

objective:- Demonstrate the use of different file accessing modes, different attribute read method.

Step 1:- Create a file object using open method and use the write access mode followed by writing some contents onto the file and then closing the file

Step 2:- Now open the file in read mode and than use read() read line() and store the output in variable and finally display the contents of variable

Step 3:- Now use the file object for finding the name of the file, the file mode in which it is opened whether the file is still open or close and finally the output of the soft space attribute

Step 4:- Now open the file obj. in write mode, write some another content close subsequently in 'wt' mode that is the update mode and write contents

Step 5 :- Open the file in append access mode  
Write the contents to be appended  
in the file by using write() method.  
Close the file using close() method.  
Now open the file in read mode and read the file  
and display the contents of the file.

a) Write a program using the file operation  
for reading the individual elements  
from the file using the cond<sup>n</sup> statement  
and inserting some character with the  
content of the file.

Algorithm:-

Step 1:- Open the file in read mode.  
Read the contents of file and store  
it in a variable.

Step 2:- Use the while cond<sup>n</sup> statement till the  
length of variable is greater than  
zero and print the character with  
the content of file.

D) Write a program for finding the length of the different lines that exist within the given file.

Algorithm :-

Step 1 :- Open the created file in the read mode.

Step 2 :- Use the readlines() method and store it in the variable

Step 3 :- Use the for cond<sup>n</sup> state and print the length of each line in the file

last

Or

Code/Out

```
1) file obj = open("RitikV.txt", "w")
   file obj.write("My name is shubhang singh. I am 18 yrs
   old. I study in TCS C.")
   file obj.close()

2) file obj = open("RitikV.txt", "r")
   r1 = fileobj.read(16)
   fileobj.close()
   file obj = open("RitikV.txt", "r")
   r2 = fileobj.readline()
   file obj.close()
   fileobj = open("RitikV.txt", "r")
   r3 = fileobj.readlines()
   fileobj.close()

print("The output of read() method upto 16
      characters is:", r1)
print("The output of readline() method is:", r2)
print("The output of readlines() method is:", r3)
```

>>> The output of read() method upto 16 characters is:

My name is Ritik.

The output of readline() method is : my name is  
Ritik Viswakarma.

The output of readline() method is : [ 'my name is  
Ritik viswakarma.'

I am 18 yrs. old. In, I study in Thakur TCS C

3) a = fileobj.name  
b = fileobj.closed  
c = fileobj.mode  
print("The output of name attribute is: ", a)  
<sup>20</sup>  
print("The output of closed attribute is: ", b)  
print("The output of mode is: ", c)

""  
The output of name attribute is: RitiRv.txt.  
The output of closed attribute is: True  
The output of mode attribute is: s

4) with open ("RitiRv.txt", "r") as fileobj:  
fileobj.read(10)  
pos = fileobj.tell()  
print("The current position of pointer in file is  
:", pos)  
fileobj = open ("RitiRv.txt", "r")  
seek = fileobj.seek(20, 0)  
print ("The output of seek() method is: ", seek)  
fileobj.close()

""  
The current position of pointer in file is: 10  
The output of seek() method is: 20

a) with `open("RitirV.txt", "r")` as `g`:

```
    s = 1
    c = g.read(s)
    while len(c) > 0:
        print(c, end=" ")
        c = g.read(s)
```

>>>

M Y K R A M E K I K S K R K I K T K I K V K I  
K H K W K A K K A K T Y K M K A K.

b) with `open("RitirV.txt", "r")` as `fileobj`:

```
r1 = fileobj.readlines()
print("\n".join(r1))
for line in r1:
    print(len(line))
```

>>>

31

26

50

## Practical No. 2

Aim :- To study the use of iterator and iterables.

1) Simple use of iter() method:-

→ Create a tuple object and assign it the value or elements. Then create a iterable object to iter through the tuple with iter() method. Now, use for cond<sup>n</sup> statement to display all the elements in the iterable object.

2) Program using the iterable method for displaying the set of odd numbers:-

Step 1 :- Define a class and within that the iter() which will initialize the first element within the container object.

Step 2 :- Now use the next() and define the logic for collecting the odd values.

Step 3 :- Define an object for the class and iter through the object

Step 4 :- Now display the values using for cond<sup>n</sup> state.

- 3) Program using the iterable object to display set of first 20 numbers.
- Step 1:- Define iter() with an argument and initialize it to the first value.
- Step 2:- For extracting the next element from the container, use the next() with an argument, and compare the no. of elements required in a container by using the cond' statement.
- Step 3:- Now create an object from the given class and pass this object as an argument to the iter() method.
- Step 4:- Now using the cond' state display all the values from the given container.

- q) Program for printing the square and cube of given set of no. using map().
- Step 1:- Define a square fn with a argument and return the square of the no.
- Step 2:- Define a fn cube with arguments and return the square of the no.
- Step 3:- Declare a list variable and call the fn square and cube in the list.
- Step 4:- Use for cond<sup>n</sup> statement and use the map() to find the square and cube given set of no.

code/output

1) mytuple1 = ("Banana", "Orange", "Apple")  
myiter1 = iter(mytuple1)  
for i in myiter1:  
 print(i)

? )  
Banana  
Orange  
Apple

2) class odd:

```
def __iter__(self):  
    self.num = 1  
    return self  
  
def __next__(self):  
    if self.num <= 25:  
        num = self.num  
        self.num += 2  
        return num  
  
    else:  
        raise StopIteration
```

```
obj = odd()  
myiter = iter(obj)  
for i in myiter:  
    print(i)
```

???

3  
5  
7  
9  
11  
13  
15

v7  
19  
21  
23  
25

Ques/Output:-

3) class ~~ss~~ myclass:

    def \_\_iter\_\_(self):

        self.num = 1

        return self

    def \_\_next\_\_(self):

        if self.num <= 20

            num = self.num

            self.num += 1

            return num

        else:

            raise StopIteration

obj = myclass()

myiter = iter(obj)

for i in myiter:

    print(i)

??>

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

```
1) def square(x):  
    return (x*x)  
def cube(x):  
    return (x*x*x)  
func1 = [square, cube]  
for i in range(5):  
    valueout = list(map(lambda x: x(i), func1))  
    print(valueout)  
))  
[1,1]  
[4,8]  
[9,27]  
[16,64]
```

~~lambda~~  
~~or?~~

## Practical No. 3

Aim :- Program to demonstrate exception handling.

1) Program for demonstrating the use of I/O Error.

Step 1:- Use the try block to define the normal course of action.  
For eg:- Define the file obj. and open the file in the write or read mode and write some content onto the file.

Step 2:- Use the except block with the I/O error as an environment error and convey the appropriate message to the user, else display the msg. that the operation is carried out successfully.

② Program to demonstrate the multiple exception viz. IOError & ValueError.

Step 1:- Use the try block and defines the file object and open the file in write or read mode and write some content onto the file.

- Step 2:- Also, accept the value from the user and if it is a valid value display the entered value and terminate the cond'n by using the break statement.
- Step 3:- Define the except blocks for IOError and ValueError.

Q3 Program to demonstrate the use of Index error.

- Step 1:- Use the try block and define a list with some element in it.
- Step 2:- Use the indexing of list and print the element of that list.
- Step 3:- Now define a list which is empty. Try to print the element of list using indexing.
- Step 4:- Use the except block with the index error keyword and print the appropriate statement.



Ques:-

1) Try:-

```
fileobj=open("abc.txt","w")
fileobj.write("Python is a interpreted language in
Python is a indented language")
except IOError:
    print("There is an environment error")
else:
    print("operation successfully")
```

??> Operation successful.

2) while True:-

try:

```
fileobj=open("abc.txt","w")
fileobj.write("Python is a interpreted language")
except IOError:
    print("Python is an interpreted language")
a=int(input("Enter a number:"))
print(a)
break
```

except IOError:

print("There is a envi. error")

except ValueError:

print("The value is invalid")

??> Enter a number: abc  
The value is invalid  
Enter a number: pqr

The value is invalid  
Enter a no: B

## code/output

26

3) try:

```
list1 = [4, 19, 26, 73, 33]
e1 = list1[2]
print(e1)
list2 = []
e2 = list2[1]
print(e2)
except IndexError:
    print("The index is out of range")
```

>>>

26

The index is out of range.

✓  
Jin

Aim: To segregate alphabetical value from numerical value in the given string.

Algorithm:

Step 1 :- Import the regular expression library

Step 2 :- Initialize a variable which takes the user input that contains name and roll no. of student.

Step 3 :- Define the pattern as per the requirement of program

Step 4 :- Use the.findall method & print the output.

Aim :- To extract consecutive two character boundaries at word boundaries

Algorithm:

Step 1: Import the regular expression library

Step 2: Take a sentence from the user and assign it to variable.

Step 3: Define a pattern that finds the consecutive two character boundaries at word boundaries

Step 4: - use the find all method and print & output.

Aim:- To find if a word is present in a given string

Algorithm:

Step 1: Import the regular expression library.

Step 2: Define a variable which takes the a word to be find. sentence from the user

Step 3: Define another variable which takes the word to be find

Step 4: Use the find method to generate the output

Step 5: Use the if cond<sup>n</sup> statement and display the msg accordingly.

es

Aim: To find a word starting from vowel

Algorithm:

Step 1: Import the regular expression library

Step 2: Take a few lines as input from the user.

Step 3: Define a pattern which finds the vowels in the string

Step 4: Use the find all word starting to find the with vowel

Step 5: Print the output.

Aim :- To remove whitespace in a given string

Algorithm

Step 1 : Import regular expression library.

Step 2 : Take the string input from user which contains whitespace.

Step 3 : Define the pattern as per the requirement.

Step 4 : Assign a variable with empty string.

Step 5 : Use the sub method to substitute the whitespace with string and print the output.

Aim :- To find numbers starting with either 8 or 9 & with length 10 from a list.

### Algorithm

Step 1 : Import regular expression library

Step 2 : Define a list variable which contains the input numbers.

Step 3 : Define the pattern required to extract number satisfy the cond.

Step 4 : Use the for loop follower by the if cond statement & print the message accordingly.

Aim : To segregate the hostname and domain name from the email address.

### Algorithm

Step 1: Import the regular expression library

Step 2: Assign a variable which contains the email address in list datatype

Step 3: Define a pattern required to split the two parts.

Step 4: Use the for loop followed by re.split method to segregate the host name & domain name & print the output.

Jyoti

Program :-

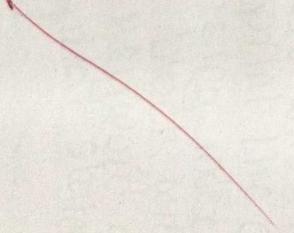
```
import re  
sequence = input("Enter name & roll no. of student  
without space:")
```

```
pattern = r'\d+'
```

```
output = re.findall(pattern, sequence)  
print(output)
```

Output :

Enter name & roll no. of students without space  
Neeraj 17w Sachin 17w Aditya 17w  
[\'17w', \'17w', \'17w']



Program:

28

```
import re  
sentence = input("Enter a sentence: ")  
pattern = r'\b\w{2,3}\b|\w{2,3}\b'  
output = re.findall(pattern, sentence)  
print(output)
```

Output:

Enter a sentence: Hello there, my name is Neeraj.

[ 'He', 'lo', 'th', 're', 'my', 'na', 'me', 'is', 'ne' ]

Ans

Programs

```
import re
x = input("Enter a sentence: ")
y = input ("Enter a word to be found from string: ")
output = re.findall(y,x)
if output:
    print("match found!")
else:
    print("Match not found!")
```

Output:

1) Enter a sentence: My name is neeraj

Enter a word to be found from the string: name

match found!

Program:-

30

```
import re  
string = input("Enter few line :")  
output = re.findall(r'\b[a,e,i,o,u]\w+\bAEIOU  
                    \w+', string)  
print(output)
```

OUTPUT

Enter few lines:- My name is Neeraj  
['is']

Ans

Q8

## Program

```
import re
String = input("Enter few names:")
pattern = re'1st'
replace = ' '
output = re.sub(pattern, replace, string)
print(output)
```

## Output:

Enter few names: Veeraj Ray Pranay  
Veeraj Ray Pranay.

```
import re
```

```
list = ['8767522106', '9821471119', '7024523410',  
        '59829047817']
```

```
pattern = r'\b[8-9]\d{9}\b'
```

```
for i in list:
```

```
    if re.match(pattern, i) and len(i) == 10:
```

```
        print("Number satisfies the cond")
```

```
    else:
```

```
        print("Number does not satisfies the cond")
```

## Output

Number satisfies the cond

Number satisfies the cond

Number doesn't satisfies the cond

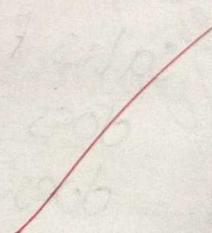
Number doesn't satisfies the cond

Program

```
import re  
email = ['xyz@tcsc.edu', 'abc@gmail.com', 'JK1@hotmail.com']  
pattern = r'[@]'  
for i in email:  
    output = re.split(pattern, i)  
    print(output)
```

Output:

```
[['xyz', '@', 'tcsc.edu']]  
[['abc', '@', 'gmail.com']]  
[['JK1', '@', 'hotmail.com']]
```



## Practical - 5

Aim : To make use of GUI application with the basic method

Algorithm:

Step 1: Use the tkinter library for import the features of text widget.

Step 2: Create a window from a text variable & position it into the text widget.

Step 3: Use the pack() along with the obj created from text method & use the parameter

Step 4: Use the mainloop method

Step 5: Now repeat above step with a label method takes the following argument

- i) Name of parent window
- ii) Background color (bg)
- iii) Foreground color (fg)

iv) fg

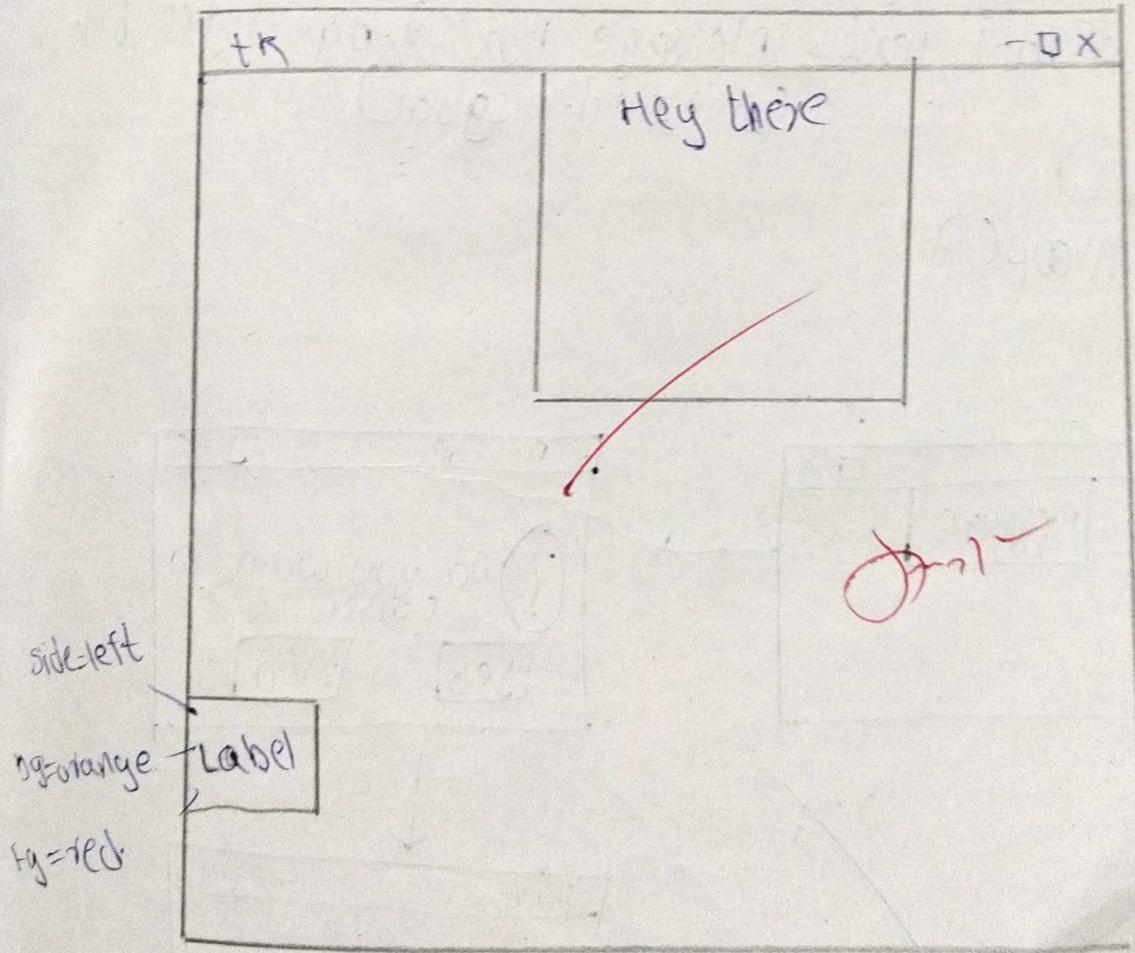
Now use pack with relevant attr.

## Program

```
from tkinter import *
root = Tk()
T1 = Text(root)
T1.insert(END, "Hey there.")
T1.pack(side=TOP, padx=20, pady=30, ipadx=40, ipady=50)
L1 = Label(root, text="Label"), bg="orange", fg="red"
L1.pack(side=LEFT, padx=10, ipadx=20, ipady=30)
root.mainloop()
```

31

## Output



D Aim :- To make use messagebox method of the GUI application.

Algorithm :

Step 1 : Import relevant method from tkinter library.

Step 2 : Define a function and use messagebox along with different methods available which contains one or more arguments.

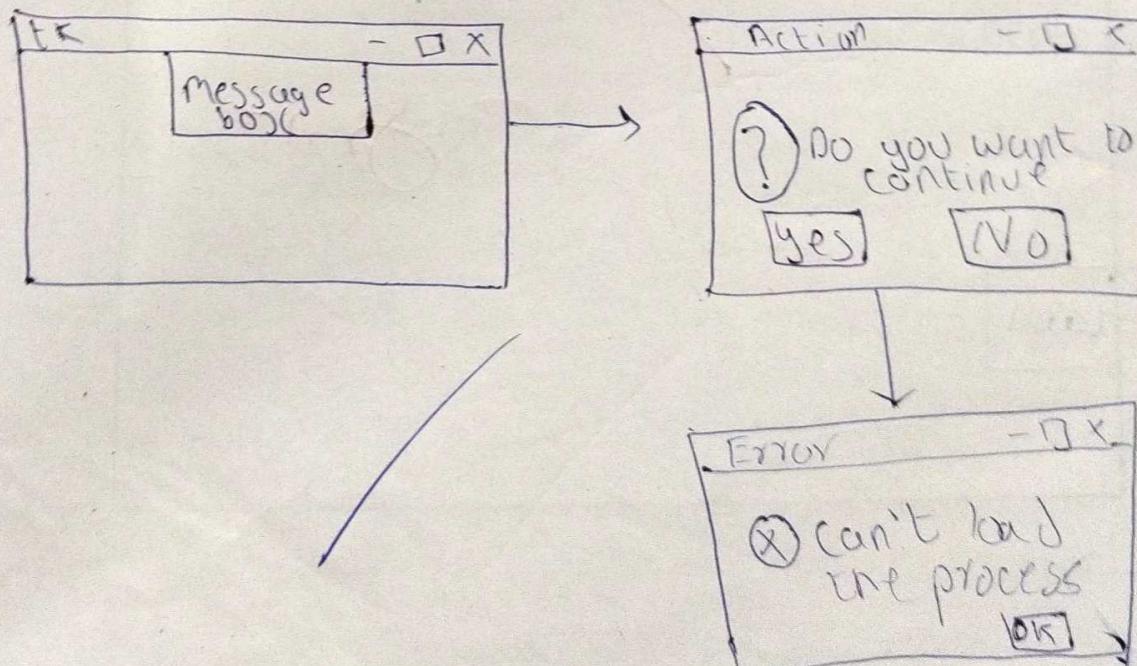
Step 3 : Create an object from button method and place it onto the parent window with text & command attribute

Step 4 : Use pack method and finally use mainloop method.

## Program

```
from tkinter import *
from tkinter import messagebox
def msg_box():
    messagebox.showaskyesno("Action", "Do you want to continue?")
    messagebox.showerror("Error", "Can't load the process!")
root = Tk()
root.config(bg="gray")
B1 = Button(root, text="Message Box", bg="light blue", command=msg_box)
B1.pack()
root.mainloop()
```

Output



### Practical-6

**Aim :-** Program to traverse various windows using the button widget.

**Algorithm:-**

**Step 1:-** Import relevant method from tkinter library

**Step 2:-** Define a button tk widget which will be placed on the current window to traverse and define another button which will be used to exit from the window & place it onto current window

**Step 3 :-** Define another fn. which will use the quit method to terminate program

**Step 4 :-** Now create an obj. of main window and use diff methods

**Step 5 :-** Define two buttons which will be placed on the main window; one to traverse another window and other to terminate.

**Step 6 :-** Define another fn. which will create various button placed on third window

Step 7:- Define two button respectively.

Step 8:- Call the Mainloop function.

Final

```

from tkinter import *
def main():
    root = Tk()
    root.geometry("200x500")
    root.config(bg="green")
    root.title("Window 1")
    B1=Button(root, text="Next", command=main1)
    B1.grid(ipadx=50, ipady=40, padx=20, pady=30)
    B2=Button(root, text="Exit", command=term)
    B2.grid(ipadx=50, ipady=40, padx=20, pady=30)
def term():
    quit()
tos=Tk()
tos.geometry("450x300")
tos.config(bg="purple")
tos.title(tos, text="continue", command=main) (main window)
B3=Button(tos, text="continue", command=main)
B3.grid(ipadx=50, ipady=40, padx=20, pady=30)
B3.grid(ipadx=50, ipady=40, padx=20, pady=30)
B2=Button(tos, text="Exit", command=term)
B2.grid(ipadx=50, ipady=40, padx=20, pady=30)
def main1():
    top=Tk()
    top.geometry("450x300")
    top.config(bg="purple")
    top.title("Window 2")

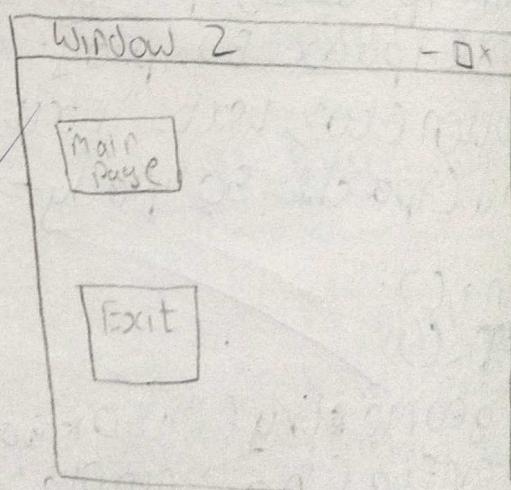
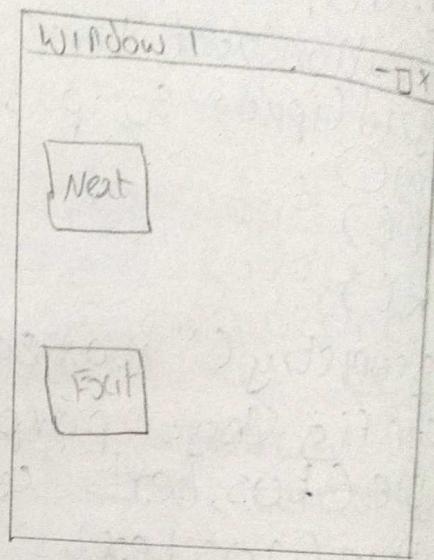
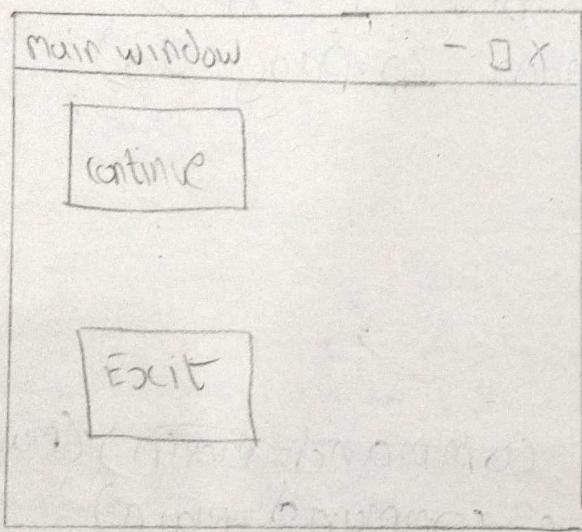
```

88

B4 = Button(tup, text="main page", command=main)  
B4.grid(ipadx=50, ipady=40, padx=20, pady=30)  
B2 = Button(tup, text="Exit", command=term)  
B2.grid(ipadx=50, ipady=40, padx=20, pady=30)

mainloop()

Output



step

Step

Practical - ?

Aim :- Use spinbox method

Algorithm

Step 1 : Create an obj. from tk method  
and subsequently create an obj.  
from the spin box method

Step 2 :- Make the obj. so created onto  
the parent window and trigger  
the corresponding events

Step 3 :- Use the pack method to provide  
the direction using anchor method

Step 4 :- Use the mainloop method to  
terminate.

Program  
from  
root =  
L = La  
L.pack  
b1 = S  
b1.pack  
root  
output

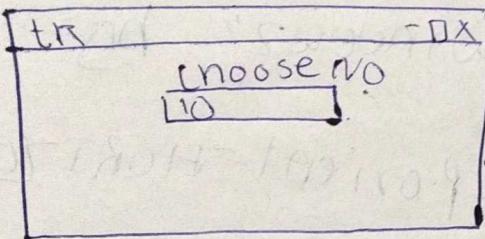
program

```
from tkinter import *
root = Tk()
L=Label(root, text="Choose no", bg="red")
L.pack()
S1=Spinbox(root, from_=-10, to=10, font=22)
S1.pack(side="top")
root.mainloop()
```

38

Program

Output



In 12m

## Practical 8

Algorithm

Aim:- Paned Window

Step 1:- Create an obj from paned window and use the pack method with the attribute fill and expand.

Step 2:- Create an object from the label method and put it onto the paned window with the text attribute and use add method to embed the new object.

Step 3:- Similarly create a second paned window obj. and add it onto the 1<sup>st</sup> paned window with orientation specified.

Step 4:- Now create another label obj. and place it onto the 2<sup>nd</sup> paned window obj. and add it onto the 2<sup>nd</sup> pane.

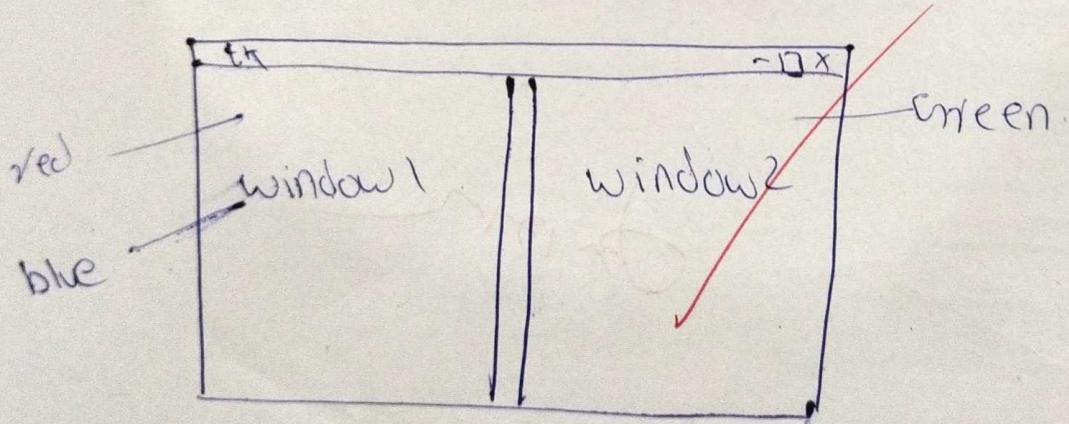
Steps:- Use mainloop method.

## Program

88

```
from tkinter import *
root = Tk()
P = PanedWindow()
P.pack(Fill=BOTH, expand=1)
L = Label(P, text="window1", bg="red", fg="blue")
P.add(L)
P1 = PanedWindow(P, orient=VERTICAL)
P.add(P1)
L1 = Label(P1, text="window2", bg="green")
P1.add(L1)
P2 = PanedWindow(P, orient=HORIZONTAL)
P.add(P2)
mainloop()
```

Output:



88

Practical-9

Aim:- Use of Canvas method

Algorithm,

Step 1:- Use the tkinter method and create an obj. from canvas method and has attribute.

Step 2:- Use the method create oval, create line and arc along with the canvas obj. so. created

Step 3:- Now call the pack method and Mainloop method

from  
root =  
c1 = Ca

line =  
oval =  
arc =

outpu

Program

40

```
from tkinter import *
```

```
root = Tk()
```

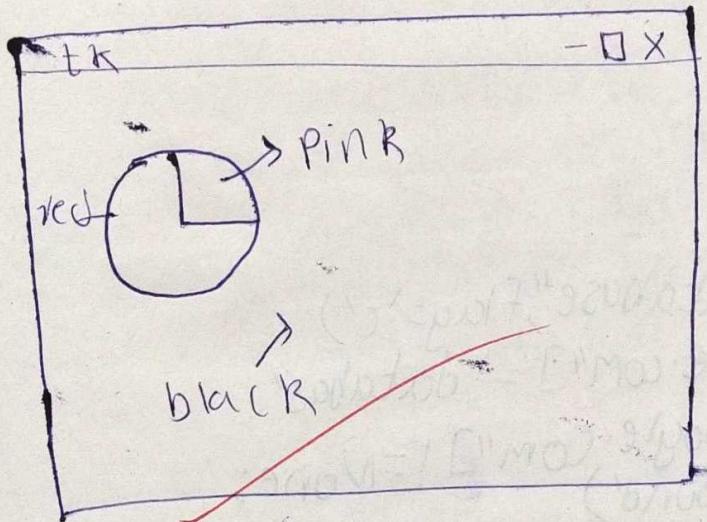
```
c1 = Canvas(root, height="300", width="400",  
            bg="black")
```

```
line = c1.create_line(10, 90, 150, 250, fill="black")
```

```
oval = c1.create_oval(10, 90, 170, 250, fill="red")
```

```
arc = c1.create_arc(10, 90, 150, 250, fill="pink")
```

Output



Jan 12

Practical - 10  
Aim:- To make use of database library.

Step 1: Import db library and use the open method for creating the database along with the corresponding flag

Step 2: Use the objects for accessing to given web size and the corresponding regular for web page size.

Step 3: Check whether the given URL address with the regular of the pages is not equal to none than displaying the msg from URL address else not found.

```
if URL == "www" -> print("Good")
```

```
else: print("Bad")
```

Output:

Good

```
import dbm
```

```
db=dbm.open("database",flag='c')
```

```
db["http://google.com"] = "database"
```

```
if db["http:// google.com"] != None:
```

```
    print(" URL Found")
```

```
else:
```

```
    print(" URL not found")
```

```
db.close()
```

Output:

URL Found

## Practical 11

Aim:- Database Connectivity

Step 1 :- Import the corresponding library for making the database connection which are "os", "sqlite3".

Step 2 :- Now create the connection object using sqlite3 library and the connect method for creating the new db.

Step 3 :- Now create cursor obj. using cursor method from the connection obj. created above.

Step 4 :- Now use the execute method for creating a table with the column name and datatype

Step 5 :- Now with cursor obj. use the insert statement.

Step 6 :- Use the commit method for completing the transaction.

```
import dbm,os,sqlite3
database=
connection=sqlite3.connect("student.db")
cursor=database.cursor()
cursor=connection.cursor()
cursor.execute('create table stud (Name char, Rollno
                Integer)')
<sqlite3.Cursor object at 0x02ED65E0>
cursor.execute('insert into stud values("shubhang",1776)')
<sqlite3.Cursor object at 0x02ED65F0>
connection.commit()
cursor.execute('Select name from stud where Rollno=1776')
<sqlite3.Cursor object at 0x02ED65E0>
cursor.fetchall()
[('Shubhang',)]
```

✓ Dr. mir

## CIVI - Project

```
import tkinter as tk  
  
from functools import partial  
  
tempVal = "Celsius"  
  
def store_temp(sel_temp):  
    global tempVal  
    tempVal = sel_temp  
  
def call_convert(rlabel1, rlabel2, inputn):  
    tem = inputn.get()  
  
    if tempVal == 'Celsius':  
        f = float((float(tem) * 9 / 5) + 32)  
        k = float((float(tem) + 273.15))  
        rlabel1.config(text="%f Fahrenheit" % f)  
        rlabel2.config(text="%f Kelvin" % k)  
  
    if tempVal == 'Fahrenheit':  
        c = float((float(tem) - 32) * 5 / 9)  
        k = c + 273  
        rlabel1.config(text="%f Celsius" % c)  
        rlabel2.config(text="%f Kelvin" % k)  
  
    if tempVal == 'Kelvin':  
        c = float((float(tem) - 273.15))
```

```
f = float((float(tem) - 273.15) * 1.8000 + 32.00)
rlabel1.config(text="%f Celsius" % c)
rlabe12.config(text="%f Fahrenheit" % f)
return
```

```
root = tk.Tk()
root.geometry('400x150+100+200')
root.title('Temperature Converter')
root.configure(background='#09A3BA')
root.resizable(width=False, height=False)
root.grid_columnconfigure(1, weight=1)
root.grid_rowconfigure(0, weight=1)
```

```
numberInput = tk.StringVar()
```

```
var = tk.StringVar()
```

```
input_label = tk.Label(root, text="Enter temperature", background='#09A3BA', foreground="#FFFFFF")
input_entry = tk.Entry(root, textvariable=numberInput)
input_label.grid(row=1)
input_entry.grid(row=1, column=1)
```

```
result_label1 = tk.Label(root, background='#09A3BA', foreground="#FFFFFF")
result_label1.grid(row=3, columnspan=4)
result_label2 = tk.Label(root, background='#09A3BA', foreground="#FFFFFF")
```

```
result_label2.grid(row=4, columnspan=4)

dropDownList = ["Celsius", "Fahrenheit", "Kelvin"]

dropdown = tk.OptionMenu(root, var, *dropDownList, command=store_temp)
var.set(dropDownList[0])

dropdown.grid(row=1, column=3)

dropdown.config(background="#09A3BA", foreground="#FFFFFF")
dropdown["menu"].config(background="#09A3BA", foreground="#FFFFFF")

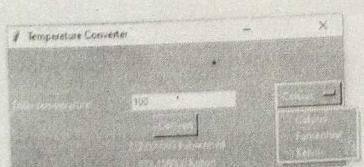
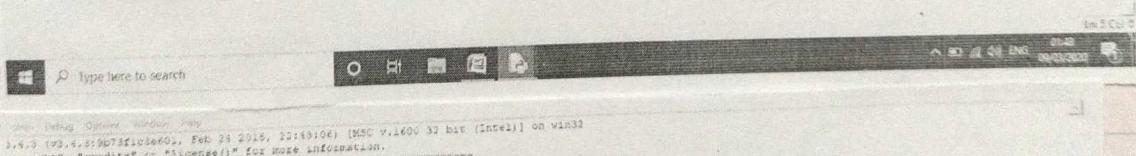
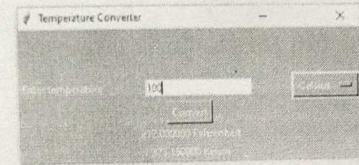
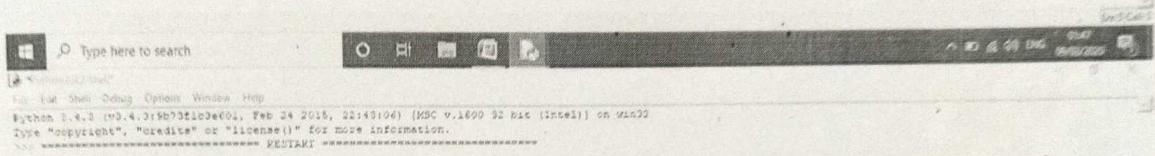
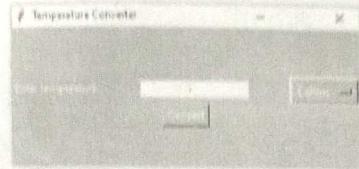
call_convert = partial(call_convert, result_label1, result_label2, numberInput)

result_button = tk.Button(root, text="Convert", command=call_convert, background="#09A3BA",
foreground="#FFFFFF")

result_button.grid(row=2, columnspan=4)

root.mainloop()
```

[1] > python -m http.server 8000  
Python 3.4.3 (v3.4.3:95672efcd63, Feb 18 2015, 22:44:06) [MSC v.1600 32 bit (Intel)]  
Type "copyright", "credits" or "license()" for more information.  
\*\*\*\*\* RESTART \*\*\*\*\*



✓ ✓ ✓



# Database Project

```
>>> import os,sqlite3  
>>> connection = sqlite3.connect("Music.db")  
>>>c1 = connection.cursor()  
c1.execute('create table MusicLibrary  
(Name,Singer,Type  
list)  
<sqlite3.Cursor object at 0x03D20F20>  
>>>c1.execute('insert into MusicLibrary  
("Shape Of You","Ed Sheeran",0)  
<sqlite3.Cursor object at 0x03D20F20>  
>>>c1.execute('insert into MusicLibrary  
("Castle Of Glass","Imagine Dragons",1)  
<sqlite3.Cursor object at 0x03D20F20>  
>>> connection.commit()  
>>>c1.fetchall()  
>>>c1.fetchall()  
[('Shape Of You', 'Ed Sheeran', 0),  
 ('Castle Of Glass', 'Imagine Dragons', 1)]  
>>>
```

