PRACTICAL NO: 1

1) Demostrate the use of basic file accessing mode, file attributes and difference in the output produced by various read operation.

Algorithm:

Step 1:- Create a text file and write some content in it.

Step 2:- Use a variable which will read the content from the given file if the contents in the file are null it will throw exceptionerror to overcome this the read operation should be precessided with the right operation.

Step 3:- Close the opened file before changing the access mode.

Step 4:- Open the file in append mode and write the content in the existing file and display the contents simultaneously.

Step 5:- Use the read, readline and readlines method and make comment on the output difference.

Code

Write Method

```
fileobj = open("prac1.txt", "w")
fileobj.write("file attribute" "\n")
fileobj.write("Accessing mode", "\n")
fileobj.write("Text file" "\n")
fileobj.write("Basic mode" "\n")
```

Incomplete for string

Read Method

```
fileobj- open("prac1.txt", "r")
str1 = fileobj.read()
print("The contents of the file are:", str1)
fileobj = open("prac1.txt", "r")
print("readline:", fileobj.readline())
fileobj = open("prac1.txt", "r")
print("readlines:", fileobj.readlines())
fileobj.close()
```

>>> The contents of file are: file attribute
Accessing mode
Text file
Basic mode

Readline: file attribute

readlines: ['file attribute\n', 'Accessing mode\n', 'Text file\n', 'Basic mode\n']

append mode

```
fileobj - open("
```

Algorithm

Step1: Create a file object and open the file in the r+ mode.

Step2: Use the read() to read the character from the file store it in the variable and display the content of the variable.

Step3:- Use tell() along with fileobject and store the output in a variable and subsequently display the content of the variable.

Step4: Now use seek() with 2 arguments as 0 & then read certain no of characters & display the read characters.
Similary use the seek with 0 & 1 values indicating the reference position of the curser and the current location while 0,2 value indicates the references position of the curser from the end of the file.

Tell and Seek Method

```
f1 = open("prac1.txt", "r+")
str1 = f1.read(20)
print("Read 20:", str1)
position = f1.tell()
print("Tell method:", position)
pos1 = f1.seek(0,0)
print("Seek method:", pos1)
str1 = f1.read(15)
print("Read 15:", str1)
pos2 = f1.seek(0, 1)
print("seek method:", pos2)
str2 = f1.read(10)
print("Read 10", str2)
pos3 = f1.seek(0, 2)
print("seek method:" pos3)
str3 = f1.read(20)
print("Read 20", str3)
```

>>> Read 20: file attributre
Tell method: 21
seek method: 0
Read 15: file attribute
seek method: 16
Read 10: Accessing
Seek method: 55
Read 20:

Algorithm

Step 1:- Create a file object and open it in the read mode and subsequently display the content of the file in a sequential manner by using the conditional statement.

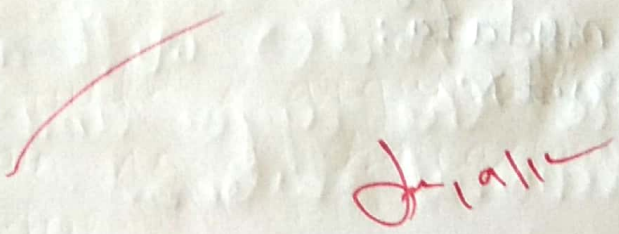
Step 2:- Initialize a counter variable to zero value and read the content from the given file upto the character specified.

Step 3:- Use the while conditional statement to check whether the content of the file is exausted.

Code

```
f= open("ch.txt", "w")
f.write("come here")
f.close()
with open("ch.txt", "r") as g:
    s=1
    c = g.read(s)
    while len(c)>0:
        print(c, end="*")
        c=g.read(s)
```

```
>>> c*o*m*e*  *h*e*r*e
```



PRACTICAL NO: 2

1) Write a program using iterable objects for displaying the number in the range 1 to 30.

Step 1: Define an iter() with argument and initialize the value and return that value.

Step 2: Define the next() with an argument and compare the upper limit by using a conditional statement.

Step 3: Now create an object of the given class and pass this object in the iter()

Code
53

```
class myrange:
    def __iter__(self):
        self.a = 1
        return self
    def __next__(self):
        if self.a <= 30:
            x = self.a
            self.a += 1
            return x
        else:
            raise StopIteration

myclass = myrange()
mydata = iter(myclass)
for x in mydata:
    print x.
```

```
>>>
1
2
3
4
5
6
7
8
.
.
.
.
```

Write a program using the iterator for calculating power of 2 which is $2^1, 2^2, 2^3, 2^4, 2^5$

Algorithm:

Step 1: Define an iter() with argument and initialize the value and return that value.

Step 2: Define the next() with an argument and compare the upper limit by using a conditional statement.

Step 3: Now create an object of the given class and pass this object in the iter method.

Code:

```
class power:
    def __iter__(self):
        self.p = 0
        return self
    def __next__(self):
        if self.p <= 10:
            num = self.p
            self.p += 1
            po = 2 ** num
            print("2**", self.p-1, "=", po)
            return po
        else:
            raise StopIteration

p = power()
x = iter(p)
for j in x:
    print(j)
```

```
>>> 2^0 = 1
1
2^1 = 2
2
2^2 = 4
2^3 = 8
8
2^4 = 16
16
2^5 = 32
32
2^6 = 64
64
2^7 = 128
128
2^8 = 256
```

```
2^9 = 512
512
2^10 = 1024
1024
```

Write a program using iterable concept find the factorial of the number in range of 1 to 10.

Algorithm

Step1:- define an iter() with argument and it initialize the value and return that value

Step2:- Define the next() with an argument and compare the upper limit by using a conditional statement.

Step4:- Now create an object of the given class and pass this object in the iter method.

1/11/24

Code

```
class fact:
    def __iter__(self):
        self.f=1
        return self
    def next(self):
        if self.f<=10:
            num=self.f
            self.f+=1
            fact=1
            for i in range(1,num+1):
                fact=fact*i
            print(self.f-1,"!=",fact)
        else:
            raise StopIteration
f=fact()
x=iter(f)
for j in x:
    print(j)
```

>>> 1 ! = 1

2 ! = 2

3 ! = 6

4 ! = 24

5 ! = 120

6 ! = 720

7 ! = 5040

8 ! = 40320

9 ! = 362880 10 ! = 3628800

18

PRACTICAL No :- 3

EXCEPTION HANDLING

1) Write a program using the exception method of the nature arithmatic error.

Algorithm :-

Step1:- Use try block and accept the input using input method and convert it into integer datatype and subsequently terminate the block.

Step2:- Use the accept block with the exception name as value error and display the appropiate message if the suspesious code is a part of the try block.

2) Write a program for accepting file in a given mode and

Code

```
# arithmatic error
while True:
    try:
        n = int(input("Enter class: "))
        break
    except ValueError:
        print("Enter Numeric Value")
```

```
>>> Enter class: fycs
Enter Numeric value
Enter class: 13
```

2) Write a program for accepting file in a given mode and use environment error as exception for the given input.

Algorithm

Step1: Within the try block open the file to using the write mode and write some content onto the file.

Step2: Use the except block with IO error and display the message regarding missing of the file or incompatibility of the mode.

Step3: Use the else block to display the message that the operation is carried out successfully

code:

```
try:
    fo = open("abc.txt", "w")
    fo.write("Computer Science")
except IOError:
    print("Error writing onto the file")
else:
    print("Operation carried out successfully")
    fo.close
```

>>>

Operation carried out successfully

3) Write a program to check the range of the age of students in a given class and if the age do not fall in the given range use the value error exception otherwise return the valid number.

Algorithm

Step1: Define a function which will accept the ~~input~~ age of the student from the standard input.

Step2: Use the if condition to check whether the input age falls in the given age and if so return the age else use the value error exception.

Step3: define while loop to check whether the boolean expression holds true use the try block to accept the age of student and terminate the looping condition.

Step4: Use the except with valueerror and print the message not a valid age.

code

```
def acceptage():
    age = int(input("Enter age:"))
    if age>30 or age<16:
        raise ValueError
    return age
valid = False
while not valid:
    try:
        age = acceptage()
        valid = True
    except ValueError:
        print("Not a valid age")
```

```
>>>
Enter age: 10
Not a valid age
Enter age: 18
```

16/07

4) Write a program using the assert method to check if the list elements are empty.

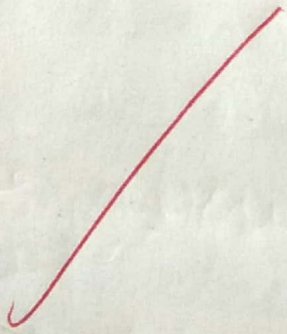
Step1: Define a function which accepts an argument and then check using the assert statement whether the given list is an empty list and accordingly return the message.

Step2: Close the function and in the body of program define certain elements in the list and take some appropriate action.

code:

```
def assert_(n):
    assert (len(n) == 0)
    print ("list is empty")
var1 = [ ]
print (assert_(var1))
```

>>> list is empty
None



PRACTICAL NO: 4

REGULAR EXPRESSION

1) Write a regular expression for segregating the numeric and alphabatical values from the given string.

Step Apply the string and pattern in the find all method and display the output \d is used for matching all the decimal digits whereas \D is used for matching all the non decimal digits.

2) Write a regular expression for finding all the matched string at the begining of the given sequence.

To achieve the above objective we will be using search()
To check the condition we will use if else conditional statement.

1) Code

```
import re
string = "hello1234 abc345"
pattern = "\d+"
pattern1 = "\D+"
result = re.findall(pattern, string)
result1 = re.findall(pattern1, string)
print(result)
print(result1)
```

Output

```
['123', '345']
['hello', 'abc']
```

2) Code

```
import re
string = "Python is interpreted language"
result = re.search('xyz', string)
if result:
    print("Match found")
else:
    print("Not found")
```

Output

Match not found

3) Write a regular expression to check whether the given mobile no starts with digit 8 or 9 and the total length of the digit should be atmost 10.

To achieve the above objective we will use match()
To extract value & check condition we will use for and if else conditional statement.

4) Write a re for extracting the word from a given string along with the space character between the word and word without space.

To achieve the above objective we will be using using findall().
To extract the word along with space we will use \w* without space \w+

5) Write a regular expression for extracting the first word and last word from a given string.

To achieve the above objective we will be using findall()
To extract the first word we will use ^\w+ and last word by \w+$

3) Code

```
import re
li = ["12345671891", "8945634334", "9345678193"]
for val in li:
    if (re.match("[8-9]{1}[0-9]{9}", val)):
        print ("Correct phone no")
    else:
        print ("Incorrect phone no")
```

Output >>> Incorrect phone no
Correct phone no
Correct phone no

4)

```
Import re
string = "Python is important"
result1 = re.findall(r"\w*", string)
print (result1)
result2 = re.findall(r"\w+", string)
print (result2)
```

Output

```
>>> ['Python', '', 'is', '', 'important']
    ['Python', 'is', 'important']
```

5)

```
Import re
String = "Python is important"
result1 = re.findall("^\w+", string)
print (result)
result2 = re.findall("\w+$", string)
print (result2)
```

Output >>> ['Python']
['important']

6) Write a re for extracting date in the format dd/mm/yyyy by using the find all method where the string has format.

To achieve the above objective we will use findall()
To extract the value we will use the pattern \d{2}-\d{2}-\d{4}

7) Write a regular expression for extracting the username from the mailID, hostname, and both username and hostname.

6) Code

```
import re
String = "Amit 201 02-09-2001, Ian 202 01-09-2001"
pattern = "\d{2}-\d{2}-\d{4}"
result = re.findall(pattern, string)
print(result)
```

Output

```
>>> ['02-09-2001', '01-09-2001']
```

7) Code

```
string = "abc@gmail.com" "abc@test-gmail.com"
p1 = "^\w+"
p2 = "t\w+."
p3 = "[\w\.-]+"
r1 = re.findall(p1, string)
r2 = re.findall(p2, string)
r3 = re.findall(p3, string)
Print(r1)
Print(r2)
Print(r3)
```

Output:

```
>>> ['abc']
    ['gmail.com']
    ['abc', 'gmail.com']
```

PRACTICAL NO. 5

GRAPHICAL USER INTERFACE (GUI)

1) Write a program to implement label and font on the parent window.

Algorithm:

1: Import relevent method from tkinter library.

2: Create an object corresponding to the parent window using TK()

3: Create an ~~font~~ object for label method and place it onto parent window. Now use pack method for positioning of label.

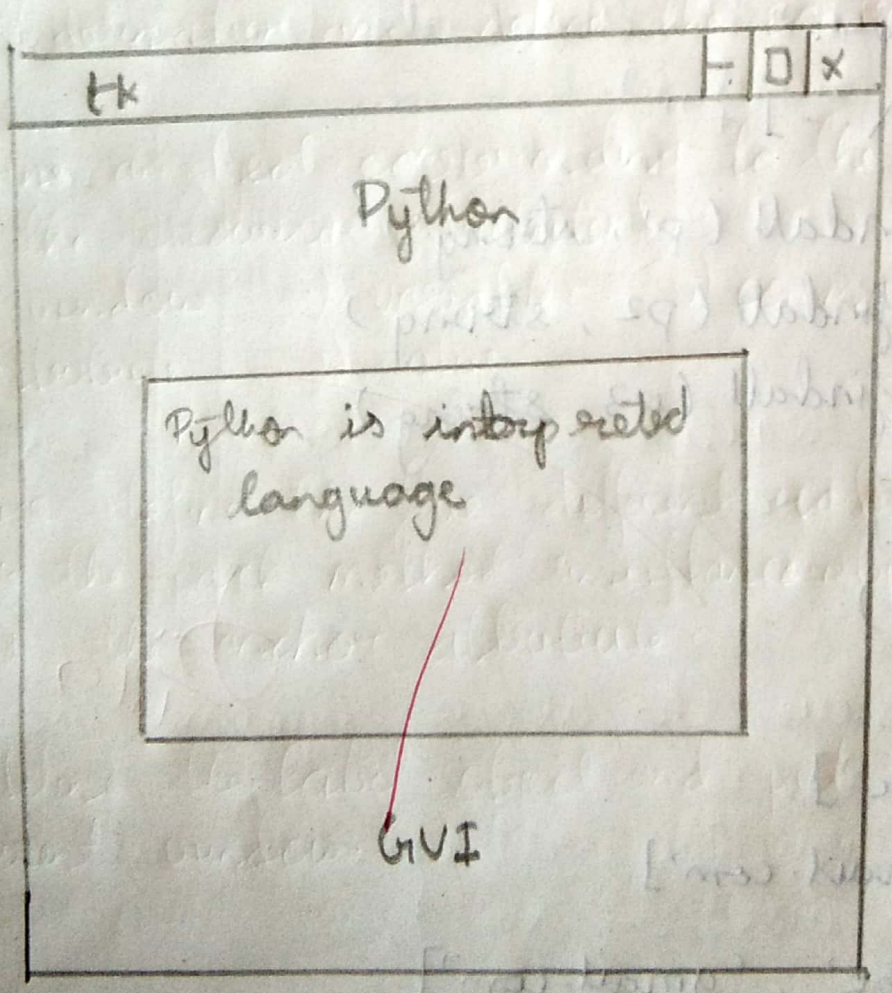
4: Similarly create ~~and~~ an another label object now in pack() use attributes of internal padding.

5: Create an object for font method and place it onto the parent window. Use the insert() method for applying string in the font.

6: Now call the mainloop().

Code :

```
from tkinter import *
root = Tk()
l1 = Label (root, text = "Python")
l1.pack (padx = 20, pady = 50, side = TOP)
l2 = Label (root, text = "GUI")
l2.pack (ipadx = 50, ipady = 80, side = BOTTOM)
t1 = Text (root)
quote = "Python is interpreted language"
t1.insert (END, quote)
t1.pack (padx = 70, pady = 90, side = RIGHT)
root.mainloop()
```



E.8:

2) Write a program making use of the radiobutton and the control variable for selection of the given choice.

Algorithm

Step 1:- Import the relevent method from tkinter library

Step 2:- Define a function and define a variable which will keep track of option selected.

Step 3:- Use the config() along with the label object and use the attribute for displaying the selection name.

Step 4:- Now define the parent window object outside the function defination and define the control variable.

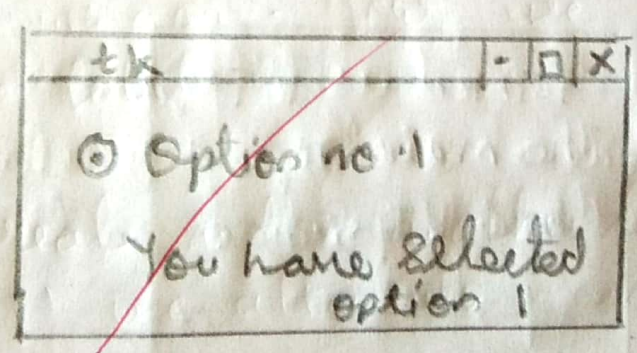
Step 5:- Now define an object corresponding to the radiobutton with the following attributes.
i) parent window iii) variable v) command.
ii) Text attribute iv) value

Step 6:- likewise define the five different radio button and use the pack method simultaneously by defining the anchor attribute.

Step 7:- Now define the label object and put it onto the parent window.

Code

```
from tkinter import *
def sel():
    selection = "You selected the option " + str(var.get())
    label.config(text = selection, justify = LEFT)
root = Tk()
var = IntVar()
r1 = Radiobutton(root, text = "Option no: 1", variable = var,
                 value = 1, command = sel).
r1.pack(anchor = W)
label = Label(root)
label.pack()
root.mainloop()
```



3) Write a program implementing the scrolling feature using the relevent python method.

Algorithm

1:- import the relevent method from tkinter library.

2: Create an object corresponding to the parent window using TK()

3: Create an object from scrollbar and place it onto the parent window.

4: Create an object from the text method placing it onto the same parent window with the height and width attribute specified.

5: Use the pack() with the argument side and fill.

6: Create an object from the scrollbar method and use the pack() with the side and the fill attribute Similarly with the text object use the pack()

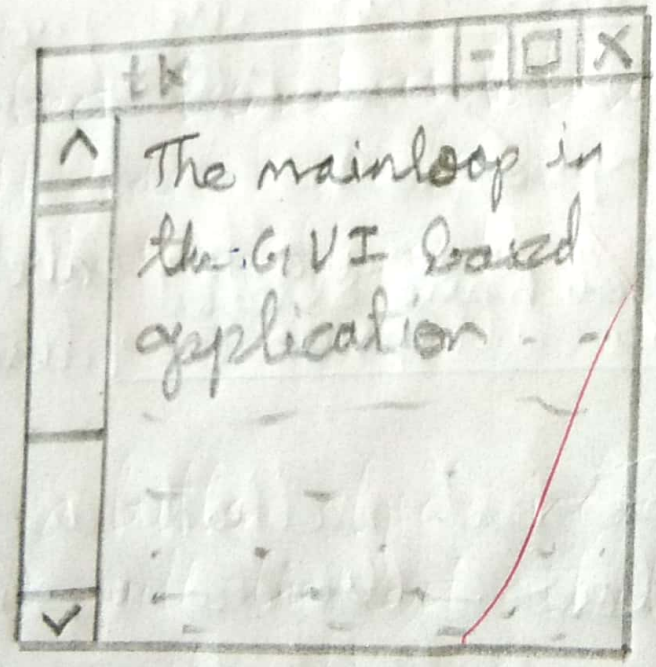
7. Now use the config() along with the scrollbar object and use the command argument.

8:- Similarly use the config() with text object and use yscrollcommand.

9. Now define string variable and use the insert method along with text object and use the mainloop()

Code

```
from tkinter import *
root = Tk()
paragraph = "The mainloop in the GUI based..
s = Scrollbar(root)
t = Text(root, height = 10, width = 20)
t.pack(side = RIGHT, fill = Y)
s.pack(side = RIGHT, fill = Y)
s.config(command = t.yview)
t.config(yscrollcommand = s.set)
t.insert(END, paragraph)
root.mainloop()
```



4) Write a program for implementing a frame widget and explain its relevance.

Algorithm

Step 1: import relevent method from tkinter library.

2: Create an object corresponding to the parent window using TK()

3: Create an object from frame method and place such an object onto the parent window so created.

4. Use the pack() for positioning the widget onto the parent window.

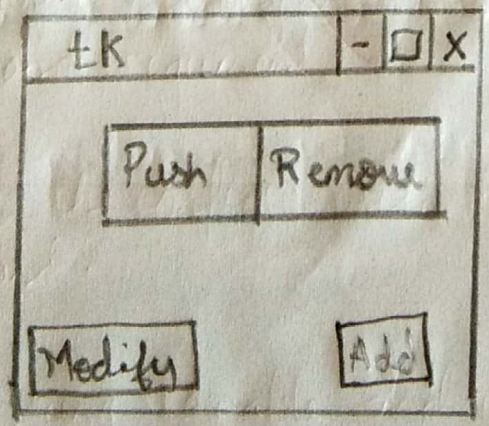
5: Create an object termed as leftframe and position it onto the left side of window: Similarly create the rightframe object and position on the right side of window.

6: Now create a button object and place it onto the frame widget with the text attribute fg & bg and position this on the left side.

7: Similarly create other button object named remove put it on frame and position on right side. Further create add button put it onto right frame and on left side and create modify button on leftframe and position on right side and call the mainloop()

Code

```
from tkinter import *
root = Tk()
frame = Frame(root)
frame.pack(padx = 20, pady = 50, side = TOP)
leftframe = Frame(root)
leftframe.pack(side = LEFT)
rightframe = Frame(root)
rightframe.pack(side = RIGHT)
buttonpush = Button(frame, text = "Push", activebackground = "red",
                    fg = "blue")
buttonpush.pack(side = LEFT)
buttonremove = Button(frame, text = "Remove")
buttonremove.pack(side = BOTTOM)
buttonadd = Button(rightframe, text = "Add")
buttonadd.pack(side = LEFT)
buttonmodify = Button(leftframe, text = "Modify")
buttonmodify.pack(side = RIGHT)
root.mainloop()
```



PRACTICAL NO:- 6

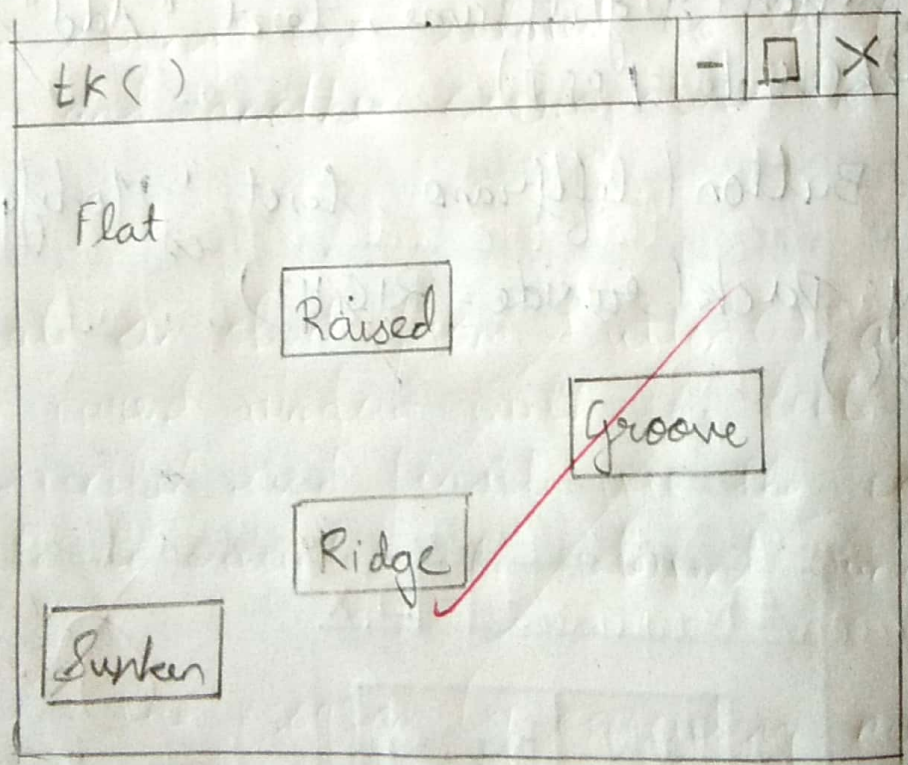
COMPONENTS OF GUI (BUTTON ATTRIBUTE, MESSAGE BOX)

Write a program on various attributes which a button widget may assume related to the relief attribute.

Algorithm:-

1. Define a Import relevent method from tkinter library.
2. Create an object corresponding to the parent window using TK()
3. Define a button object and place it onto the object corresponding to the parent window.
4. Use the text attribute for specifying the title to the button object.
5. Use the relief attribute with one style at a time involved for event triggering.
6. Use either the pack() or grid() for positioning the widget object onto the parent window & and trigger the corresponding event by calling the mainloop().

```
from tkinter import *
top = Tk()
B1 = Button (top, text = "Flat", relief = FLAT)
B1.pack (side = LEFT)
B2 = Button (top, text = "Raised", relief = RAISED)
B2.pack()
B3 = Button(top, text = "Groove", relief = GROOVE)
B3.pack(side = RIGHT)
B4 = Button (top, text = "Ridge", relief = RIDGE)
B4.pack()
B5 = Button (top, text = "Sunken", relief = SUNKEN)
B5.pack(side = LEFT)
top.mainloop()
```



2) Write a program for messagebox widget and the different methods which this widget may assume

Algorithm:

1:- Import relevent method from tkinter library.

2:- Create an object corresponding to the parent window using Tk()

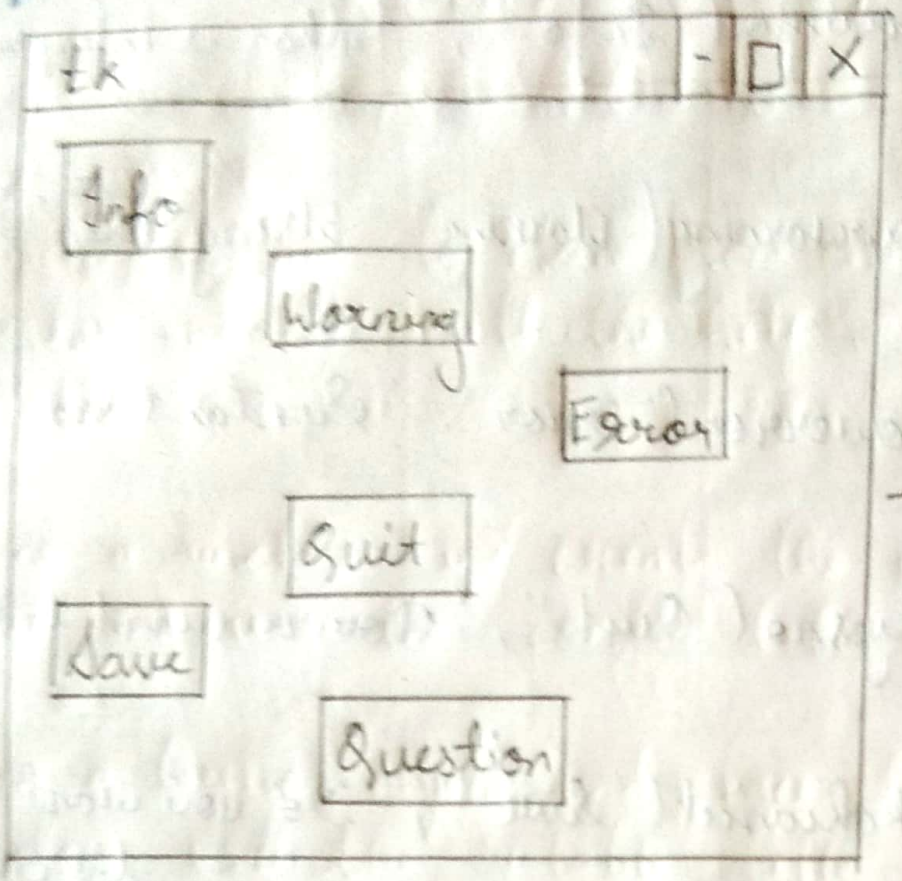
3:- Define a function which will use the one of the 6 method derived from the messagebox library.

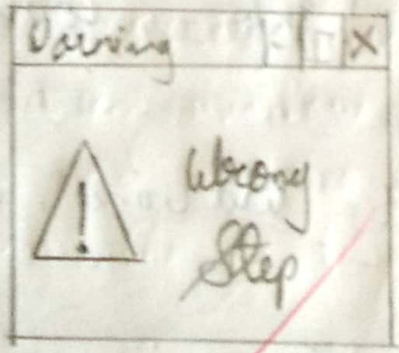
4:- The attribute which a given method takes will specify the two strings one related to the title of the window and another corresponds to the message display.

5:- Now create an object from Button() and place it onto the parent window with the title of button object specified and finally use the command attribute to execute the relevent function.

6:- Terminate the program by executing mainloop()

```
from tkinter import *
top = TK()
def msgb1():
    messagebox.showinfo("Info", "Python is Interpreted lang")
def msgb2():
    messagebox.showwarning("Warning", "Wrong Step")
def msgb3():
    messagebox.showerror("Error", "Syntax Error")
def msgb4():
    messagebox.askyesno("Quit", "Do you want to Quit?")
def msgb5():
    messagebox.askokcancel("Save", "Do you want to save
                                    the changes?")
def msgb6():
    messagebox.askquestion("Question", "Do you want to save th
                                        changes?")
B1 = Button(top, text="Info", command = msgb1)
B1.pack(side = LEFT)
B2 = Button(top, text="Warning", command = msgb2)
B2.pack()
B3 = Button(top, text="Error", command = msgb3)
B3.pack(side = RIGHT)
B4 = Button(top, text="Quit", command = msgb4)
B4.pack()
B5 = Button(top, text="Save", command = msgb5)
B5.pack(side = LEFT)
B6 = Button(top, text="Question", command = msgb6)
B6.pack()
top.mainloop()
```





Message Box

PRACTICAL No:- 7

TRAVERSAL OF WINDOW

1) Write a program to move from one window to another window with the help of the Button.

Algorithm:-

Step 1:- Define a function and create the parent window object and use the config, title and minsize method

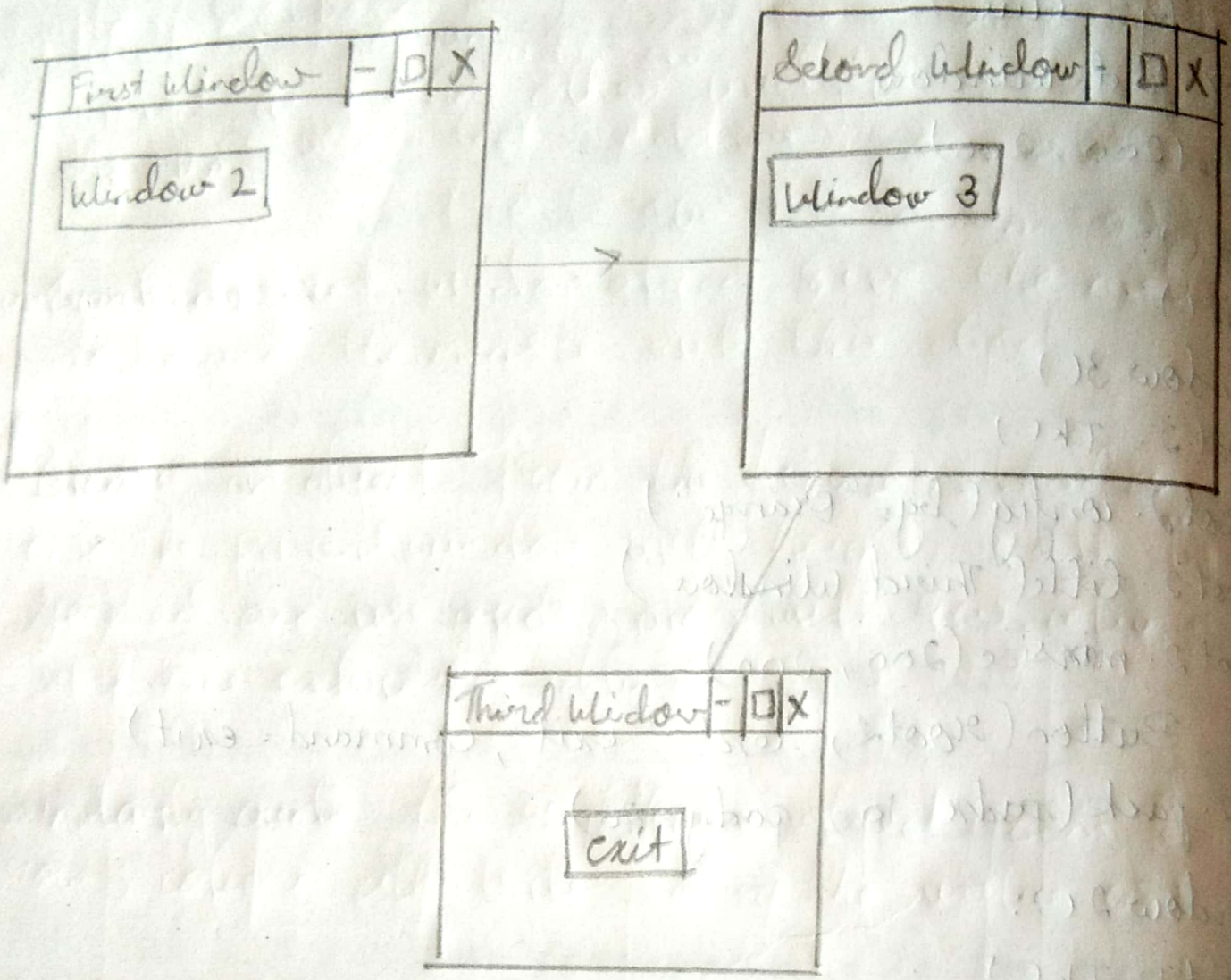
Step 2:- Define a button object and place it onto the parent window with the suitable title and the command attribute to call the next function with grid method specifying the external padding.

Step 3:- Now define a function corresponding to the second window and create another parent window object at with the method config, title and minsize again place the Button object calling the next function.

Step 4:- Similarly create the function and use the Button widget and finally create a function which will terminate the application by using the quit method.

Code:

```
from tkinter import *
x = Tk()
x.config(bg = "Blue")
x.title("first window")
x.minsize(200, 200)
def exit():
    quit()
def window3():
    root2 = Tk()
    root2.config(bg = "orange")
    root2.title("Third window")
    root2.minsize(200, 200)
    b2 = Button(root2, text = "exit", command = exit)
    b2.pack(padx = 10, pady = 20)
def window2():
    root1 = Tk()
    root1.config(bg = "pink")
    root1.title("second window")
    root1.minsize(200,200)
    b1 = Button(root1, text = "window3", command = window3)
    b1.grid(padx = 10, pady = 20)
b = Button(x, text = "window 2", command = window2)
b.grid(padx = 10, pady = 2)
x.mainloop()
```



2) Write a program for insertion of image in frame widget using the other widgets.

Algorithm:

Sp1:- Create a parent window object & use the method title, config and the minsize with this object

Sp2:- Create an object from the frame method and place it onto the parent window with height, width, bg colour specified and use grid() along with row, column as 0, 0 with some external padding.

Sp3: Similarly create the rightframe object from frame() with row & column attribute taking the value 0 & 1.

Sp4:- Use the a label() & the parent window object correspond to leftframe with text & relief attribute specified row, column value as 0, 0 in grid().

Sp5: Similarly create the label corresponding to rightframe and use the title & row, column value as 0

Sp6: Use the photoimage() with file attribute specified and subsequently ~~the~~ use the subsample method for specifying the image object

Sp7: Now use the label() using the leftframe and the image attribute and the row, column value specified in grid().

8:- Similarly create the label() using the rightframe object with the image attribute and some bg colour with the row, column value specified as 0,0.

9:- Now define a function using the print statement which shall be called on clicking the button

10:- Create an object from frame() named as toolbox and place it onto the leftframe with row column value as 2,0

11:- Now create the Button object and place it onto the toolbox object with the text and the command attribute specified.

12:- To display the toolbox onto the given frame use the label method with the toolbox object representing the parent window and specifying text and relief attribute with row, column value as 0,0 & to make the more options visible the row and column value need to be changed.

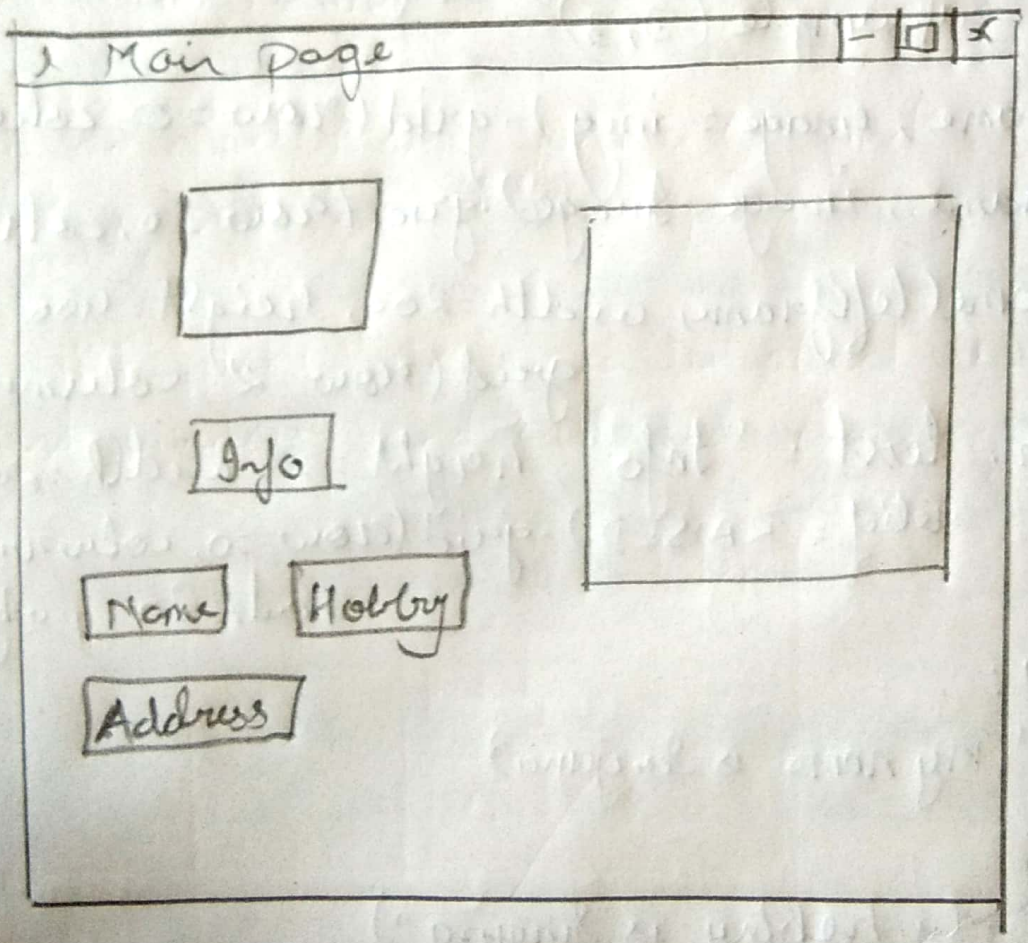
```
from tkinter import *
root = Tk()
root.title("Main page")
root.minsize(1000, 900)
root.config(bg="black")
leftframe = Frame(root, bg="Blue", height=400, width=200)
leftframe.grid(row=0, column=0)
rightframe = Frame(root, height=400, width=200, bg="red")
rightframe.grid(row=0, column=0)
Image = PhotoImage(file="image.gif")
img = Image.subsample(3,3)
Label(leftframe, image=img).grid(row=0, column=1)
Label(rightframe, image=Image).grid(row=0, column=1)
toolbar = Frame(leftframe, width=200, height=400).
                                grid(row=2, column=0)
Label(toolbar, text="Info", height=2, width=20,
      relief=RAISED).grid(row=0, column=0,
                          padx=20, pady=20)

def name():
    print("My name is Sanjana")
def hob():
    print("My hobby is Singing")
def add():
    print("Borivali")

Button(toolbar, text="Name", command=name).grid(row=1,
                                column=0)
```

```
Button(toolbox, text="Hobby", command=hob)
        .grid(row=1, column=1)
Button(toolbox, text="Address", command=add)
        .grid(row=2, column=0)
root.mainloop()
```

Output :-



CP

PRACTICAL NO: 8

COMPONENTS OF GUI

1) Write a program to make use of spinbox widget

Algorithm

Step 1: Use the tkinter library to import the relevent methods.

Step 2: Create the parent window object.

Step 3: Create an object from spinbox() and place it on the parent window with the option specified.

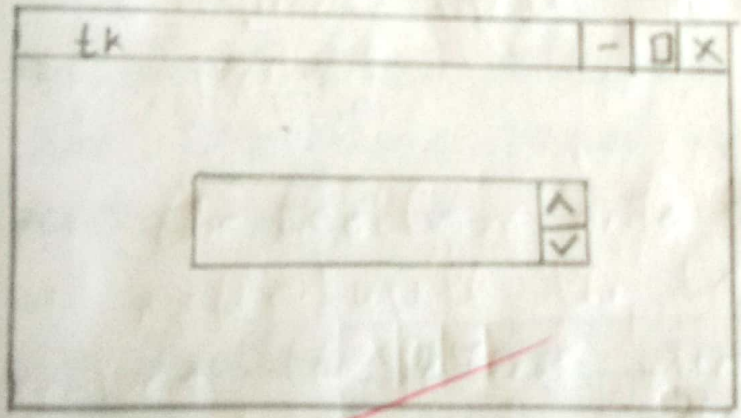
Step 4: Now use pack() to make the object visible onto the parent window and call the mainloop()

SPINBOX

Code:

```
from tkinter import *
root = Tk()
s1 = Spinbox(root, bg="orange")
s1.pack(padx=50, pady=50)
root.mainloop()
```

Output



2) Write a program to make use of panned window

Algorithm

1:- Create an object from panned window() and use the pack method to make this object visible.

2. Now create an object from the entry widget and place it onto the panned window and use the add method.

3- Similarly create an object of the panedwindow and add it onto the existing window.

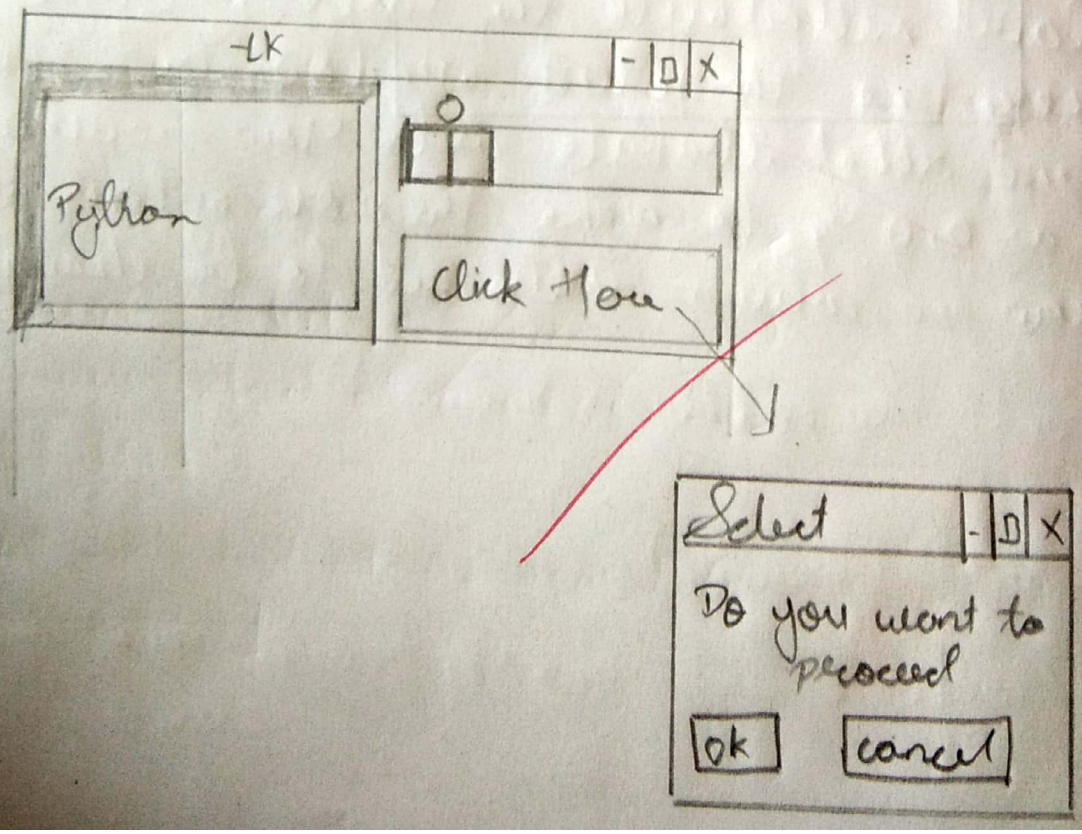
4:- Create an object from the scale method and place it onto the preceding paned window and use the add() accordingly.

5:- Create a button widget and place it onto the panedwindow and define a functionality along with the button widget.

6:- Use the pack() and mainloop() method for the corresponding event to be triggered.

Code:

```
from tkinter import *
root = Tk()
def msgb:
    messagebox.askokcancel("select", "Do you want to
                            proceed with program")
m1 = PanedWindow()
m1.pack(fill = BOTH, expand = 1)
e = Entry(m1, bd = 6)
m1.add(e)
m2 = PanedWindow(m1, orient = VERTICAL)
m1.add(m2)
top = Scale(m2, orient = HORIZONTAL)
m2.add(top)
Button1 = Button(m2, text = "Click Here", command = msgb)
m2.add(Button1)
root.mainloop()
```



3) Write a program to make use of canvas.

Algorithm:-

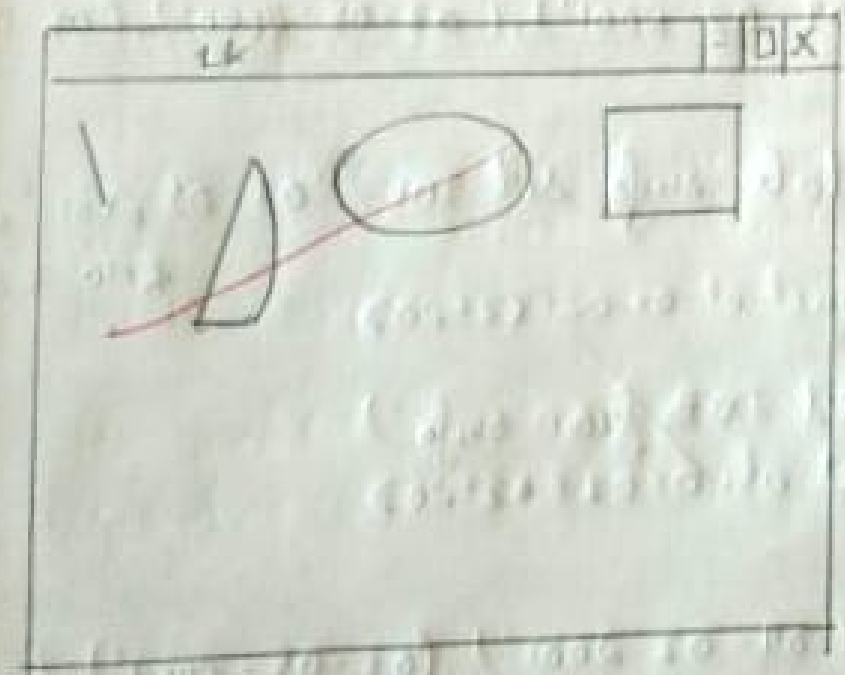
Step 1:- Create an object from the canvas widget by using the attribute height, width, bg colour and the parent window object.

Step 2:- Use the corresponding method for drawing the simple geometrical shapes like arc, oval, and line and specify the co-ordinate value.

Step 3:- finally use the pack() and mainloop().

Code

```
from tkinter import *
root = Tk()
c = Canvas(root, bg="Orange", height="60", width="80")
arc = c.create_arc(30,200,90,10, start=0, extent=50,
                   outline="White", fill="red")
line = c.create_outline(10, 20, 30, 40)
rec = c.create_rectangle(220, 10, 290, 80, outline="blue")
oval = c.create_oval(110,210,210,80, outline="Black",
                     fill="White")
c.pack(padx=200, pady=200)
root.mainloop()
```



PRACTICAL NO: 9

DATABASE CONNECTIVITY

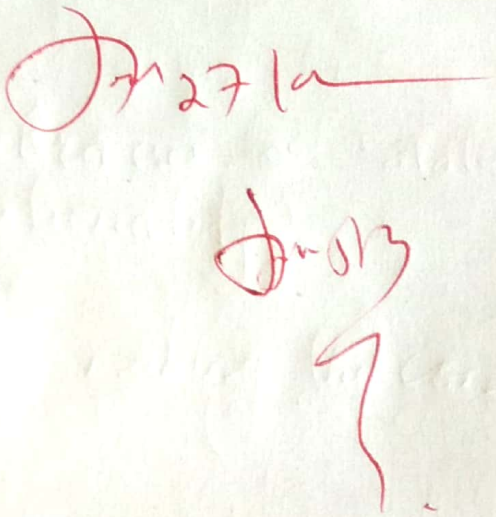
Algorithm:

1. Import the relevent libraries for Database and the operating system functionality.
2. Now create an object for making the connection to the given Database.
3. further create object corresponding to the cursor area for execution of the different query statement.
4. Use the cursor object so created for implementing the structure of the database and the values within the DB.
5. Use the execute() for implementation of the select clause for filtering the information.
6. Now use the fetchall() along with the cursor object for displaying the value onto the screen.

Code:-

```
>>> import os, sqlite3
>>> conn = sqlite3.connect("Student.db")
>>> cur = conn.cursor()
>>> cur.execute('create table sinfo(RNO int, Name text,
                                    DOB date)')
<sqlite3.Cursor object at 0x02F62F20>
>>> cur.execute('insert into sinfo values(01, "Sanjana",
    "02-09-2001"),(02, "Pooja","09-02-2001"),(03, "Bhavna",
    "03-05-2000"),(04, "Ram","05-06-2000",(05, "Lavi",
    "03-09-2002")')
<sqlite3.Cursor object at 0x02F62F20>
>>> cur.execute('Select DOB from sinfo')
<sqlite3.Cursor object at 0x02F62F20>
>>> cur.fetchall()
[('02-09-2001'),('09-02-2001'),('03-05-2000'),('05-06-2000'),
 ('03-09-2002')]
>>> cur.execute('update sinfo set DOB="03-05-2001" where
                                    RNO=03')
<sqlite3.Cursor object at 0x02F62F20>
>>> cur.execute('Select DOB from sinfo')
<sqlite3.cursor object at 0x02F62F20>
>>> cur.fetchall()
[('02-09-2001'),('09-02-2001'),('03-05-2001'),('05-06-2000'),
 ('03-09-2002')]
```

>>> cur.execute('alter table sinfo add Address ~~sinfo~~ 51 test')
<sqlite3.cursor object at 0x02F62F20>
>>> cur.close()



PROJECT
GUI BASED APPLICATION
TOPIC :- MUSIC REVIEW APP

```
fromtkinter import *
def review1():
 motsp=Tk()
 motsp.config(bg="pink")
 motsp.title("Map of the soul:Persona album Review")
 motsp.minsize(1000,900)
 title=Label(motsp,text="BTS").grid(sticky="n")
 cover=PhotoImage(file="motsp.gif")
 Label(motsp,image=cover).grid(row=1,column=0,sticky="e")
 info="Released:-April 12, 2019\nStudio:-Big Hit (Seoul),The Village (Los Angeles)\nGenre:-Pop,R&B,rap rock\nLength:-
26:05\nLanguage:-Korean\nLabel:-Big Hit entertainment\nProducer:-
Pdogg,\nHissNoiseArcadesFredGibsonBadMilkMarcusMcCoan"
 Label(motsp,text=info).grid(row=2,column=0)
 discription="Boy With Luv's Korean title, 작은것들을위한시 translates to 'A Poem for the Small Things'.\n...."
 Label(motsp,text=discription).grid(row=1,column=2)
 Label(motsp,text="The list of songs in the album").grid(row=2,column=2)
 list=Listbox(motsp)
 list.insert(1,"Intro:Persona")
 list.insert(2,"Boy with Luv")
 list.insert(3,"Mikrosomos")
 list.insert(4,"Make it Right")
 list.insert(5,"HOME")
 list.insert(6,"Jamais Vu")
 list.insert(7,"Dionysus")
 list.grid(row=3,column=2)
motsp.mainloop()
def review2():
```

```
boss.title("Boss album Review")
boss.minsize(1000,900)
title=Label(boss,text="NCT U(Neo Culture Tecnology united)").grid()
cover=PhotoImage(file="boss.gif")
Label(boss,image=cover).grid(row=1,column=0)
info="NCT U (엔씨티유) is the first sub-unit of the boy group NCT. \nNCT U doesn't have fix members, the lineup keep changing for every comeback. \nNCT U debuted on  April 9th, 2016 under SM Entertainment."
Label(boss,text=info).grid(row=2,column=0)
discription="With an overall focus on the group's rappers and the seven members' stomping,\n ...."
Label(boss,text=discription).grid(row=1,column=2)
Label(boss,text="The list of songs in album").grid(row=2,column=2)
list=Listbox()
list.insert(1,"Boss")
list.insert(2,"Limitless")
list.insert(3,"The 7th sense")
list.insert(4,"Without you")
list.grid(row=3,column=2)
boss.mainloop()
def review3():
tempo=Tk()
tempo.config(bg="red")
tempo.title("Dont mess up my tempo album Review")
tempo.minsize(1000,900)
title=Label(tempo,text="EXO").grid()
cover=PhotoImage(file="tempo.gif")
Label(tempo,image=cover).grid(row=1,column=0)
info="Released:-November 2, 2018\nStudio:-SM Studios, Seoul, South Korea\nGe"
Label(tempo,text=info).grid(row=2,column=0)
```

```
discription="'Sign' is an electropop track with an intense bass and 'bombastic' \ngroove with lyrics..."
Label(tempo,text=discription).grid(row=1,column=2)
Label(tempo,text="The list of songs in album").grid(row=2,column=2)
list=Listbox(tempo)
list.insert(1,"Tempo")
list.insert(2,"Sign")
list.insert(3,"Ooh La La La")
list.insert(4,"Gravity")
list.insert(5,"24/7")
list.insert(6,"Damage")
list.insert(7,"Smile On My Face")
list.insert(8,"Tempo(Chinese version)")
list.grid(row=3,column=2)
tempo.mainloop()
def review4():
mono=Tk()
mono.config()
mono.title("Mono Mixtape Review")
mono.minsize(1000,900)
title=Label(mono,text="RM(Kim Namjoon)").grid()
cover=PhotoImage(file="mono.gif")
Label(mono,image=cover).grid(row=1,column=0)
info="Released:-October 23, 2018\nGenre:-Hip hop\nLength:-24:47\nLanguage:-Korean,English\nLabel:-Big Hit\nProducer:-RM,Pdogg,Hiss Noise"
Label(mono,text=info).grid(row=2,column=0)
discription="'Forever Rain' opens with a feeling of weariness and loneliness,\n w"
Label(mono,text=discription).grid(row=1,column=2)
Label(mono,text="The list of songs in album").grid(row=2,column=2)
list=Listbox(mono)
```

```
list.insert(1,"Tokyo")

list.insert(2,"Seoul")

list.insert(3,"Moonchild")

list.insert(4,"Badbye")

list.insert(5,"Uhgood")

list.insert(6,"Everythingoes")

list.insert(7,"Forever Rain")

list.grid(row=3,column=2)

mono.mainloop()

defhome_page():

main=Tk()

main.config(bg="pink")

main.title("Home Page")

main.minsize(1300,1000)


title=Label(main,text="Your music destination").grid(row=0,column=2,ipadx=50,ipady=20)

img=PhotoImage(file="logo1.gif")

logo=img.subsample(6,6)

Label(main,image=logo).grid(row=0,column=0)

Label(main,text="Here you can look for the review on the music you listen
to.").grid(row=1,column=2,ipadx=50,ipady=20)

Label(main,text="Latest releases").grid(row=2,column=0)


home=Frame(main,bg="orange")

home.grid(row=3,column=2,sticky="sw")

a1=PhotoImage(file="motsp.gif")

Label(home,image=a1).grid(row=3,column=0,ipadx=5,ipady=10,sticky="e")

Label(home,text="Map of the soul:Persona").grid(row=4,column=0)

Button(home,text="Review",command=review1).grid(row=5,column=0)
```

```
  a2=PhotoImage(file="boss.gif")
Label(home,image=a2).grid(row=3,column=1,ipadx=5,ipady=10)
Label(home,text="Boss").grid(row=4,column=1)
Button(home,text="Review",command=review2).grid(row=5,column=1)

  a3=PhotoImage(file="tempo.gif")
Label(home,image=a3).grid(row=3,column=2,ipadx=5,ipady=10)
Label(home,text="Dont mess up my tempo").grid(row=4,column=2)
Button(home,text="Review",command=review3).grid(row=5,column=2)

  a4=PhotoImage(file="mono.gif")
Label(home,image=a4).grid(row=3,column=3,ipadx=5,ipady=10)
Label(home,text="Mono").grid(row=4,column=3)
Button(home,text="Review",command=review4).grid(row=5,column=3)
main.mainloop()
home_page()
```

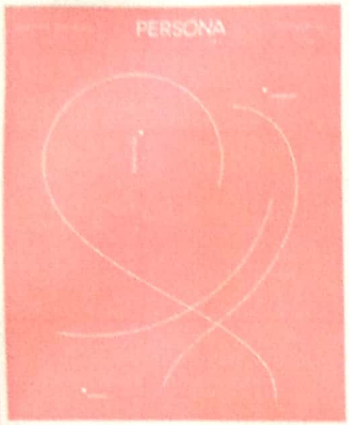
Home Page

#MUSIC

Your music destination

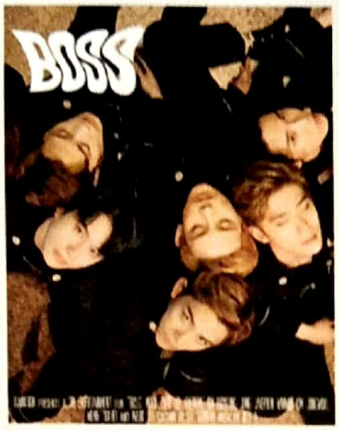
Here you can look for the review on the music you listen to.

Latest releases



Map of the soul:Persona

Review



Boss

Review

Dont mess up my tempo

Review

Mono

Review

Map of the soulPersona album Review

BTS

Released:-April 12, 2019
Studio:-Big Hit (Seoul), The Village (Los Angeles)
Genre:-Pop,R&B,rap rock
Length:-26:05
Language:-Korean
Label:-Big Hit entertainment
Producer:-Pdogg,
Hiss NoiseArcadesFred GibsonBad MilkMarcus McCoan

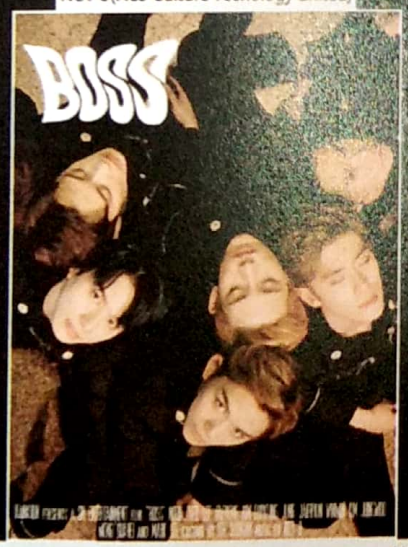
Boy With Luv's Korean title, 작은 것들을 위한 시 translates to 'A Poem for the Small Things'.
Its title suggests a follow-up to their intense 2014 teen anthem 'Boy In Luv,'
about the confusion of young love that seems out of reach.
Now, five years wiser, BTS acknowledge the love that they already possess:
sweet-voiced Jimin sees it in the simplest gesture of asking about your day
while rapper J-Hope sees it as knowing that while he can't be a superhero
to everyone, he can do his best to protect those he cares about.

The list of songs in the album

Intro:Persona
Boy with Luv
Mikrosomos
Make it Right
HOME
Jamais Vu
Dionysus

Boss album Review

NCT U(Neo Culture Tecnology united)



NCT U (엔시티 유) is the first sub-unit of the boy group NCT.
NCT U doesn't have fix members, the lineup keep changing for every comeback.
NCT U debuted on April 9th, 2016 under SM Entertainment.

With an overall focus on the group's rappers and the seven members' stomping,
in-your-face dance moves, the hostile song culminates in a fight between the younger members
(Jungwoo, Mark, and Lucas) and the older ones
(Taeyong, Jaehyun, Doyoung, and Winwin), and shows Mark and Taeyong actively fighting
one another atop of a table. Throughout the video,
the members alternate between stylishly preppy, logo-bearing athleisure
outfits and black uniforms, a nod to the figurative battle they are fighting.
Upon its release, "Boss" trended worldwide on Twitter with a variety of
hashtags as fans heralded NCT U's return as a group and
the formal start of NCT 2018.

The list of songs in album

Boss
Limitless
The 7th sense
Without you

Dont mess up my tempo album Review

EXO

Released:-November 2, 2018
Studio:-SM Studios, Seoul, South Korea
Genre:-K-pop,DanceR&B
Length:-39:10
Language:-Korean,Mandarin
Label:-SMRIVER
Producer:-Lee Soo-man

'Sign' is an electropop track with an intense bass and 'bombastic'
groove with lyrics expressing
a man's growing doubt due to lies told by the woman he loves.
'Ooh La La La' is a Latin pop song,
combining a heavy 808 bass and an 'exotic' atmosphere of Spanish guitar.
The lyrics of the song are about imagined love when two people accidentally make eye contact.
'Gravity' is an electropop track with a retro sound and funky rhythm, which the British duo LDN Noise worked on.
The lyrics were co-written by member Chanyeol, and are described as about the
desire to capture a girl's attention with gravity and charm.

The list of songs in album

Tempo
Sign
Ooh La La La
Gravity
24/7
Damage
Smile On My Face
Tempo(Chinese versio

Mono Mixtape Review

RM(Kim Namjoon)

Released:-October 23, 2018
Genre:-Hip hop
Length:-24:47
Language:-Korean,English
Label:-Big Hit
Producer:-RM,Pdogg,Hiss Noise

'Forever Rain' opens with a feeling of weariness and loneliness,
with conscientious lyrics such as
'I wish it rains all day
Cuz then people wouldn't stare at me
Cuz the umbrella would cover the sad face sad face
Cuz in the rain people are busy minding themselves.'
It is an animated video directed by Choi Jaehoon, who also served as creative director.
The animation supervisors were Lee Jinhee and PD Gim Boseong,
while editing was completed by Lee Jonhoon.
The animators for the music video were Gim Boseong, Kim Bom, Kim Jihwan, Shin Dohyun, Ahn Jaeyoon, Yoon Jongha, Yoon Juri, Lee Jinhee, Jeon Soeun.
Jeong dawoon, Jeong Jimin, and Hyun Yujeong.
The song 'Seoul' is a lyric video, showing various famous places located in Seoul,
such as the Han River, Cheonggyecheon stream, and Seonyudo Island.
It features a brief shot of RM sitting in a car outside a convenience store
and includes a short phone conversation where he describes his plans to leave.
'Seoul' was directed by Choi Yongseok of Lumpens with assistant directors Guzza,
Park Hyejeong and Jeong Minje, also from Lumpens.
Nam Hyunwoo of GDW acted as the director of photography and Min Joonki of Sunshine Underground is credited with composition.

The list of songs in album

Tokyo
Seoul
Moonchild
Badbye
Uhgood
Everythingoes
Forever Rain

PROJECT 2

DATABASE CONNECTIVITY

```
from tkinter import *

import os,sqlite3

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

cur=conn.cursor()

cur.execute('create table if not exists album_info(albumNo int PRIMARY KEY,a_name text,artist_name text,Release_date date,genre text,length int,language text,label text,producer text,sales text)')

cur.execute('insert into album_info values(1,"Map of the soul:persona ","BTS","12-08-2019","K-pop","26:05","Korean ","BigHit Entertainment ","Hiss Noise","Two million")')

cur.execute('insert into album_info values(2,"Boss","NCT U","27-02-2018","K-pop","22:00","Korean","SM Entertainment","Taeyong","One Million")')

cur.execute('insert into album_info values(3,"Dont Mess Up My Tempo","EXO","02-11-2018","Dance.R&B.K-pop","39:10","Korean.Manderin","SM Entertainment","Lee Soman","One million copies")')

cur.execute('insert into album_info values(4,"Mono","RM","23-10-2018","Hip-Hop","24:47","Korean.English","BigHit Entertainment","RM.PDogg","One Million")')

cur.execute('insert into album_info values(5,"Call My Name","GOT7","4-11-2019","K-pop","21:00","Korean","JYP Entertainment","JB.Mark","One Million")')

cur.execute('insert into album_info values(6,"Shangri-La","VIXX","15-05-2017","K-pop","23:11","Korean","Jellyfish Entertainment ","Ravi ","Ninety Lakh")')

cur.execute('insert into album_info values(7,"Cherry Bomb","NCT 127","14-06-2017","EDM.Hip-Hop.Electropop","24:38","Korean","SM Entertainment","Lee Sooman","One Lakh")')

cur.execute('insert into album_info values(8,"We Boom ","NCT Dream ","26-07-2019 ","K-Pop ","21:22 ","Korean ","SM Entertainment ","Jaemin ","Five lakh")')

cur.execute('insert into album_info values(9,"The Dream Chapter:Magic ","TXT","21-11-2019","K-pop.Tropical House.R&B ","27:43","Korean ","BigHit Entertainment ","HitMan Bang ","One Lakh Twentyfive Thousand")')

cur.execute('insert into album_info values(10,"Cle:LEVANTER","Stray Kids ","09-12-2019","K-pop","21:34 ","Korean ","JYP Entertaainment ","Bang Chan ","Two lakh copies")')


root=Tk()

root.config(bg="pink")
```

```
root.title("Album Database")
list=Listbox(root)
list.grid(ipadx=200,ipady=50)
list.insert(1,"1.Map of the Soul:Persona")
    list.insert(2,"2.Boss")
    list.insert(3,"3.Dont Mess Up My Tempo")
    list.insert(4,"4.Mono")
    list.insert(5,"5.Call My Name")
    list.insert(6,"6.Shangri-La")
    list.insert(7,"7.Cherry Bomb")
    list.insert(8,"8.We Boom")
    list.insert(9,"9.The Dream Chapter:Magic")
    list.insert(10,"10.Cle:LEVANTER")
    Label(root,text="Enter the number for the Album You want to see Info about. ").grid()
    num=StringVar()
    e1=Entry(root,textvariable=num).grid()
    def search():
      cur.execute('select * from album_info where albumNo=?',(num.get()))
      row=cur.fetchall()
      for i in row:
        list1.insert(END,i)
Button(root,text="SUBMIT",command=search).grid()
list1=Listbox(root)
list1.insert(1,"Album_name  Artist_Name  Release_Date  Genre Length Language Label Producer Sales")
list1.grid(ipadx=500,ipady=100)
root.mainloop()
```