

INDEX

| S. No. | PARTICULARS | Page No. |
|-------------------|---------------------------|---------------------|
| 1. | Topic of the Project | 1 |
| 2. | Background of the Project | 2 |
| 3. | Introduction | 3 |
| 4. | Functions and Modules | 4 |
| 5. | Complete Programming | 7 |
| 6. | Output and Working | 18 |
| 7. | Bibliography | 27 |

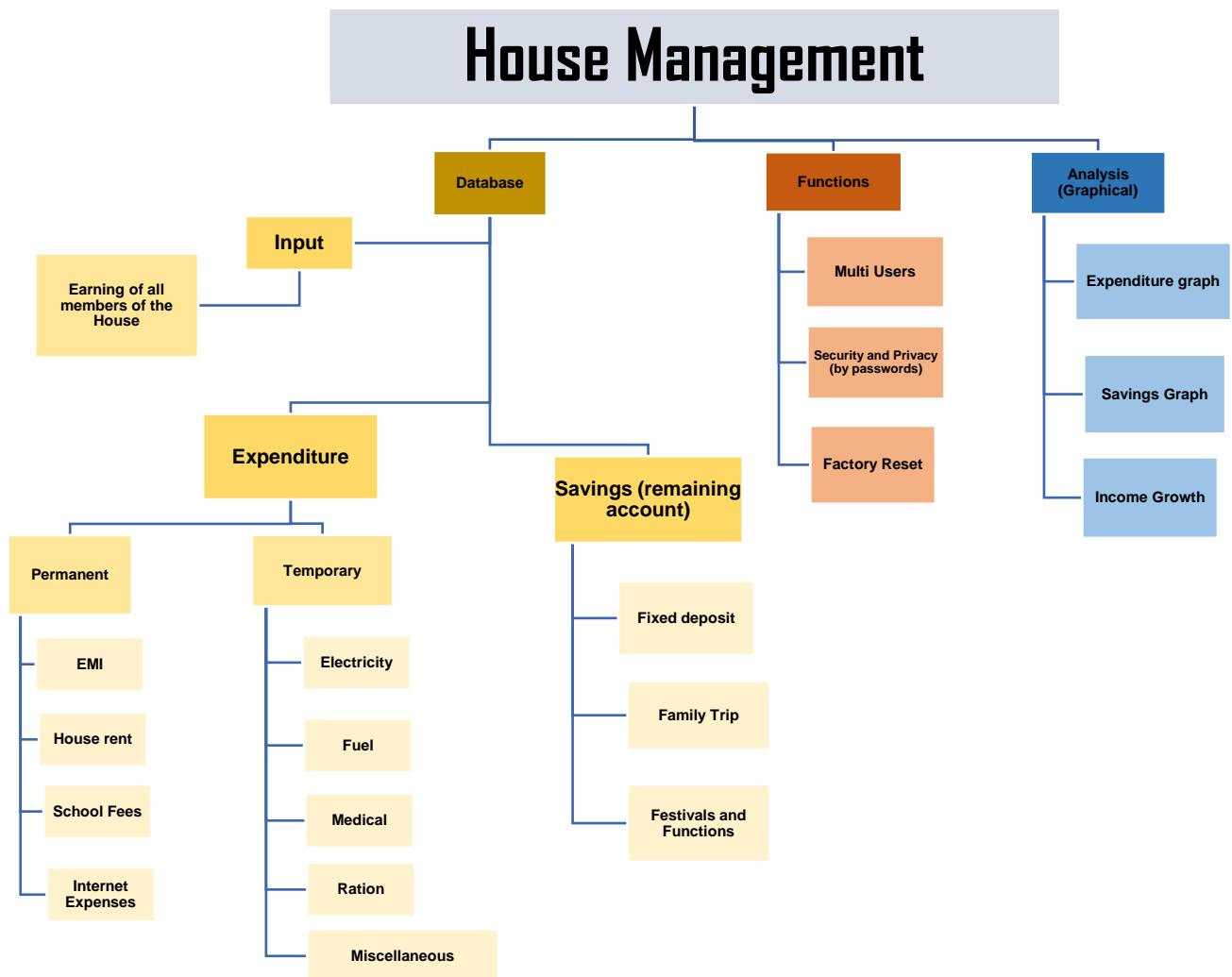
Topic Of Our Project



Group Members:

1. Daksh Agrawal
2. Sannidhya K. Sonber
3. Sai Krishnan Iyer

Background Of The Project



Introduction

Python

What is python?

Python programming language is one of the dynamic and object-oriented programming languages used for the development of diverse kinds of software developed by the Python Software foundation. Its significant advantage is that facilitates integration with other programming languages and software development tools. In addition, it has in-built standard libraries that are extensive. This means that it facilitates the development of a better source code.

An important feature of Python language that makes it suitable as a software development tool is that it has a dynamic type system and its memory management strategy is automatic. In addition, it can support the implementation of scripting applications. It is important to note that the development model of Python language is community based, implying that its reference implementation is free and bases on the open source platform. There are various interpreters for the language for various systems software, although programs developed by Python are executable in any environment irrespective of operating system environment.

Python language has extensive in-built tools that can provide support to Internet protocols, coupled with its code readability characteristic, Python is therefore one of the most appropriate programming languages that can be used in the development of dynamic web content using the concept of dynamic programming.

What is Python used for?

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it's relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

Functions and Modules Used

• Functions-

1. In-Built functions-

a. General-

- ✓ **time.sleep(0.4)**- The running program will halt for '0.4' seconds.
- ✓ **return 'object1'**- This will store the value of 'object1' in 'object2' and when 'object2' calls of a function have 'object1' .
- ✓ **len('object')**- This Function counts the length of the 'object'.
- ✓ **str('object')**- This Function converts object type to string type.
- ✓ **os.remove('file location')**- This Function removes the file from the file location.
- ✓ **dict.get('object')**- This Function uses key value from 'dict' dictionary where key name is 'object' .
- ✓ **quit()**-This Function leaves the program.
- ✓ **string.split()**-This Function converts sting to list format.
#Ex; 'We are the change' ~ ['We','are','the','change']

b. For Text Data File (.txt)

- ✓ **file = open('file location.txt','mode')**- This Function opens the file for back-work of the given mode.
- ✓ **file reading-when 'mode' = 'r','w+','a+','r+' i.e. reading mode**
- ✓ **file.read(n)**:reads first nth character in 'file'.
- ✓ **file.readline()**:reads a line of the 'file'.
- ✓ **file.readlines()**:reads the whole 'file' in single line.
- ✓ **file.write('object')**-writes the values of 'object' in 'file' mode' :- 'w','a' i.e. writing or appending mode.
- ✓ **file.close()**-opened file to be closed after work.

c. For library matplotlib (import matplotlib.pyplot as plt)

- ✓ **a=plt.xlabel('axis label')**-gives lable to axis.
- ✓ **b=plt.title('title')**-gives title to the graph.
- ✓ **c=plt.plot('x list','y list')**-plots graph by values of x and y.
- ✓ **d=plt.show()**-displays the graph plotted.

2. User Define Functions-

- i. **newuser():** checks and create new account
- ii. **password(username):** log-in for a user
- iii. **create_password(username):** creates new password for the 'username'
- iv. **change_password(username):** changes password for a user
- v. **monthname(username):** creates half of the location of the file to be manipulated according to the inputted month and year.
 - Location format : 'Database\\\'+'month'+year'+username'
 - Ex# location = 'Data Base\Jan2023userID'
- vi. **input_earning(username):** input earning of the house in a monthly format
- vii. **display_earning(username):** display earning of a user-defined function
- viii. **newpermaexp(username):** creates user-defined categories for permanent expenditure to be made
 - Example# House Rent, EMI, Fees
- ix. **permaexp(username):** add record for user-defined month|year
- x. **displaypermaexp(username):** display the record of permanent expenditure for user-defined month and year
- xi. **permaexpeditor(username):** edits record of permanent expenditure for the user-defined month
- xii. **newtempexp(username):** creates user defined categories for temporary expenditure to be made
- xiii. **addrec_tempexp(username):** add record for user-defined month|year
- xiv. **heading(username,type):** gives heading for table
- xv. **display_tempexp_month(username):** display the record of permanent expenditure for user-defined month of year
- xvi. **display_tempexp_all(username):** display the record of permanent expenditure for user-defined year
- xvii. **savings(username):** future planned expenditure made by user
- xviii. **display_savings(username):** display the record of savings
- xix. **create_records(username):** creates a current balance sheet of all the record of the user
- xx. **analysis(username):** tabular and graphical analysis of the records
- xxi. **graph(title,x,y) :** plots graph

#please install matplotlib library before running the codes

- xxii. **balance(username):** Initial account balance of the user
- xxiii. **factoryreset(username):** Deletes account and all records of the user
- xxiv. **Add_Rec(username):** Asks user to enter record for earnings, permanent expenditure, temporary expenditure or savings
- xxv. **Display_Rec(username):** Asks user to display record for earning, permanent expenditure, temporary expenditure or savings
- xxvi. **Done(username):** Quit from the program

- Modules used-

1. Import os:

The os module in python provides functions for creating and removing a directory(folder), fetching its content, changing and identifying the current directory, etc.

2. Import time:

As the name suggests Python time module allows to work with time in Python. It allows functionality like getting the current time, pausing the Program from executing, etc.

3. Import matplotlib:

This module is used to display the data in graphical form so that the data can be analyzed in systematic manner and variation can be studied.

(#this library has to be installed on the computer separately using command prompt. In command prompt type ' pip install matplotlib' to install)

Complete Programming

```
#system settings & creadentials

def newuser():
    time.sleep(1.5)
    print('\nGet started\n\tCreate new account . . ')
    time.sleep(1.5)
    username = str(input('Enter new username \t:\t'))
    time.sleep(1.5)
    try:
        f = open('Data Base\\'+username + ' credential.txt','r')
        r = f.readlines()
        time.sleep(1.5)
        print('This user name already exist ! ! ! ! \Try another')
        f.close()
    except IOError :
        create_password(username)
        newpermaexp(username)
        newtempexp(username)
        balance(username)
    return username

def create_password(username):
    #from getpass import getpass
    time.sleep(1.5)
    print('\tCreate Password\nMinimum length of password is 4')
    time.sleep(1.5)
    pwd1 = str(input('Enter New Password\t:\t'))
    time.sleep(1.5)
    x = len(pwd1)
    while x < 4 :
        print('Minimum length of password is 4')
        time.sleep(1.5)
        pwd1 = str(input('Enter New Password\t:\t'))
        time.sleep(1.5)
        x = len(pwd1)
    pwd2 = str(input('Conform Password \t:\t'))
    time.sleep(1.5)
    while pwd1!=pwd2:
        print('Password entered don\'t match')
        time.sleep(1.5)
        pwd1 = str(input('Enter New Password\t:\t'))
        time.sleep(1.5)
        x = len(pwd1)
    while x < 4 :
        print('Minimum length of password is 4')
        time.sleep(1.5)
        pwd1 = str(input('EnterPassword\t:\t'))
        time.sleep(1.5)
        x = len(pwd1)
    pwd2 = str(input('Conform Password \t:\t'))
    time.sleep(1.5)
    print('Password created sucessfully #\n\n')
    time.sleep(1.5)
    data = open('Data Base\\'+username + ' credential.txt','w')
    data.write(str(pwd1))
    data.close()
```



```

#-----
#functions

#earnings
def input_earning(username) :
    location = monthname(username)
    try:
        file = open(location+' Earning.txt' , 'r')
        rec = file.readline()
        time.sleep(1.5)
        print('\n\n', rec)
        time.sleep(1.5)
        print('File already Exists\n')
        time.sleep(1.5)
        choice = input('Enter (yes) to change the value else (no) to continue\t:\t')
        time.sleep(1.5)
        if choice == 'yes' or choice == 'Yes' or choice == 'y' :
            inp = int(input('Enter new amount of earning\t:\t'))
            time.sleep(1.5)
            file.close()
            file = open(location+' Earning.txt' , 'w')
            file.write('Earning'+'\t'+str(inp))
    except IOError:
        inp = int(input('Enter amount of earning\t:\t'))
        time.sleep(1.5)
        file = open(location+' Earning.txt' , 'w')
        file.write('Earning'+'\t'+str(inp))
    file.close()

def display_earning(username) :
    location = monthname(username)
    try:
        file = open(location+' Earning.txt' , 'r')
        x = file.readline()
        y = x.split()
        a = int(y[-1])
        time.sleep(1.5)
        print('Entered earning was\t:\t',a)
        time.sleep(1.5)
        file.close()
    except IOError:
        time.sleep(1.5)
        print('No record found')
        time.sleep(1.5)
        choice = input('Press (yes) to enter the record \t:\t')
        if choice == 'yes' or choice == 'Yes' or choice == 'y' :
            input_earning(username)

#permanent expanditure
def newpermaexp(username) :
    file = open('Data Base\Recent_Permanent '+username+'.txt','w+')
    time.sleep(1.5)
    print('Enter name and amount of permanent expanditure\n\tExample;\nHouse rent,EMI,School fees, gas cylinder etc')
    time.sleep(1.5)
    choice = 'yes'
    total = 0
    while choice == 'yes' or choice == 'Yes' or choice == 'y' :
        name = input('Enter name of expanditure\t:\t')
        value = input('Enter value\t:\t')
        time.sleep(1.5)
        file.write(name+'\t'+ value +'\r')
        total += int(value)
        choice = input('Enter (yes) to input more records else(no)\t:\t')
        time.sleep(1.5)
    file.write('Total'+'\t'+str(total))
    file.close()

```

```

def displaypermaexp(username):
    try :
        file = open('Data Base\Recent_Permanent '+username+'.txt','r')
        time.sleep(1.5)
        print('*'*100, '\nName\t\tExpandiseure\n', '*'*100)
        time.sleep(1.5)
        total = 0
        y = file.readline()
        x = y.split()
        while str(x[0]) != 'Total' :
            for i in range(0,len(x)-1) :
                print(x[i],end = ' ')
                time.sleep(.3)
            print('\t',x[-1],'\t')
            time.sleep(.3)
            total += int(x[-1])
            y = file.readline()
            x = y.split()
        time.sleep(1.5)
        print('*'*100, '\nTotal Expandiseure\t:\t', x[-1] , '\n', '*'*100)      #displays total
        time.sleep(1.5)
        file.close()
        return total
    except IOError:
        time.sleep(1.5)
        print('\nNo File Found\n\n')
        newpermaexp(username)

def permaexpeditor(username):
    import os
    pexp= 'Data Base\Recent_Permanent '+username+'.txt'
    file = open(pexp,'r+')
    tran = open('Data Base\Replace.txt', 'w')
    y = file.readline()
    x = y.split()
    total = 0
    while x[0] != 'Total' :
        print(x[0] ,'\t:\t' , x[-1])
        time.sleep(1.5)
        choice = input('Enter (yes) to continue else (no) to change value \t:\t')
        if choice == 'no' :
            x[-1] = input('Enter new value\t:\t')
            tran.write(x[0]+'\t'+ str(x[-1])+'\r')
            total += int(x[-1])
            y = file.readline()
            x = y.split()
    tran.write('Total'+ '\t'+str(total))
    file.close()
    tran.close()
    os.remove(pexp)
    os.rename('Data Base\Replace.txt',pexp)
    return total

def permaexp(username):
    location = monthname(username)
    file = open ( location + ' Perma_Exp.txt' , 'w' )
    total = displaypermaexp(username)
    time.sleep(1.5)
    choice = input('Enter (yes) to continue else (no) to change values \t:\t')
    if choice == 'no':
        total = permaexpeditor(username)
    file.write(str(total))

```

```

#temperary expanditure
def newtempexp(username):
    file = open ('Data Base\\'+username + ' Temp_Exp.txt' , 'w')      #To read no. of records
    time.sleep(1.5)
    print('\n\tTemporary Expanditure\nDescription ; \n\tYou can add categories of records\
of the expanditure which you make other then permanent expenditure whick are not fix\n\
For Example; Electricity , Fuel , Medicine , Ration and \
others categories.')
    time.sleep(1.5)
    choice = 'yes'
    while choice != 'no' and choice == 'yes' or choice == 'Yes' or choice == 'y' :
        time.sleep(1.5)
        cat = input ('Enter name of expanditure\t:\t')
        time.sleep(1.5)
        file.write(cat + '\r')
        choice = input('Press (yes) if you want to enter more catogaries else (no) : ')
    file.close()

# noinspection PyUnreachableCode
def addrec_tempexp(username) :
    try :
        f = open ('Data Base\\'+username + ' Temp_Exp.txt' , 'r')
        location = monthname(username)+ ' Temp_Exp'
        file = open ( location +'.txt','w')
        total = 0
        b = f.readline()
        a = b.split()
        if b == '':
            newtempexp(username)
            b = f.readline()
            a = b.split()
        while a :
            time.sleep(1.5)
            print( 'Enter ',a[0], ' expanditure\t:',end='\t')
            time.sleep(1.5)
            print( 'Enter ',a[0], ' expanditure\t:',end='\t')
            time.sleep(1.5)
            b = int(input())
            file.write(str(a[0])+'\t'+str(b) +'\r')
            total += b
            b = f.readline()
            a = b.split()
        time.sleep(1.5)
        print('Total Temperary expanditure\t:\t',total)
        time.sleep(1.5)
        file.write('Total'+ '\t' + str(total))
        return total
        f.close()
    except IOError:
        time.sleep(1.5)
        print('No category of expanditure found')
        newtempexp(username)

```

```

def heading(username,type):
    try :
        f = open ('Data Base\\'+username + type +'.txt' , 'r' )
        time.sleep(1.5)
        print('-*100,end = '\nMonth\t\t')
        rec = f.readline()
        rec = rec.split()
        while rec :
            print(rec[0],end='\t')
            rec = f.readline()
            rec = rec.split()
        print('Total\n',-*100)
        f.close()
    except IOError:
        time.sleep(1.5)
        print('No category of expenditure found')
        time.sleep(1.5)
        newtempexp(username)

def display_tempexp_month(username):
    time.sleep(1.5)
    print('Display record of month')
    time.sleep(1.5)
    location = monthname(username)
    try :
        file = open ( location + ' Temp_Exp.txt','r' )
        x =slice(10,-1,1)
        name = location[x]
        heading(username,' Temp_Exp')
        print( name, end ='\t')
        rec = file.readline()
        x= rec.split()
        while rec :
            print (x[-1],end = '\t')
            rec = file.readline()
            x= rec.split()
        print('\n',-*100)
        file.close()
    except IOError :
        time.sleep(1.5)
        print('No record found')
        time.sleep(1.5)

def display_tempexp_all(username) :
    data = open('Data Base\Temp_Exp '+username+'.txt','w')
    mo = ['Jan','Feb','March','April','May','June','July',
          'Aug','Sept','Oct','Nov','Dec']
    time.sleep(1.5)
    print('\n\n\tData Fetching in Progress\n\
Wait > It Might Take Few Seconds\n\n')
    time.sleep(1.5)
    print('DISPLAY ALL RECORDS')
    time.sleep(1.5)
    year = 2000
    while year!= 2200 :
        for j in mo :

```

```

while year!= 2200 :
    for j in mo :
        na = str(j)+str(year)
        f = 'Data Base\\'+na + username + ' Temp_Exp.txt'
        try:
            file = open(f, 'r')
            #graph()
            print(na)
            data.write(na + '\t')
            rec = file.readline()
            x = rec.split()
            while rec :
                data.write(x[-1]+ '\t')
                rec = file.readline()
                x= rec.split()
            data.write('\r')
            file.close()
        except IOError:
            io = 'IOE'
    year+=1
data.close()

#display
heading(username,' Temp_Exp')
data = open('Data Base\Temp_Exp '+username+'.txt','r')
rec = data.readline()
while rec :
    x = rec.split()
    for i in x :
        print(i,end = '\t')
    print('\n')
    rec = data.readline()
print('\n','-'*100)
data.close()

#Savings and planed expanditure
def savings(username):
    time.sleep(1.5)
    print('enter family time expanditures of this month, if any\n')
for example; F.D. Family Trip, Festival, Function and many more categories'
location = monthname(username) + ' Planed_Exp.txt'
file = open(location , 'w' )
total = 0
choice = 'yes'
while choice == 'yes' or choice == 'Yes' or choice == 'y' :
    time.sleep(1.5)
    a = input('Enter category of expanditure\t:\t')
    time.sleep(1.5)
    b = int(input('Enter amount\t:\t'))
    file.write(a + '\t' + str(b) + '\r')
    time.sleep(1.5)
    choice = input('Press (yes) if you want to enter more categories, else (no)\t:\t')
    total += b
file.write('Total'+'\t'+str(total))
time.sleep(1.5)
print ('Total Planned Expanditure\t:\t',total)
file.close()

def display_savings(username) :
    time.sleep(1.5)
    print('Saved and Planned Expanditure')
    location = monthname(username) + ' Planed_Exp.txt'
    file = open(location , 'r' )
    rec = file.readline()
    while rec:
        print(rec,end='\n')
        time.sleep(1.5)
        rec = file.readline()
    file.close()

```

```

#Analysis
def create_record(username) :
    data = open('Data Base\All Records '+username+'.txt','w')
    ext = [' Earning.txt',' Perma_Exp.txt',' Temp_Exp.txt',' Planed_Exp.txt']
    mo = ['Jan','Feb','March','April','May','June','July',
          'Aug','Sept','Oct','Nov','Dec']
    time.sleep(1.5)
    print('\n\n\tData Fetching in Progress\n\n')
    Wait > It Might Take Few Seconds\n\n)
    time.sleep(1.5)
    print('FETCHING ALL RECORDS')
    time.sleep(1.5)
    try :
        bank = open('Data Base\Initial Balance '+username+'.txt','r')
        A = bank.readline()
        balance = A.split()
        left = int(balance[0])
        bank.close()
    except IOError :
        time.sleep(1.5)
        print('No bank balance found')
        balance(username)
    year = 2000
    while year!= 2051 :
        for j in mo :
            na = str(j)+str(year)
            data.write(na +'\t')
            for type in ext :
                try:
                    file = 'Data Base\\'+na + str(username) + type
                    f = open(file,'r')
                    ##print('\n\nType   ', type)
                    rec = f.readline()
                    ##print('rec   ',rec)
                    x = rec.split()
                    ##print('x   ',x)
                    if rec == '':
                        break
                    else:
                        while rec:                                #to reach last term
                            y = x
                            rec = f.readline()
                            ##print('rec   ',rec)
                            x = rec.split()
                            ##print('x   ',y)
                            ##print('x last',y[-1],len(y))
                            x = y
                            if len(x) != 0 :
                                data.write(str(x[-1])+'\t')
                                ##print(x[-1])
                                if type == ext[0]:
                                    left += int(x[-1])
                                else :
                                    left -= int(x[-1])
                            f.close()
                except IOError:
                    io = None
                    data.write(str(0) + '\t')
                    data.write(str(left) + '\r')
                year+=1
                bank.close()
            data.close()
            bank = open('Data Base\Balance '+username+'.txt','w')
            bank.write(str(left))
            bank.close()
            time.sleep(1.5)
            print('\tRecord created sucessfully!!!\n\n')
            time.sleep(1.5)
            print('Balance left in your account\t:\t',left)

```

```

def analysis(username):
    create_record(username)
    time.sleep(1.5)
    print('DISPLAY ALL RECORDS')
    time.sleep(1.5)
    file = open('Data Base\All Records '+username+'.txt','r')
    year = int(input('Enter year to show analysis\t:\t'))
    time.sleep(1.5)
    if year< 100 :
        year += 2000
    mo = ['Jan','Feb','March','April','May','June','July',
          'Aug','Sept','Oct','Nov','Dec']
    print( '-'*110, '\nMonth\t\tEarning\tTemp Exp\tPerma Exp\tPlanned Exp\tBalance Left\n', '-'*110)
    time.sleep(1.5)
    earning, expenditure , balance = [] , [] , []
    for month in mo :
        na = month +str(year)
        line = file.readline()
        word = line.split()
        print(month,end= '\t\t')
        x = False
        while x != True :
            if word[0] == na :
                for i in range(5):
                    print(word[i+1],end = '\t\t')
                earning.append(int(word[1]))
                expenditure.append(int(word[2]) + int(word[3]) + int(word[4]))
                balance.append(int(word[5]))
                print('\n')
                x = True
            else :
                line = file.readline()
                word = line.split()
    file.close()
    print( '-'*110)
    time.sleep(1.5)
    graph('Earning Graph '+ str(year) , mo, earning)
    time.sleep(1.5)
    graph('Expedititure Graph '+str(year) , mo , expenditure)
    time.sleep(1.5)
    graph('Balance Variation Graph '+str(year), mo , balance)

def graph ( title , x , y ) :
    import matplotlib.pyplot as plt
    plt.plot(x,y,marker = 'D')
    plt.title(title)
    plt.xlabel('Month')
    plt.ylabel('Amount')
    for i in range(len(x)):
        # I rounded the y values as string and used the same x and y coords as the locations
        # next we can give a constant offset points to offset the annotation from each value
        # here I used (-20,20) as the offset values
        plt.annotate(f"{str(round((y[i])))}", (x[i],y[i]), xycoords='data',
                    xytext=(-20,20), textcoords='offset points')
    plt.show()

def balance(username) :
    file = open('Data Base\Initial Balance '+username+'.txt','w')
    time.sleep(1.5)
    bal = int(input('Enter you total bank Balance\t:\t'))
    file.write(str(bal))
    file.close()

```

```

def system(username) :
    time.sleep(1.5)
    print('\n\tSystem Menu\n\
Switch User-----1\n\
Change Password----2\n\
Delete Account----3')
    time.sleep(1.5)
    choice = int(input('Enter your choice\t:t:\t'))
    time.sleep(1.5)
    if choice == 1 :
        password()
    elif choice == 2 :
        create_password(username)
    elif choice ==3 :
        choice = input('Enter (yes) to continue deleting your account \t:t:')
        if choice == 'yes' or choice == 'Yes' :
            username = password()
            factoryreset(username)

def factoryreset(username):
    import os
    files = ['Data Base\\'+username + ' credential.txt',
              'Data Base\Recent_Permanent '+username+'.txt',
              'Data Base\\'+username + ' Temp_Exp.txt',
              'Data Base\Temp_Exp '+username+'.txt',
              'Data Base\Initial Balance '+username+'.txt',
              'Data Base\Balance '+username+'.txt',
              'Data Base\All Records '+username+'.txt']
    time.sleep(3)
    print('\n\n\tData Fetching in Progress\n\
Wait > It Might Take Few Seconds\n\n')
    time.sleep(3)
    print('DELETING ALL RECORDS')
    time.sleep(1.5)
    print('*'*60,'\n\tFiles Removed\n', '*'*60)
    time.sleep(1.5)
    for x in files :
        try :
            f = open(x, 'r')
            print(x)
            time.sleep(.25)
            f.close()
            os.remove(x)
        except IOError :
            IO = None
    ext = [' Earning.txt',' Perma_Exp.txt',' Temp_Exp.txt',' Planed_Exp.txt']
    mo = ['Jan','Feb','March','April','May','June','July',
          'Aug','Sept','Oct','Nov','Dec']
    year = 2000
    while year!= 2200 :
        for j in mo :
            na = str(j)+str(year)
            for type in ext :
                file = 'Data Base\\'+na + str(username) + type
                try :
                    f =open(file,'r')
                    print(file)
                    time.sleep(.25)
                    f.close()
                    os.remove(file)
                except IOError :
                    IO = None
            year +=1
    time.sleep(1.5)
    print('Account Deleted Sucessfully')
done(username)

```

```

def Add_Rec(username) :
    time.sleep(1.5)
    dict2 = {1 : input_earning, 2 : permaexp ,
            3 : addrec_tempexp , 4 : savings }
    choice = -1
    if choice != 0 :
        print('\n\tAdd Records\n\
Earning-----1\n\
Permanent Expandise-----2\n\
Temperary Expandise-----3\n\
Planned Expandise-----4')
        time.sleep(1.5)
        choice = int(input('Enter your choice\t:\t'))
        time.sleep(1.5)
        dict2.get(choice)(username)

def Display_Rec(username) :
    dict3 = { 1 : display_earning, 2 : displaypermaexp , 3 : display_tempexp_month ,
              4 : display_tempexp_all , 5 : display_savings }
    choice = -1
    if choice != 0 :
        time.sleep(1.5)
        print('\n\tDisplay Records\n\
Earning of a month-----1\n\
Currenet Permanent Expandise----2\n\
Temperary Expandise(by Month)---3\n\
Temperary Expandise(all Month)---4\n\
Planned Expandise-----5')
        time.sleep(1.5)
        choice = int(input('Enter your choice\t:\t'))
        time.sleep(1.5)
        dict3.get(choice)(username)

def done(username) :
    time.sleep(1.5)
    print('Have a good day ',username)
    time.sleep(1.5)
    quit()

# _____ MAIN MENU _____
import time
time.sleep(1.5)
print(' \n\t\t\t Welcome to House Management \n\t')
time.sleep(1.5)
username = password()
dict1 = { 0 : done , 1 : Add_Rec , 2 : Display_Rec, 3 : analysis , 4 : system}
choice = -1
while choice != 0 :
    print('\n\tMain Menu\n\
Quit-----0\n\
Add Records-----1\n\
Display Records-----2\n\
Complete Analysis-----3\n\
System Settings-----4')
    time.sleep(1.5)
    choice = int(input('Enter your choice\t:\t'))
    dict1.get(choice)(username)

```

All Output and Working

1. Creating/Logging in to the account

```
Welcome to House Management

Enter username      :      daksh
Username not found
Enter (yes) to create new account      :      yes

Get started
    Create new account .
Enter new username      :      daksh
    Create Password
Minimum length of password is 4
Enter New Password      :      1111
Conform Password        :      1111
Password created sucessfully  #
```

2. In a New Account entering the permanent expenditures according to users wish(these will be permanently stored in the account and user have to specify them only once and can even change it according to the wish)

```
Enter name and amount of permanent expanditure
    Example;House rent,EMI,School fees, gas cylinder etc
Enter name of expanditure      :      School Fees
Enter value      :      4300
Enter (yes) to input more records else(no)      :      yes
Enter name of expanditure      :      Mobile EMI
Enter value      :      2000
Enter (yes) to input more records else(no)      :      yes
Enter name of expanditure      :      Car EMI
Enter value      :      10000
Enter (yes) to input more records else(no)      :      yes
Enter name of expanditure      :      Internet Connection
Enter value      :      1000
Enter (yes) to input more records else(no)      :      no
```

3. For every month the fields would be fixed(can be changed) and the price has to put every month for temporary expenditures.

```

Temporary Expenditure
Description ;
      You can add categories of records of the expenditure which you make other than permanent expenditure which are not fix
For Example; Electricity , Fuel , Medicine , Ration and others categories.
Enter name of expenditure : Electricity
Press (yes) if you want to enter more categories else (no) : yes
Enter name of expenditure : Ration
Press (yes) if you want to enter more categories else (no) : yes
Enter name of expenditure : Medicine
Press (yes) if you want to enter more categories else (no) : yes
Enter name of expenditure : Clothing
Press (yes) if you want to enter more categories else (no) : yes
Enter name of expenditure : Fuel
Press (yes) if you want to enter more categories else (no) : yes
Enter name of expenditure : Outside Food
Press (yes) if you want to enter more categories else (no) : no

```

4. Before the main menu it asks for the current bank balance because the before purchasing/using the program the person might have some money already and then the main menu is displayed.

```

Enter your total bank Balance : 200000

Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : |

```

5. Now we must enter the records for a particular month.
a. Under this there are four categories of adding data 1st is the earning of the month.

```

Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 1

Add Records
Earning-----1
Permanent Expenditure-----2
Temporary Expenditure-----3
Planned Expenditure-----4
Enter your choice : 1
Enter the Month (XX) : 12
Dec
Enter the Year(XXXX) : 2022
Enter amount of earning : 80000

```

- b. The second option is to add permanent expenditure it by default shows the pre set values with a option to change it if user want.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 1

Add Records
Earning-----1
Permanent Expandise-----2
Temperary Expandise-----3
Planned Expandise-----4
Enter your choice : 2
Enter the Month (XX) : 12
Dec
Enter the Year(XXXX) : 2022
-----
Name      Expandise
-----
School    4300
Mobile    2000
Car      10000
Internet   1000
-----
Total Expandise : 17300
-----
Enter (yes) to continue else (no) to change values : no
School    "200
```

- c. Now we enter the temporary expenditure of that particular month that we are working on.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 1

Add Records
Earning-----1
Permanent Expandise-----2
Temperary Expandise-----3
Planned Expandise-----4
Enter your choice : 3
Enter the Month (XX) : 12
Dec
Enter the Year(XXXX) : 2022
Enter Electricity expanditure : 2500
Enter Ration expanditure : 3500
Enter Medicine expanditure : 2000
Enter Clothing expanditure : 6000
Enter Fuel expanditure : 6000
Enter Outside expanditure : 3000
Total Temperary expanditure : 23000
```

- d. And the last option under records is planned expenditure.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 1

Add Records
Earning-----1
Permanent Expandise-----2
Temporary Expandise-----3
Planned Expandise-----4
Enter your choice : 4
enter family time expanditures of this month, if any
for example; F.D. Family Trip, Festival, Function and many more categories
Enter the Month (XX) : 12
Dec
Enter the Year(XXXX) : 2022
Enter category of expanditure : Trip
Enter amount : 80000
Press (yes) if you want to enter more categories, else (no) : no
Total Planned Expandise : 80000
```

6. Display of records. It contains 5 fields.

- a. It displays the earning of the month.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 2

Display Records
Earning of a month-----1
Currenet Permanent Expandise----2
Temporary Expandise(by Month)---3
Temporary Expandise(all Month)---4
Planned Expandise-----5
Enter your choice : 1
Enter the Month (XX) : 12
Dec
Enter the Year(XXXX) : 2022
Entered earning was : 80000
```

- b. The second option is to display the permanent expenditure of a month.

```
        Display Records
Earning of a month-----1
Currenet Permanent Expanditure---2
Temperary Expanditure(by Month)---3
Temperary Expanditure(all Month)---4
Planned Expanditure-----5
Enter your choice      :      2
-----
Name          Expanditure
-----
School    4300
Mobile    2000
Car       10000
Internet   1000
-----
Total Expandise      :      17300
-----
```

- c. Third is to display temporary expenditure of a particular month.

```
        Display Records
Earning of a month-----1
Currenet Permanent Expanditure---2
Temperary Expanditure(by Month)---3
Temperary Expanditure(all Month)---4
Planned Expanditure-----5
Enter your choice      :      3
Display record of month
Enter the Month (XX)   :      12
                                Dec
Enter the Year(XXXX)   :      2022
-----
Month      Electricity      Ration      Medicine      Clothing      Fuel      Outside Total
-----
Dec2022daks  2500      3500      2000      6000      6000      3000      23000
-----
```

- d. Fourth is the display of temporary expenditure of all months in a year(Its only showing December because we have only registered 1 month in this account as the number of months increase this column will automatically display all months).

```
        Display Records
Earning of a month-----1
Currenet Permanent Expanditure---2
Temperary Expanditure(by Month)---3
Temperary Expanditure(all Month)---4
Planned Expanditure-----5
Enter your choice      :      4

        Data Fetching in Progress
Wait > It Might Take Few Seconds

DISPLAY ALL RECORDS
Dec2022
-----
Month      Electricity      Ration      Medicine      Clothing      Fuel      Outside Total
-----
Dec2022  2500      3500      2000      6000      6000      3000      23000
-----
```

- e. Last option is the display of planned expenditure of a month.

```
Display Records
Earning of a month-----1
Currenet Permanent Expanditure----2
Temperary Expanditure(by Month)---3
Temperary Expanditure(all Month)---4
Planned Expanditure-----5
Enter your choice      :      5
Saved and Planned Expanditure
Enter the Month (XX)   :      12                                Dec
Enter the Year(XXXX)   :      2022
Trip      80000

Total    80000
```

7. This the major part of our program which is complete analysis of the whole year in the form of graphs and tables. First a complete table for the year appears.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice      :      3
```

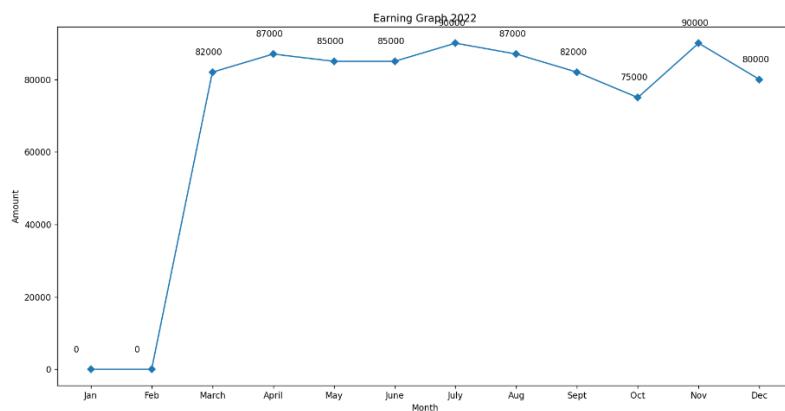
```
Data Fetching in Progress
Wait > It Might Take Few Seconds

FETCHING ALL RECORDS
Record created sucessfully!!!

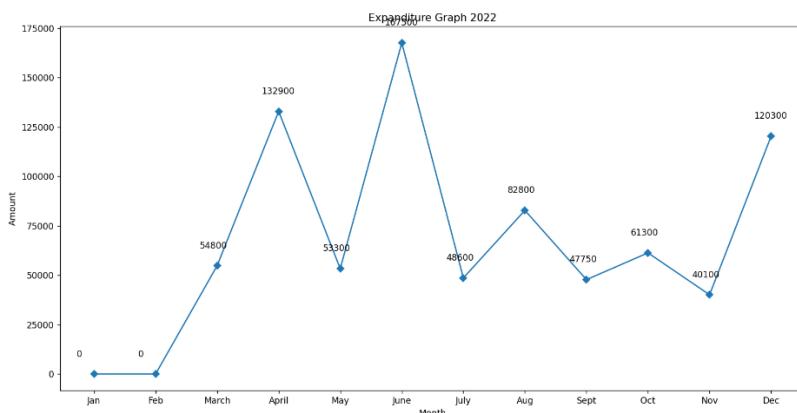
Balance left in your account      :      233650
DISPLAY ALL RECORDS
Enter year to show analysis      :      2022
-----
Month          Earning Temp Exp      Perma Exp      Planned Exp      Balance Left
-----
Jan            0           0           0           0           200000
Feb            0           0           0           0           200000
March          82000        25300        29500        0           227200
April          87000        25300        29600        78000        181300
May            85000        25300        28000        0           213000
June          85000        20300        27200        120000       130500
July          90000        20300        28300        0           171900
Aug            87000        25000        45800        12000       176100
Sept          82000        25000        22750        0           210350
Oct            75000        17300        32000        12000       224050
Nov            90000        17300        22800        0           273950
Dec            80000        17300        23000        80000       233650
-----
```

8. Under complete analysis only we have 3 graphs for easy comparisons.

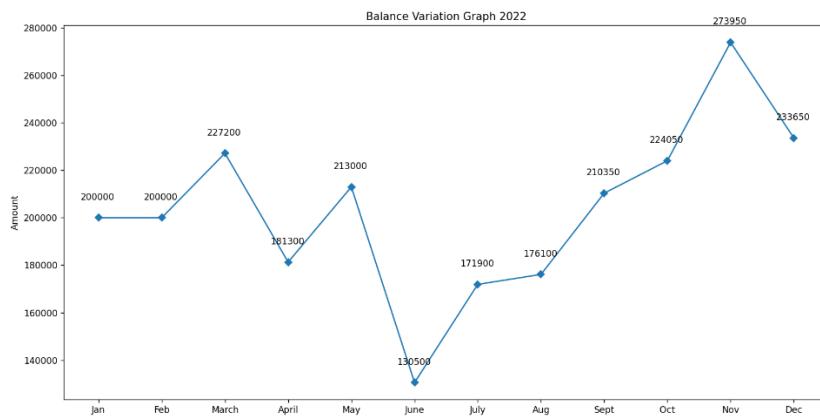
a. Earning Graph of the year.



b. Total Expenditure Graph of the year.



c. Balance Variation Graph of the year.



9. The Last we have the system settings which provide three options.

1. Switch User so that multiple users can operate the same program.
2. Change password for security reasons.
3. Delete Account.

```
Main Menu
Quit-----0
Add Records-----1
Display Records-----2
Complete Analysis-----3
System Settings-----4
Enter your choice : 4

System Menu
Switch User-----1
Change Password-----2
Delete Account-----3
Enter your choice : |
```

Bibliography

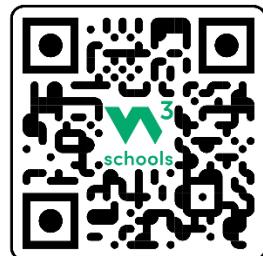
1. <https://www.geeksforgeeks.org/>



2. <https://stackoverflow.com/>



3. <https://www.w3schools.com/python/>



4. <https://www.youtube.com/watch?v=lHXdRPYWXzw>



5. [Preeti Arora](#)