

```
# write method
```

```
file obj = open("abc.txt", "w")
file obj.write("Subject in " + 'hCS')
file obj.write("In " + 'Calculus' + 'in Stats')  
file obj.close()
```

```
# read . method
```

```
file obj = open("abc.txt", "r")
str1 = file obj.read()
print("The output of read method is : " + str1)
file obj.close()
```

```
>>> The outputs of read method is : Subjects in CS  
[  
    Calculus  
    Stats  
    Python
```

```
# readline method
```

```
file obj = open("abc.txt", "r")
str2 = file obj.readline()
print("The output of readline method is : " + str2)
file obj.close()
```

```
>>> The output of readline method is : Subject in CS
```

```
# readlines method
```

```
file obj = open("abc.txt", "r")
str3 = file obj.readlines()
print("The outputs of readlines method is : " + str3)
file obj.close()
```

```
>>> The output of readlines method is : Subject in CS
```

```
[  
    'Subjects in CS\n', 'in Stats\\n', 'Calculus\\n',  
    'Python']
```

19
25/11/19

PRACTICAL - 1

Objective:

Demonstrate the use of different file according modes different attributes and read method.

Algorithm

Step 1: Create a file objects 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 the use read(), readline() & readlines() and store the output in variable & finally display the contents of variable.

Step 3: Now use the file objects for finding the name of file, the file name, the file mode in which it's opened whether the file is still open or close & finally the outputs of the softspace attribute

81

Step 4: Now open the file
in write mode write some
content close, subsequently
open the file object in 'out'
that is the update mode and
contains.

Step 5: Open file object in mode
display the update written
content and close, open again
in rt mode that is with passed
parameter & display the output
subsequently.

Step 6: Now open file in append
mode open write method
write contents close the file
object. Open the file obj
in read mode & display the
append output.

file attributes

```
a = fileobj.name  
b = file obj.closed  
C = file obj.mode  
d = file obj.softspace  
print("Name of file:", a)  
print("closed", b)  
print("file mode", c)  
print("Softspace", d)
```

20

O/P

```
>>> Name of file : abc.txt  
>>> closed : True  
>>> FileMode : r  
>>> Softspace : 0
```

write mode

```
file obj = open("abc.txt", "w")  
file obj = write("C")  
fileobj.close()
```

read modes

```
theobj = open("abc.txt", "r")  
x = file obj.read()  
print("The output of readmode is:", x)
```

```
>>> The output of read mode is : (
```

w+

```
fileobj = open("abc.txt", "wt")  
fileobj = write("Kushal Sir")  
fileobj.close()
```

```
# >+
fileobj = open("abc.txt", "r+")
x1 = fileobj.read(10)
```

point ("The output of r+ is: ~~Kushal~~ " +
"x1")
Step 7

Output: The output of r+ is: Kushal Sir

```
# append method
```

```
fileobj = open("abc.txt", "a")
```

```
fileobj.write("\n FOSS")
```

```
fileobj.close()
```

```
fileobj = open("abc.txt", "r")
```

```
x2 = fileobj.read()
```

point ("The output of append mode
is", x2)

Output: The output of append mode
is FOSS

```
# Tell method
```

```
fileobj = open("abc.txt", "r+")
pos = fileobj.tell()
```

```
point ("Tell", pos)
```

```
fileobj.close()
```

Output: 0

Step 8
mod

and

Step 8

as
o
s

Step 9

a
th

St

Step 7 : Open the file obj.read mode declare a variable & perform file objects dot tell method and store the output.

Step 8 : USE seek method with the arguments with opening the file obj in read mode and closing subsequently,

Step 9 : Open file obj with read mode also use the readlines () method and print the same by counting the length use for conditional statement & display the length.

~~not for all~~

2/12

is

PRACTICAL - 2

: Aim - To display elements of a tuple using iterator method

Algorithm

Step 1: Form a tuple with certain elements inserted in it.

Step 2: Use iter method with tuple and assign it to a variable.

Step 3: Use the next method with variable & point the elements.

Aim: To use iter method with for loop for a

Algorithm: Step 1: Form a tuple with certain elements inserted in it.

Step 2: Use the for conditional statement to access each element of tuple.

Step 3: Point the elements of tuple

Neerja
mytup1

myrec

point

point C

point C

point C

output

Neerja

Raj

Parana

Sachin

Neerja

mytup

for a

output

Neerja

Raj

Parana

Sachin

Program 1:

```
mytuple = ("Neeraj", "Raj", "Paranay", "Sachin")
myiter = iter(mytuple)
print(next(myiter))
print(next(myiter))
print(next(myiter))
print(next(myiter))
```

Output

```
Neeraj
Raj
Paranay
Sachin
```

Program 2:

```
mytuple = ("Neeraj", "Raj", "Paranay", "Sachin")
for a in mytuple
    print(a)
```

Output :

```
Neeraj
Raj
Paranay
Sachin.
```

class

Program

class odd:

```
def __iter__(self):
    self.num = 1
    return self
```

def __next__(self):

```
    num = self.num
    self.num += 2
    return num
```

myobj = odd()

myiter = odd().__iter__(myobj)

• x in myiter:

print(next(myiter))

O/P:

Enter no: 10

1

3

5

7

9

Algorithm

+ Algorithm

Step 1. Define a class within that define an iter method w/ u initialize the first element within the odd b container object -> get odd all .1
add b condition to stand in max

Step 2. Now use next method & # define the logic for displaying the above odd values .01°) and nego
odd value display same direction

class odd :

~~def __init__ =~~

IF this should display ~~for all~~ 211-2

and from mark 100 as an excess

then repeat (1) but just in half

using 1/2 & then out it in excess

and again just don't operation out

the below is aiming go to

why 222012

PRACTICAL - 3

Exception Demonstrating Program

Algorithm

1. Use the try block to defined the normal block of action.
2. Example : Define a file object and open the file in the write mode and write some content onto the file.
3. Use an except block with to errors as an environment error and convey the appropriate message to the user, else display the message that the exception or operation is carried out successfully.

Program

try:-

24

```
to = open("Pinank.txt", 'w')
```

```
to.write("My name is pinank")
```

except IOError:

```
    print("Environment error")
```

else:

```
    print("Successfull")
```

Output :-

Environment Error.

] Program:-

try:-

```
a = int(input("Enter"))
```

except ValueError:

```
    print("Arithmatic error")
```

else:

~~```
 print("Successfull")
```~~

Output :-

Enter: 14

Arithmatic error.

Program to demonstrate the multiple exception & value error

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

Step 2: Also, accept the value from the user and if it is a valid value, display the extend value & terminate the condition by using the break statement.

~~Just~~

25..

## PRACTICAL-4

### Topic - Regular expression.

Step 1: Import re module declare pattern and declare sequence use match method with declare arguments if arguments matched than print the same otherwise print NOT FOUND.

Step 2: Import re module declare pattern with literal and metacharacter declare string value - use the.findall() with argument & print same.

Step 3: Import re-module declare pattern with metacharacter with split() & print output.

Step 4: Import re module declare & accordingly declare pattern replace with no space use re.sub() with 3 arguments & print the string without spaces.

Step 5: Import re module declare a sequence use search method for finding subsequently use the group() with show up the matched string.

```

>>> # no-space : 123 864 78945 67890 09876543210
 import re
 String = 'abc def ghi'
 Pattern = '8' / 26
 replace = " "
 U1 = re.sub(Pattern, replace, String)
 print(U1)
 >>> abc def ghi
group()
 import re
 sequence = 'Python is an interesting language'
 V = re.search('Python', sequence)
 print(V)
 V1 = V.group(0)
 print(V1)
 >>> Python
verifying the given set of phone numbers - 2
 import re
 list1 = ['8004567891', '9145673210', '7685429811',
 '897654321']
 for values in list1:
 if re.match(r'[8-9][1-3][0-9]{9}', values):
 print("Criteria matched for ceu number!")
 else:
 print("Criteria failed")

```

Criteria matched for cell numbers  
Criteria matched for cell numbers  
Criteria failed!  
Criteria matched for cell numbers.

# vowels

import & e

Impost & S&T = "Plant is life overall"

```
output = re.findall(r'lb[aeiouAEIOU]w+', str)
print(output)
```

>>> ['is', 'overau']

# host & domain

importing and exporting of goods

seq = abc.tcsctcsc@edu.com xyz@gmail.com  
pattern = x'(1m) + (1n)

$$\text{Output} = \sigma e^{-\frac{1}{2}(\|w\|_2^2 - b^2)}$$

```
print(output)
```

```
>>> ['abc':trace@edu.com]
```

Import of first 2 letters

$s = 1$  یعنی  $s = 1$  ہے

$P = 41 [1ns/μs]$

Paint( $\sigma$ )

$$j=0$$

$b=0$  [B-3]  $\{ \text{[B-3]} \}$   $\frac{1}{2} \text{[B-3]}$   $\frac{1}{2} \text{[B-3]}$   $\frac{1}{2} \text{[B-3]}$

bv info; (97-82-2)

БАШНЯ ИЗДОВІЛІВСЯ ПО ЕОДЕМУ

ప్రాణ వ్యవహార వైద్యత్వం కుటుంబములో

Chlorophyll a = 13.2

1965-1966 (2nd) + 9728

Digitized by srujanika@gmail.com

Step 7: Import re module declare string use  
and the module with findall() for finding  
the vowels in string & declare same

Step 8: Import re module declare the host &  
domain name declare pattern for  
separating the host & domain name. use  
findall() & print output

Step 9: Import re module enter a string  
use pattern to display only two elements  
of particular string use findall(). use  
for condition & subsequently use  
an if condition.

which is 022 2629 905 is a big <  
> 28-3000.

Question with Ans

Ques question with 32 92 2  
combined to give 192 and  
then 192 is divided by 32  
gives 6 with remainder 8  
-(8) can be repeated 2 times  
(8) can be repeated 2 times  
thus (8) goes out 32 times  
224 is GCD through

ss

## PRACTICAL - 5

Aim : To make use of GUI app along with the basic pack method.

7. Pack

Algorithm  
Step 1 Use the tkinter library for importing the features of text widgets.

-1. Lab  
2. Pac root.

Step 2 : Create a variable from a text variable widget at section 220.

Step 3 : Use pack( ) along with the object created from text method use the parameters.

1) side = Top , padx = 20 , ipadx = 60  
2) pady = 50

Step 4 : Use the mainloop

Step 5 : Now the mainloop repeat above step with a label method

1) Name Parent window

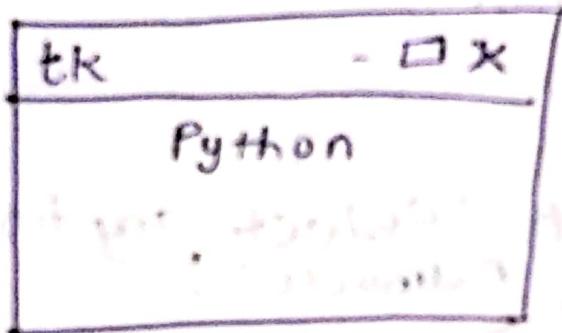
2) The text attribute which defines string.

3) The background color (bg)

Now, use the pack( ) with relevant attributes

```

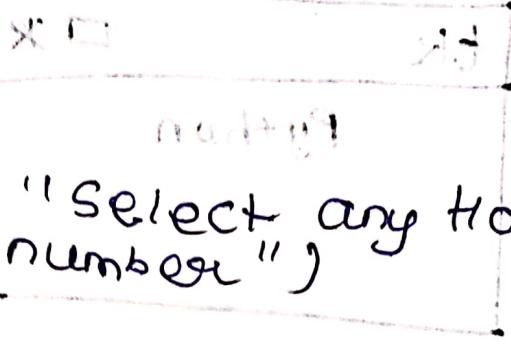
from tkinter import *
root = Tk()
T1 = Text(root)
T1.insert(END, "Hey There! My Name is
Anni.")
T1.pack(side=TOP, padx=20, pady=30, ipadx=40,
 ipady=50)
L1 = Label(root, text="Label", bg="red", fg="blue")
L1.pack(side=LEFT, padx=10, ipadx=20, ipady=30)
root.mainloop()
```



```

program import
from tkinter import *
def sel1():
 selection = "Malcolm"
 label.config(text=selection)
def sel2():
 selection = "BRO"
 label.config(text=selection)
def sel3():
 selection = "CHIRAG"
 label.config(text=selection)
def sel4():
 selection = "Sachin"
root = Tk()
var = IntVar()
L1 = Label(root, text="Select any hold number")
L1.pack(side=TOP)
R1 = Radiobutton(root, text="1721", variable=var, value=0, command=sel1)
R2 = Radiobutton(root, text="1722", variable=var, value=1, command=sel1)
R3 = Radiobutton(root, text="1743", variable=var, value=2, command=sel2)
R4 = Radiobutton(root, text="1744", variable=var, value=3, command=sel3)
label = Label(root)
label.pack(side=Bottom)

```



Aim: To make use of Radio button widget for selection of one on the multiple option.

Algorithm: Step 1: Use the Tkinter Method to import the relevant method.

Step 2: Define a function which tell us about given selection mode from

Step 3: Use the config method along with label method.

Step 4: Now define the parent window of define using control vari.

Step 5: Now Create object of Radio button which will take following

1) Position on parent window.

2) Text variable.

3) Define variable argument.

4) Corresponding value & trigger.

5) Corresponding value of the given function.

Jyoti

CS

## PRACTICAL - 5 Message box

Aim - To make use of message box

Algorithm

S1: Import relevant method.

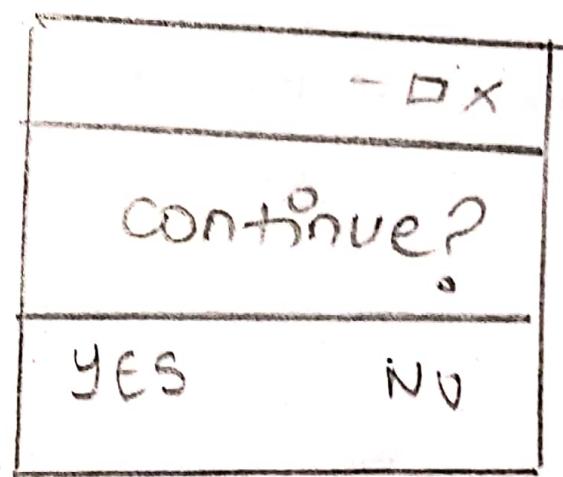
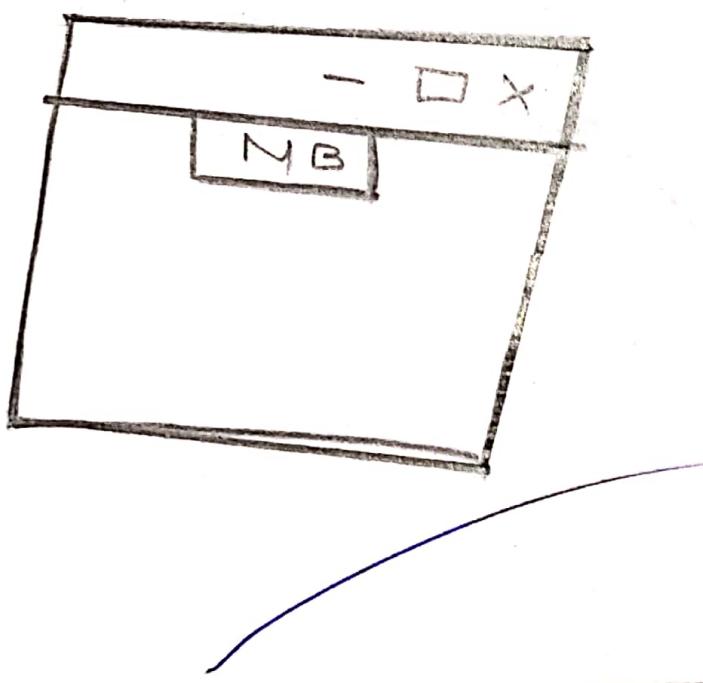
S2: Define function & use message box along with available function.

S3: Create object from buffered method place on parent window & command attribute.

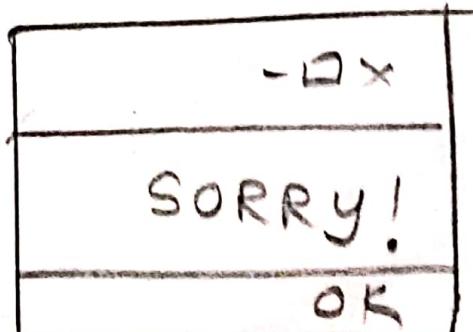
S4: Use pack and finally mainloop.

>>>

```
from tkinter import *
from tkinter import messagebox
def main():
 messagebox.showaskyesno("Action",
 "continue?")
 messagebox.showerror("Error", "No can't
 go ahead")
root = Tk()
root.config(bg = "green")
B1 = Button(root, text = "message box")
B1.pack()
mainloop()
```



Janitor

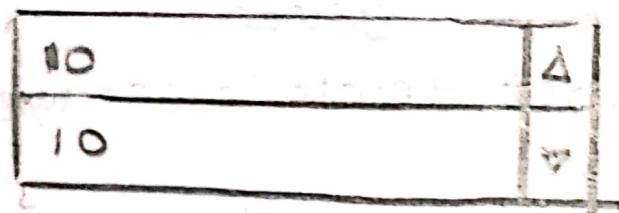


Q8

Code

```
>>> from tkinter import *
root = Tk()
s1 = Spinbox(root, from_=0, to = 10)
s1.pack(anchor = S)
root.mainloop()
```

\* Output



Space  
under scope

## PRACTICAL-5 (SPINBOX)

Aim : To make use of Spinbox

; Step 1 : To create an object from the tk method & subsequently.

Step 2 : Make the object so created onto the parent window & trigger.

Step 3 : Use the anchor pack method to provide direction

Step 4 : Use mainloop to terminator.

## Panned Window

S1: Create an obj from panned window & use the pack method with attribute fill and expand.

S2: Create an object from label method & put it onto the panned window with the text attribute and use the add method to label the new object.

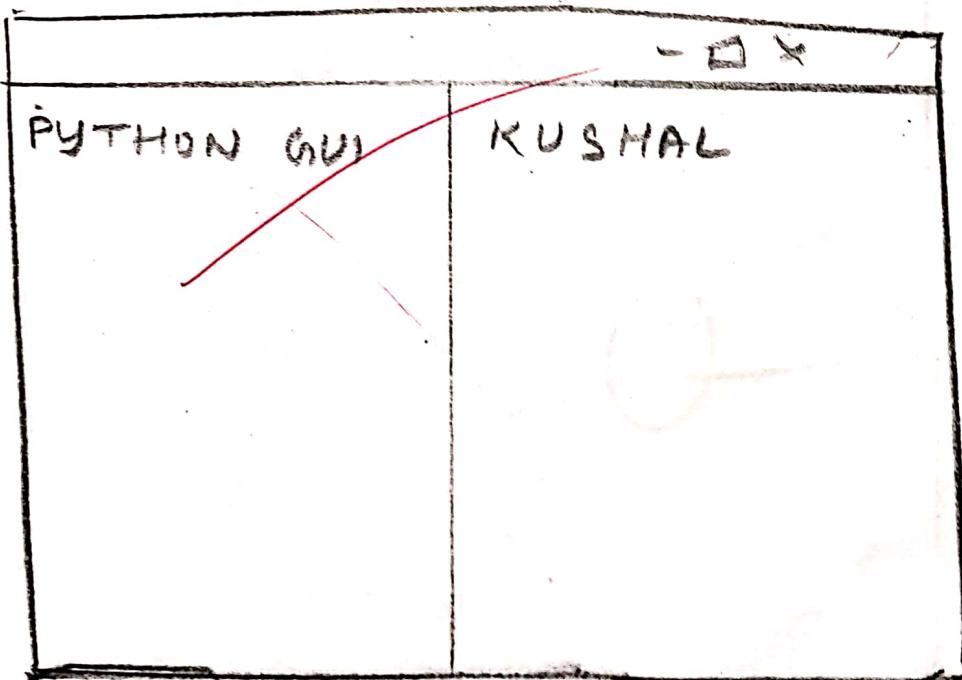
S3: Now create another label object & place it onto the 2nd pane.

Step 4: Use mainloop.

## Code

```
>>> from tkinter import * 32
>>> root=TK()
P=PanedWindow(bg="red")
P.pack(fill="y", "Python GUI", bg="green")
P.add(L1)
P1=PanedWindow(P, orient=VERTICAL, bg="blue")
P.add(P1)
L2=Label(P1, text="KUSHAL", bg="gray")
P1.add(L2)
root.mainloop()
```

Output:



→ from tkinter import \*

root = Tk()

c1 = canvas (root, height=400, width=400)

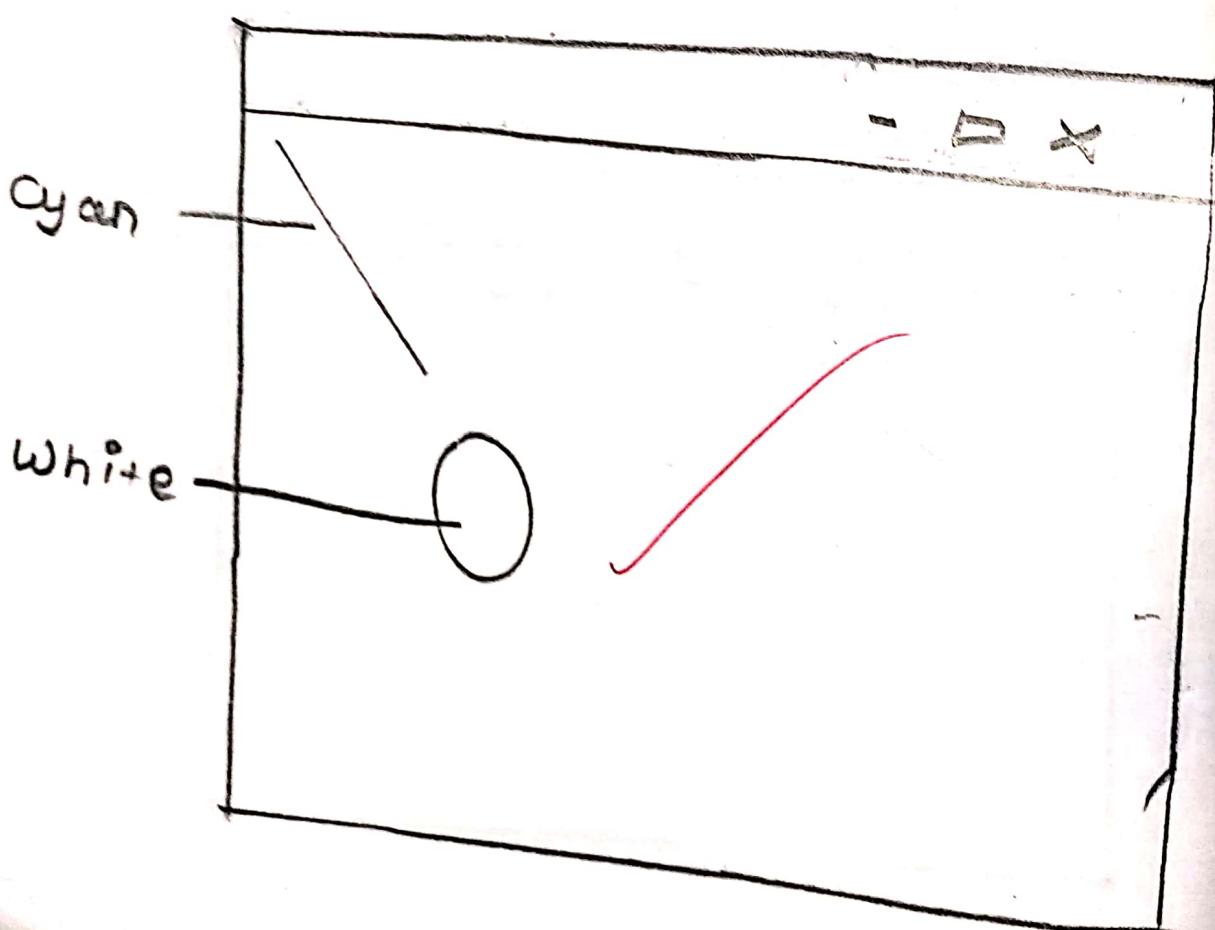
oval = c1.create\_oval (20, 140, 150, 250, fill="white")

line = c1.create\_line (30, 40, 50, 60, fill="cyan")

c1.pack()

root.mainloop()

OUTPUT



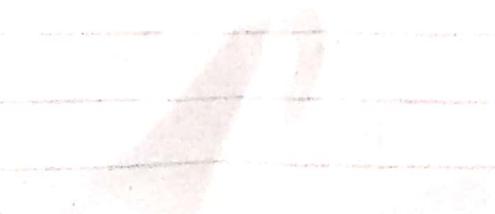
## Canvas Window

S1° Use the `tkinter` window and create an object from the canvas method & use attribute height, weight, bg color & parent window object.

S2° Use the method "create\_oval", create arc along with the canvas object so created & use the co-ordinate values. Also use the field attribute to design various colours.

S3° Now call back method & mainloop method.

~~with its help we can do many things~~



## PRACTICAL - 6

Aim: Database connectivity.

Step 1: Import db library  
use the open method for  
database by specifying  
name of the database  
along with corresponding file.

Step 2: Use the objects by  
accessing the given webpage  
and the corresponding regular  
web size.

Step 3: Check whether the  
given url objects address  
with the regular of the  
pages is not equal to none  
then display the message  
from the address else  
not formed.

# Program

Program #

34

```
>>> import dbm
db=dbm.open("database", flag, "c")
if db["www"]!=None:
 print("good")
else:
 print("Not good")
```

Output: good

~~import dbm~~

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

~~if db["www"]!=None:~~

~~print("good")~~

~~else:~~

~~print("Not good")~~

## # program

```
>>> import os,sqlite3
>>> connection=sqlite3.connect("Student.db")
>>> c1=connection.cursor()
>>> c1.execute("Create Student Table
< sqlite3 Cursor object at 0x02D2F020
>>> c1.execute("Insert into Student
values ("Rakesh", 1840
< sqlite3 Cursor object at 0x02D2F020
>>> c1.execute("Insert into Student
values ("Sachin", 1842,
< sqlite3 Cursor object at 0x02D2F020
>>> c1.execute("Commit")
>>> c1.fetchall()
>>> c1.execute("Drop table Student")
```

### Output:

```
[("Rakesh", 1840, 25-01-2000),
 ("Sachin", 1842, 24-04-2001)]
```

JrnV2

## b) CURSOR objects.

Step 1: Import the corresponding library taking the data of database.

Step 2: Now create connection objects using Sqlite Library & connecting method.

Step 3: Now create the cursor objects using cursor method from connection.

Step 4: Now use the executing method for creating the table with the column name & respective datatype.

Step 5: Now with the cursor objects insert the different field considering datatype.

~~Step 6: use commit method.~~

Step 7: Use the execute statement along with the cursor objects using selecting from where clause.

Step 8: Use the execute method & drop table Syntax for terminating the database finally use close method.