can do it. So, are you ready to start this journey?'''

            ad2 = '''Great! An investment is an asset or item that
is purchased with the hope that it
    will generate income or appreciate in value at some point in the
future. Investment may generate
    income for you in two ways. One, if you invest in a saleable
asset, you may earn income by way of
    profit. Second, if Investment is made in a return generating
plan, then you will earn an income
    via accumulation of gains. There are various types of
investments, such as Stocks, Bonds,
    investment funds, bank products, annuities, retirement, saving
for education,
    alternative and complex Products, initial coin offerings and
cryptocurrencies, and insurance'''

            ad3 = '''That is a good question! Investing is putting
money to work in order to grow it.
    When you invest in stocks or bonds, you are putting that capital
to work under the supervision of
    a firm and its management team. Although there is some risk,
that risk is rewarded with a positive
    expected return in the form of capital gains and/or dividend &
interest flows. Cash, on the other
    hand, will not grow, and may very well lose buying power over
time due to inflation. Without
    investment, companies would not be able to raise the capital
needed to increase the economy.'''
```

ad4 = '''Most ordinary individuals can easily make investments in stocks, bonds, and CDs.

With stocks, you are investing in the equity of a company, which means you invest in some residual

claim to a company's future profit flows and often gain voting rights (based on the number of shares

owned) to give your voice to the direction of the company. Bonds and CDs are debt investments, where

the borrower puts that money to use in a pursuit that is expected to bring in cash flows greater than

the interest owed to the investors. When you are young, first understand fully about what investment

is and its role, and then start. At an early age, you have few responsibilities and thus, have a better

tendency to experiment with different investment investments and leverage those, which suit your

requirements best. Investing early is also better because of the compounding benefits on investments

that help grow your money. With more years ahead, you can reap maximum benefits on your investments,

provided you first understand and evaluate different aspects of what investment is and then start early. '''

ad5 = '''Yes they are. Here is a risk ladder for investments which shows

major asset classes in ascending order of risk:

Cash

A cash bank deposit is the simplest, most easily understandable investment asset — and the safest.

Not only does it give investors precise knowledge of the interest they will earn, but it also

guarantees they will get their capital back. On the downside, the interest earned from cash put away

in a savings account rarely beats inflation. Certificates of deposit (CDs) are highly liquid instruments,

very similar to cash which are instruments that typically provide higher interest rates than those in

savings accounts. However, the money is locked up for a while, and there are potential early withdrawal

penalties involved.

Bonds

A bond is a debt apparatus representing a loan made by an investor to a borrower. A typical bond will

involve either a corporation or a government agency where the borrower will issue a fixed interest rate

to the lender in exchange for using their capital. Bonds are common in organizations that use them

to finance operations, purchases, or other projects. Bond rates are essentially determined by interest

rates. Due to this, they are heavily traded during periods of quantitative easing or when the

Federal Reserve—or other central banks—raise interest rates.

Mutual Funds

A mutual fund is a type of investment where more than an investor pools their money to

purchase securities. Mutual funds are not necessarily passive as they are managed by portfolio

managers who designate and distribute the pooled investment into stocks, bonds, and other

securities. Individuals may invest in mutual funds for as little as $1,000 per share, letting

them expand into as many as 100 different stocks included within a given portfolio. Mutual funds

are sometimes designed to simulate underlying indexes such as the S&P 500 or DOW Industrial Index.

Many mutual funds are actively managed, meaning they are updated by portfolio managers who carefully

track and adjust their allocations within the fund. However, these funds generally have greater

costs—such as yearly management fees and front-end charges—which can cut into an investor's returns.

Mutual funds are valued at the end of the trading day and all buy and sell transactions are likewise

executed after the market closes.

Exchange-Traded Funds (ETFs)

Exchange-traded funds (ETFs) have become quite attractive since their introduction back in the

mid-1990s. ETFs are similar to mutual funds, but they trade throughout the day, on a stock exchange. In

this way, they mirror the buy-and-sell behaviour of stocks. This also means their value can change

drastically during a trading day. ETFs can track an underlying index such as the S&P 500 or any other

'basket' of stocks the issuer of the ETF wants to underline a specific ETF with. This can include anything

from emerging markets, commodities, individual business sectors such as biotechnology or agriculture, and

more. Due to the ease of trading and broad coverage, ETFs are extremely popular with investors.

Stocks

Shares of stock let investors participate in the company's success via raises in the stock's price and

through dividends. Shareholders have a claim on the company's assets in the event of liquidation

(that is, the company going bankrupt) but do not own the assets. Holders of common stock enjoy voting

rights at shareholders' meetings. Holders of preferred stock do not have voting rights, but they do

receive preference over common shareholders in terms of the dividend payments.

Alternative Investments

There is a vast world of alternative investments, including the following sectors:

Real estate: Investors can acquire real estate by directly buying commercial or residential properties.

Alternatively, they can purchase shares in real estate investment trusts (REITs). REITs act like

mutual funds wherein a group of investors pool their money together to purchase properties. They

trade like stocks on the same exchange.

Hedge funds and private equity funds:

Hedge funds, which may invest in a variety of assets designed to deliver beyond market returns,

called 'alpha.' However, performance is not guaranteed, and hedge funds can see unimaginable

shifts in returns, sometimes underperforming the market by a significant margin. Typically, only

available to certified investors, these vehicles (assets offered by the investment industry to

help investors move money from the present to the future, with the hope of increasing the value

of their money) often require high initial investments of $1 million or more. They also tend to

impose net worth requirements. Both investment types may tie up an investor's money for

substantial periods.

Commodities: Commodities refer to physical resources such as gold, silver, crude oil, as well as

agricultural products.

'''

ad6 = '''Some tips for investing are:

1. Set investment Goals

You have to decide what you want to get out of investing? Consider things like income, capital

appreciation, and safety of capital. Also, consider your age, your personal circumstances, and

your financial position.

2. Invest Early

I have covered this before.

3. Be cautious of commissions

Some professionals are well known for selling products that pay them big commissions, but don't

pay much to their buyers. They would talk you into buying investments that give them high

commissions. Do some research before considering the investment.

4. Diversify your investments

You should have a diversified portfolio to avoid losing a lot of money when stocks go down.

This way, you will have some stocks that are rising, even when others are falling.

5. Make investments automatic

Set aside a certain amount of money to be automatically invested each month. You can set up

automatic investment plans through various brokerage service firms and automated investment

services available. This way, you will avoid stalling and consistently invest.

I hope I have helped you! If you are not satisfied, please meet a financial advisor in our nearest

Auxilium branch who can help you more. If you need me, I'm here!

'''

```
            if b1['text'] == 'Hi! Yes, I want to know more about investments':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'

            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
```

```
                T.insert(t.END, ad2)

                b1['text'] = 'Why invest when you can save money
with zero risk?'


            elif b1['text'] == 'Why invest when you can save money
with zero risk?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad3)

                b1['text'] = 'What are some types of investments I
can make & when should I start investing?'


            elif b1['text'] == 'What are some types of investments I
can make & when should I start investing?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad4)

                b1['text'] = "Aren't investments subject to market
risk? Which investment has a high risk?"


            elif b1['text'] == "Aren't investments subject to market
risk? Which investment has a high risk?":

                T.delete(1.0, t.END)

                T.insert(t.END, ad5)

                b1['text'] = 'What are some tips for investing?'


            elif b1['text'] == 'What are some tips for investing?':

                T.delete(1.0, t.END)

                T.insert(t.END, ad6)

                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':

                T.delete(1.0, t.END)

                T.insert(t.END, '''No problem. See you soon!
```

```
    Credits: FINRA, Investopedia, Lifehack, MaxLife Insurance''')

                b1['text'] = ''


        def backclick():

            ad1 = '''Becoming an investor can seem like a far-
fetched goal for anyone if they start their

    career. However, they are not required to be rich or even
financially well-off to start investing.

    Anyone can do it. So, are you ready to start this journey?'''


            ad2 = '''Great! An investment is an asset or item that
is purchased with the hope that it

    will generate income or appreciate in value at some point in the
future. Investment may generate

    income for you in two ways. One, if you invest in a saleable
asset, you may earn income by way of

    profit. Second, if Investment is made in a return generating
plan, then you will earn an income

    via accumulation of gains. There are various types of
investments, such as Stocks, Bonds,

    investment funds, bank products, annuities, retirement, saving
for education,

    alternative and complex Products, initial coin offerings and
cryptocurrencies, and insurance'''


            ad3 = '''That is a good question! Investing is putting
money to work in order to grow it.

    When you invest in stocks or bonds, you are putting that capital
to work under the supervision of

    a firm and its management team. Although there is some risk,
that risk is rewarded with a positive

    expected return in the form of capital gains and/or dividend &
interest flows. Cash, on the other

    hand, will not grow, and may very well lose buying power over
time due to inflation. Without

    investment, companies would not be able to raise the capital
needed to increase the economy.'''
```

ad4 = '''Most ordinary individuals can easily make investments in stocks, bonds, and CDs.

With stocks, you are investing in the equity of a company, which means you invest in some residual

claim to a company's future profit flows and often gain voting rights (based on the number of shares

owned) to give your voice to the direction of the company. Bonds and CDs are debt investments, where

the borrower puts that money to use in a pursuit that is expected to bring in cash flows greater than

the interest owed to the investors. When you are young, first understand fully about what investment

is and its role, and then start. At an early age, you have few responsibilities and thus, have a better

tendency to experiment with different investment investments and leverage those, which suit your

requirements best. Investing early is also better because of the compounding benefits on investments

that help grow your money. With more years ahead, you can reap maximum benefits on your investments,

provided you first understand and evaluate different aspects of what investment is and then start early. '''

ad5 = '''Yes they are. Here is a risk ladder for investments which shows

major asset classes in ascending order of risk:

Cash

A cash bank deposit is the simplest, most easily understandable investment asset — and the safest.

Not only does it give investors precise knowledge of the interest they will earn, but it also

guarantees they will get their capital back. On the downside, the interest earned from cash put away

in a savings account rarely beats inflation. Certificates of deposit (CDs) are highly liquid instruments,

very similar to cash which are instruments that typically provide higher interest rates than those in

savings accounts. However, the money is locked up for a while, and there are potential early withdrawal

penalties involved.

Bonds

A bond is a debt apparatus representing a loan made by an investor to a borrower. A typical bond will

involve either a corporation or a government agency where the borrower will issue a fixed interest rate

to the lender in exchange for using their capital. Bonds are common in organizations that use them

to finance operations, purchases, or other projects. Bond rates are essentially determined by interest

rates. Due to this, they are heavily traded during periods of quantitative easing or when the

Federal Reserve—or other central banks—raise interest rates.

Mutual Funds

A mutual fund is a type of investment where more than an investor pools their money to

purchase securities. Mutual funds are not necessarily passive as they are managed by portfolio

managers who designate and distribute the pooled investment into stocks, bonds, and other

securities. Individuals may invest in mutual funds for as little as $1,000 per share, letting

them expand into as many as 100 different stocks included within a given portfolio. Mutual funds

are sometimes designed to simulate underlying indexes such as the S&P 500 or DOW Industrial Index.

Many mutual funds are actively managed, meaning they are updated by portfolio managers who carefully

track and adjust their allocations within the fund. However, these funds generally have greater

costs—such as yearly management fees and front-end charges—which can cut into an investor's returns.

Mutual funds are valued at the end of the trading day and all buy and sell transactions are likewise

executed after the market closes.

Exchange-Traded Funds (ETFs)

Exchange-traded funds (ETFs) have become quite attractive since their introduction back in the

mid-1990s. ETFs are similar to mutual funds, but they trade throughout the day, on a stock exchange. In

this way, they mirror the buy-and-sell behaviour of stocks. This also means their value can change

drastically during a trading day. ETFs can track an underlying index such as the S&P 500 or any other

'basket' of stocks the issuer of the ETF wants to underline a specific ETF with. This can include anything

from emerging markets, commodities, individual business sectors such as biotechnology or agriculture, and

more. Due to the ease of trading and broad coverage, ETFs are extremely popular with investors.

Stocks

Shares of stock let investors participate in the company's success via raises in the stock's price and

through dividends. Shareholders have a claim on the company's assets in the event of liquidation

(that is, the company going bankrupt) but do not own the assets. Holders of common stock enjoy voting

rights at shareholders' meetings. Holders of preferred stock do not have voting rights, but they do

receive preference over common shareholders in terms of the dividend payments.

Alternative Investments

There is a vast world of alternative investments, including the following sectors:

Real estate: Investors can acquire real estate by directly buying commercial or residential properties.

Alternatively, they can purchase shares in real estate investment trusts (REITs). REITs act like

mutual funds wherein a group of investors pool their money together to purchase properties. They

trade like stocks on the same exchange.

Hedge funds and private equity funds:

Hedge funds, which may invest in a variety of assets designed to deliver beyond market returns,

called 'alpha.' However, performance is not guaranteed, and hedge funds can see unimaginable

shifts in returns, sometimes underperforming the market by a significant margin. Typically, only

available to certified investors, these vehicles (assets offered by the investment industry to

help investors move money from the present to the future, with the hope of increasing the value

of their money) often require high initial investments of $1 million or more. They also tend to

impose net worth requirements. Both investment types may tie up an investor's money for

substantial periods.

Commodities: Commodities refer to physical resources such as gold, silver, crude oil, as well as

agricultural products.

'''

ad6 = '''Some tips for investing are:

1. Set investment Goals

You have to decide what you want to get out of investing? Consider things like income, capital

appreciation, and safety of capital. Also, consider your age, your personal circumstances, and

your financial position.

2. Invest Early

I have covered this before.

3. Be cautious of commissions

Some professionals are well known for selling products that pay them big commissions, but don't

pay much to their buyers. They would talk you into buying investments that give them high

commissions. Do some research before considering the investment.

4. Diversify your investments

You should have a diversified portfolio to avoid losing a lot of money when stocks go down.

This way, you will have some stocks that are rising, even when others are falling.

5. Make investments automatic

Set aside a certain amount of money to be automatically invested each month. You can set up

automatic investment plans through various brokerage service firms and automated investment

services available. This way, you will avoid stalling and consistently invest.

I hope I have helped you! If you are not satisfied, please meet a financial advisor in our nearest

Auxilium branch who can help you more. If you need me, I'm here!'''

```
            if b1['text'] == 'Hi! Yes, I want to know more about
insurance':
                new_window.destroy()


            elif b1['text'] == 'Sure!':
                T.delete(1.0, t.END)
                T.insert(t.END, 'Hey there! You needed help?')
                b1['text'] = 'Hi! Yes, I want to know more about
insurance'
```

```
            elif b1['text'] == 'Why invest when you can save money
with zero risk?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad1)
                b1['text'] = 'Sure!'


            elif b1['text'] == 'What are some types of investments I
can make & when should I start investing?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad2)
                b1['text'] = 'Why invest when you can save money
with zero risk?'


            elif b1['text'] == "Aren't investments subject to market
risk? Which investment has a high risk?":
                T.delete(1.0, t.END)
                T.insert(t.END, ad3)
                b1['text'] = 'What are some types of investments I
can make & when should I start investing?'


            elif b1['text'] == 'What are some tips for investing?':
                T.delete(1.0, t.END)
                T.insert(t.END, ad4)
                b1['text'] = "Aren't investments subject to market
risk? Which investment has a high risk?"


            elif b1['text'] == 'Thank you! That was helpful. I will
click the exit button now.':
                T.delete(1.0, t.END)
                T.insert(t.END, ad5)
                b1['text'] = 'What are some tips for investing?'


            elif b1['text'] == '':
                T.delete(1.0, t.END)
```

```
                T.insert(t.END, ad6)
                b1['text'] = 'Thank you! That was helpful. I will
click the exit button now.'


        new_window = t.Toplevel()
        new_window.configure(bg = 'lightblue')
        new_window.resizable(width = False, height = False)


        l = t.Label(new_window, text = 'Investments', bg =
'lightblue')
        l.grid(row = 0, column = 0)
        l.config(font = ('Comic Sans', 14))
        T = st.ScrolledText(new_window)
        T.grid(row = 1, column = 0)
        T.config(font = ('Comic Sans', 12))


        ad = 'Hey there! You needed help?'
        b1 = t.Button(new_window, text = 'Hi! Yes, I want to know
more about investments', command = lambda: bclick(), font = ('Comic
Sans', 12))
        b1.grid(row = 3, column = 0, sticky = 'nesw')
        b3 = t.Button(new_window, text = 'Back', command = lambda:
backclick(), font = ('Comic Sans', 12))
        b3.grid(row = 4, column = 0, sticky = 'nesw')
        b2 = t.Button(new_window, text = 'Exit', command = lambda:
ex(new_window), font = ('Comic Sans', 12))
        b2.grid(row = 5, column = 0, sticky = 'nesw')
        T.insert(-1.-1, ad)
        new_window.mainloop()


    img = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Financial Advice.png"
    w, h = 1200, 1080


    root = t.Toplevel()
```

```
    root.geometry('900x500')

    root.maxsize(900, 500)

    root.title('Auxilium Interactive Financial Advice User Manuals')

    root.tk.call('wm', 'iconphoto', root._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

    root.resizable(width = False, height = False)


    canvas = t.Canvas(root, width = w, height = h)

    canvas.grid(row = 0, column = 0)


    img = ImageTk.PhotoImage(Image.open(img).resize((900, 500),
Image.ANTIALIAS))

    canvas.background = img

    bg = canvas.create_image(0, 0, anchor = t.NW, image = img)


    frame = t.LabelFrame(root, padx = 25, pady = 25)

    frame.grid(row = 0, column = 0, padx = 10, pady = 10)


    mylabel = t.Label(frame, text = 'Hey there! Need financial
advice?\n', font = ('Arial', 15)).grid(row = 0, column = 0)


    l = (
        ['Retirement', 'Retirement'],

        ['Funding College', 'Funding College'],

        ['Debts', 'Debts'],

        ['First Job', 'First Job'],

        ['Insurance', 'Insurance'],

        ['Investments', 'Investments'],

        )


    var = t.StringVar()

    var.set('Retirement')

    i = 0
```

```
    for text, mode in l:
        r = t.Radiobutton(frame, text = text, variable = var, value
= mode, command = lambda: clicked(var.get()), font = ('Arial', 10))
        r.grid(row = 1 + i, column = 0)
        i += 1


    window = canvas.create_window(10, 10, anchor = t.NW, window =
frame)


    def clicked(value):
       if value == 'Retirement':
          root.wm_state('iconic')
          retirement()


       elif value == 'Funding College':
          root.wm_state('iconic')
          college()


       elif value == 'Debts':
          root.wm_state('iconic')
          debt()


       elif value == 'First Job':
          root.wm_state('iconic')
          job()


       elif value == 'Insurance':
          root.wm_state('iconic')
          insurance()


       elif value == 'Investments':
          root.wm_state('iconic')
          investment()
```

```
    mylabel = t.Label(frame, text = '').grid(row = 8, column = 0)

    exit_button = t.Button(frame, text= 'Exit', command =
root.destroy).grid(row = 9, column = 0)

    root.mainloop()


def main_page1():


    root1 = Toplevel()

    root1.geometry("800x500")

    root1.title('Auxilium Main Page')

    bg = PhotoImage(file =
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Greenoptions.png")

    root1.tk.call('wm', 'iconphoto', root1._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

    bg_label = Label(root1, image = bg)

    bg_label.place(x = 0, y = 0)

    root1.bg = bg


    def minimise():

      root1.wm_state('iconic')


    def open_img():

        x = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\MONEY2.jpg"

        img = Image.open(x)

        img = img.resize((300, 280), Image.ANTIALIAS)

        img = ImageTk.PhotoImage(img)

        panel = Label(root1, image = img).place(x = 480, y = 150)

        root1.img = img

    open_img()


    label_0 = tk.Label(root1, text = "WELCOME TO AUXILIUM",
```

```
                      width = 25, fg = 'black', font = ("bold",
24))
    label_0.place(x = 180, y = 50)
    label_1 = Label(root1, text="PLEASE SELECT AN OPTION: ",
                    width = 25, fg = 'red', font = ('normal', 16))
    label_1.place(x = 60, y = 130)


    var1 = IntVar()


    Checkbutton(root1, text = "Tracking expenses and budgeting ",
variable = var1,
                 command = lambda: [option_1(), minimise()], fg =
'blue', font = ("normal", 14), onvalue = "Yes", offvalue =
"No").place(x=150, y=180)


    var2 = IntVar()
    Checkbutton(root1, text = "Banking", variable = var2, command =
lambda: [banking(), minimise()], fg = 'blue', font = ("normal",
14)).place(x = 150, y = 220)


    var3 = IntVar()
    Checkbutton(root1, text = "Choosing a perfect loan scheme",
command = lambda: [loan_details(), minimise()], variable = var3, fg
= 'blue', font = ("normal", 14)).place(x = 150, y = 260)


    var4 = IntVar()
    Checkbutton(root1, text = "Virtual financial advisor", variable
= var4, fg='blue', font=("normal", 14), command = lambda: [vfa(),
minimise()]).place(x = 150, y = 300)


    var5 = IntVar()
    Checkbutton(root1, text="Checking net worth", variable = var5,
command = lambda: [net_worth(), minimise()], fg = 'blue', font =
("normal", 14)).place(x = 150, y = 340)


    var6 = IntVar()
```

```
    Checkbutton(root1, text = "Checking expenses bar graph",
variable = var6, command = lambda: [expense_graph(), minimise()], fg
= 'blue', font = ("normal", 14)).place(x = 150, y = 380)


    Button(root1, text = 'BACK', width = 30, bg = "RED", fg =
'white',
           font=("normal", 12)).place(x = 100, y = 440)


    root1.mainloop()


#LOGIN PAGE


def submit():


    try:
        userd = Username.get()
        passwd2 = password.get()
        global anc
        global PUser


        db = mysql.connector.connect(host="localhost", user="root",
password="Root", database="Auxilium")
        cursor = db.cursor()
        savequery = "select * from login"
        cursor.execute(savequery)
        print("ANC at login 1", anc)
        myresult = (cursor.fetchall())
        check_login = "SELECT * FROM login WHERE UserID = '" + userd
+ "' AND PSWD = '" + passwd2 + "'"
        cursor.execute(check_login, userd)
        myresult1 = (cursor.fetchall())
        print(myresult1[0][0])
        uk = myresult1[0][0]  # user key
        PUser = userd
```

```
        anc = uk


        #config.x = uk
        #print("ANC after setting global post submit", anc)
        rowcount = cursor.rowcount
        #print(rowcount)


        if cursor.rowcount == 1:
            rootL.wm_state('iconic')
            tkinter.messagebox.showinfo('INFORMATION', "Welcome
"+userd)
            main_page1()


        else:
            tkinter.messagebox.showerror(
                "ERROR", "Please enter a valid password or click
'Forgot password'.")
            print("Error occurred in submitting your login info")


    except:
        tkinter.messagebox.showerror(
            "ERROR", "Please enter a valid password or click 'Forgot
password'.")
        print("Error occurred in submitting your login info")



def new_user_account():
    # USER ACCOUNT CREATION


    usern = Username.get()
    passwd = password.get()
    savequery = "select * from login"
    '''try:'''
    db = mysql.connector.connect(
```

```
        host="localhost", user="root", password="Root",
database="Auxilium")
    cursor = db.cursor()


    cursor.execute(savequery)
    myresult = (cursor.fetchall())


    insert_query = "insert into LOGIN (UserID,PSWD) values (%s, %s)"
    val = (usern, passwd)
    cursor.execute(insert_query, val)
    db.commit()
    cursor.close()
    tkinter.messagebox.showinfo(
        "CONGRATULATIONS", "Your details have been inserted
successfully 👍 ")
    print("You have inserted ", {usern}, {passwd})


    '''except:
        tkinter.messagebox.showerror(
            "ERROR", "Could not create a new account.")
        print("Error occurred in user account creation")
    '''
def forgot_password():
    global New_password
    root1 = tk.Tk()
    root1.geometry("400x300")
    root1.title('Forgot Password')
    label_0 = tk.Label(root1, text="Enter a new password: ",
                       width=20, font=("bold", 15))
    label_0.place(x=35, y=60)
    label_1 = tk.Label(root1, text="New Password", width=20,
font=("bold", 10))
    label_1.place(x=80, y=130)
```

```
    New_password = tk.Entry(root1)
    New_password.place(x=240, y=130)
    Button(root1, text='Next', width=20, bg="blue", fg='white',
           command=submit_login).place(x=220, y=180)
    Button(root1, text='Back', width=20, bg="blue",
           fg='white').place(x=35, y=180)
    #Button(root1, text='Back' , width=20,bg="blue”, fg='white’,
command = root1.destroy()).place(x=35,y=180)
    root1.mainloop()



def submit_login():


    savequery = "select * from login"
    usern1 = Username.get()
    passwd1 = New_password.get()
    try:
        db = mysql.connector.connect(
            host="localhost", user="root", password="Root",
database="Auxilium")
        cursor = db.cursor()


        cursor.execute(savequery)
        myresult = (cursor.fetchall())


        update_query = "UPDATE LOGIN set PSWD = %s where UserID =
%s"
        values = (passwd1, usern1)
        cursor.execute(update_query, values)
        db.commit()
        cursor.close()
        tkinter.messagebox.showinfo(
            "CONGRATULATIONS", "Your details have been inserted
successfully 👍 ")
```

```
        print("You have inserted ", {usern1}, {passwd1})


    except:
        tkinter.messagebox.showerror("ERROR", "Could not update new
password.")
        print("Error occurred in user account creation")



def key():
    userd = Username.get()
    passwd2 = password.get()


    db = mysql.connector.connect(
        host="localhost", user="root", password="Root",
database="Auxilium")
    cursor = db.cursor()
    print("ANC at login 1")
    check_login = "SELECT * FROM login WHERE UserID = '" + \
        userd + "' AND PSWD = '" + passwd2 + "'"
    cursor.execute(check_login, userd)
    myresult1 = (cursor.fetchall())
    print('anan')
    print(myresult1[0][0])
    uk = myresult1[0][0]  # user key
    #print ("UK at login" +uk)
    anc = uk


    #config.x = uk
    print(anc)



img = r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\User login.png"
w, h = 1000, 800
```

```
rootL = tk.Tk()

rootL.geometry('800x500')

rootL.maxsize(800, 500)

rootL.title('Auxilium Login Page')

rootL.tk.call('wm', 'iconphoto', rootL._w, tk.PhotoImage(file=
r"C:\Users\Vijay\Desktop\Sahaana\School\Comp Sci\GR12 -
Project\Auxilium_icon.png"))

rootL.resizable(width = False, height = False)


canvas = tk.Canvas(rootL, width = w, height = h)

canvas.grid(row = 0, column = 0)


img = ImageTk.PhotoImage(Image.open(img).resize((800, 500),
Image.ANTIALIAS))

canvas.background = img

bg = canvas.create_image(0, 0, anchor = tk.NW, image = img)


frame = tk.LabelFrame(rootL, padx = 25, pady = 25)

frame.grid(row = 0, column = 0, padx = 10, pady = 10)


label_1 = tk.Label(rootL, text="USERNAME: ", width=20, font=("bold",
10))

label_1.place(x=360, y=160)


Username = tk.Entry(rootL)

Username.place(x=550, y=160)

Username.insert(0, "")


label_3 = tk.Label(rootL, text="PASSWORD: ", width=20, font=("bold",
10))

label_3.place(x=360, y=200)


password = tk.Entry(rootL, show = '•')
```

```
password.place(x=550, y=200)

password.insert(0, "")


var = IntVar()


global anc

anc = 777

Radiobutton(rootL, text="Create new account", width=15, font=(

    "bold", 11), variable=var, value=1,
command=new_user_account).place(x=330, y=250)

Radiobutton(rootL, text="Forgot password", width=15, font=("bold",
11),

            variable=var, value=2,
command=forgot_password).place(x=550, y=250)


Button(rootL, text='SUBMIT', width=30, bg="blue", fg='white',

       font=("bold", 12), command=submit).place(x=400, y=300)


rootL.mainloop()
```