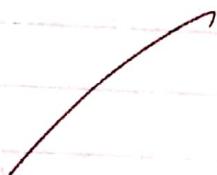


- S1: Take the 2ⁿbit or def two variable for taking the input.
- S2: def function which return the value with respect to the given algebraic formula.
- S3: Point the define function such as add, sub, mult, div, moddiv, fmoddiv to allow the user to choose accordingly.
- S4: Use the of loop respectively to give the user proper function as he requires.
- S5: Point the answer with respect to the proper function.
- S6: Repeat the fourth & fifth step for all the defined function such as def add, def sub, def mult, def div, def moddiv, def fmoddiv.



```

x = int(input("Enter 1st value"))
y = int(input("Enter 2nd value"))

def add(x,y):
    return x+y

def sub(x,y):
    return x-y

def multi(x,y):
    return x*y

def div(x,y):
    return x/y

def moddiv(x,y):
    return x%y

def slowdiv(x,y):
    return x//y

```

Point ("1. addition \n 2. subtraction \n 3. multiplication \n 4. division \n 5. mo")

if choice == 1:

Point ("Addition of x,y is", add (x,y))

elif choice == 2:

Point ("Subtraction of x,y is", sub (x,y))

elif choice == 3:

Point ("Multiplication of x,y is", multi (x,y))

elif choice == 4:

Point ("Division of x,y is", div (x,y))

elif choice == 5:

Point ("Modulus of x,y is", moddiv (x,y))

elif choice == 6:

Point ("Slow division of x,y is", slowdiv (x,y))

Out put

Enter 1st value : 8

Enter 2nd value : 4

1. addition

2. subtraction

3. multiplication

4. Division

5. Modulus

6. float division

Enter selection :

Addition of x,y is = 12.

4. Objective : Demonstrate the use of different file accessing mode , different other bytes read method.

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

Step 2: Now open the file in read mode and then read(), readline() and store the output in variable and finally display the contents of variable.

Step 3: Also use the file object for checking the no. the file is the five mode i.e. which is opened what! the file is still open or close and finally the output of the softspace attribute.

Step 4: Now open the file object in wfile mode write some another content, close subsequently then again open the file object in wt mode that is the wb+ mode and write contents.

Step 5: Open file object in read mode, display the written contents and close off again in rt mode with parameter passed and display the output subsequently.



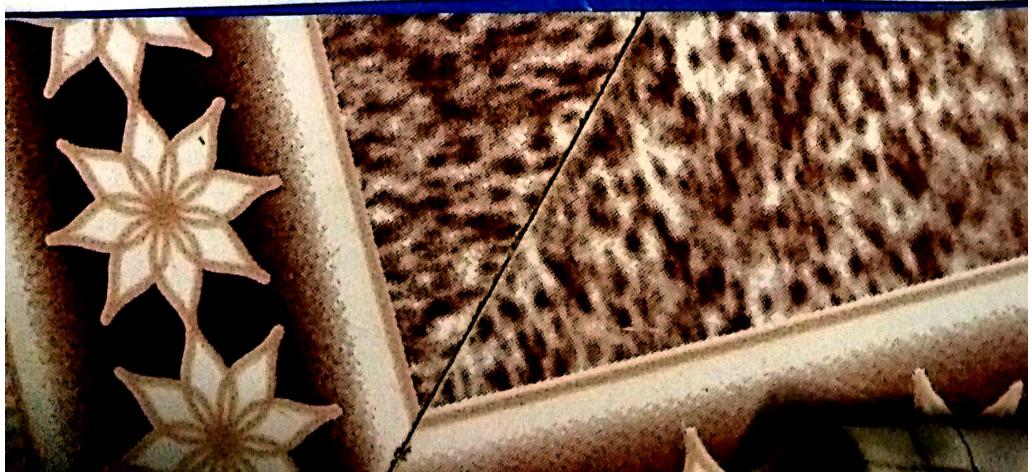
IS

Step 6: Now open jobobject with method write with file object again open the file object in append mode and close the job object in append mode op.

Step 7: open the file object in read mode , declares a variable and perform job object dot tell method and store the output consequently in variable .

Step 8: Use the seek method with the arguments with opening the file object in read mode and closing subsequently

Step 9: open jobobject the read mode also use the readlines methods and close the output consequently in and print the same for calculating the length use the join function statement and display the length .



```

# file attribute
a = fileobj.name
print ("name of file (name attribute):", a)
>>> ('name of file (name attribute): abc.txt')

b = fileobj.closed
print ("closed attribute:", b)
>>> ('closed attribute:', 'True')

c = fileobj.mode
print ("file mode", c)
>>> ('file mode', 'r')

d = fileobj.softspace
print ("softspace")
print ("softspace", d)
>>> ('softspace!', 0)

```

```

# w+ mode
fileobj = open ("abc.txt", "w+")
fileobj.write ("sawabh")
fileobj.close()

```

```

# r+ mode
fileobj = open ("abc.txt", "r+")
s1 = fileobj.read()
print ("output of r+", s1)
fileobj.close()
>>> (output of r+, sawabh)

```

22

```
# append mode  
fileobj = open ("abc.txt", "a")  
fileobj.write ("data structure")  
fileobj.close()  
fileobj = open ("abc.txt", "a")  
fileobj.write ("data structure")  
fileobj.close()  
>>> print ("Output of append mode", fileobj.read())  
>>> fileobj.close()  
>>> print ("Output of append mode", "chandresh", "data structure")
```

tell

```
fileobj = open ("abc.txt", "r")  
pos = fileobj.tell()  
print ("tell()", pos)  
fileobj.close()  
>> print ("tell()", pos)
```

seek

```
fileobj = open ("abc.txt", "r")  
stx3 = fileobj.seek (0, 0)  
str = fileobj.read (100)  
print ("The beginning of file =", str)
```



finding Length of different line exist with in file

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

```
strq = fileobj.readlines()
```

```
print ("Output" + strq)
```

```
for line in strq
```

```
    print (len (line))
```

```
fileobj.close ()
```

>>> Output :- ['College database']

✓
note

Jan 16/2017

IS

mytuple1 = ("Abhay", "Deepak", "Aniket")
mytuple1 = iter(mytuple1)
Point (next(mytuple1))
Point (next(mytuple1))
Point (next(mytuple1))

>>> Abhay
>>> Deepak
>>> Aniket.

Q) mytuple = ("Abhay", "Singh")
for x in mytuple
print(x)

>>> Abhay
Singh



Practical 2

Q) Aim: To display elements of a tuple using the iterator method

Algorithm:

- ① Form a tuple containing some elements.
 - ② Use the `iter` method followed by the next method to print the variable.
 - ③ Print the element.
-) Aim - to use the iterator method with `for loop`.

Algorithm:

- ① Form a tuple contains element in it.
- ② Use the `for` conditional statement to the elements.
- ③ Print the elements.

78

(3) Aim To print odd number using after method.

Algorithm:
① Define a class and within that define a after method which will initialize the first variable within the contain object

② Now use the next method and define the logic for displaying the odd values.

③ Create an object to print the value.

(4) Aim Using afterable displaying first 20 number

① Define a after method with a arguement and initialize it a first value.

② For extracting the next element from the contain use the next method with an arrangement and compare no. of element required in a contain.

③ Now create an object from the given class and the object as an argument After method



```

⑥ class odd:
    def __init__(self):
        self.num = 1
        return self

    def __next__(self):
        num = self.num
        self.num += 2
        return num

```

```

myobj = odd()
myiter = iter(myobj)
x = int(input("Enter a number"))
for i in myiter:
    print(x)

```

Output :-
=> Enter a number 8

1
2
5
7

⑦ class myclass:

```

def __iter__(self):
    self.a = 1
    return self

def __next__(self):
    if self.a <= 20:
        x = self.a
        self.a += 1
        return x
    else:
        raise StopIteration

```

12
soft a += 1
return &

class:
raise StopIteration

myobj = my class()
myiter = iter(myobj)
for x in myiter
print(x)

Output

1 2 3 4 5 6 7 8 9 10
11 12 13 14 15 16 17 18 19 20

Step 4 : Now use in the conditional statement disp all the values from the given condition.

75

Q Aim : Write a program check the given what whether odd or even number.

Step 1 : Declare a listnum variable and declare d element.

Step 2 : Now use map method with help of Lambda function.

Step 3 : Given two argument display the output

Step 4 : Define even function with in parameter the using conditional Statement check whether the number is even and odd and return respectively.

```
listnum = [0, 4, 5, 7, 11, 13, 15, 17, 19]
```

```
listnum = list(map(lambda x: x%5, listnum))
```

```
print(listnum)
```

```
def even(x):
```

```
    if x%2 == 0:
```

```
        return "even"
```

```
    else:
```

```
        return "odd"
```

```
list(map(even, listnum))
```

Output:

```
[0, 1, 0, 1, 2, 3]
```

```

def square(x):
    return (x * x)

def cube(x):
    return (x * x * x)

func1 = [square, cube]
for i in range(4):
    value = list(map(lambda x: x(i), func1))
    print(value)

```

Output:

[16, 64]

[1, 8]



Q. How is to find square root of a number by long division method?

Ans: 1. Define a function named Square with parameter which will obtain output of square of given number.

Ans: 2. In similar function declare fraction and get the return value raised 3rd and 2nd time same.

Ans: 3. Call the declared function using f.



Q.S

③ Aim :- without using map function.

Step 1: Define a list variable with a given set of numbers.

Step 2: Define an empty list which will contain output use the for conditional statement and subsequently use append() method for finding the sequence Square Value and finally print statement for the output.

```

l = [1, 2, 3, 4, 5]
empty = []
for i in l:
    empty.append(i * 2)
print(empty)

```

Output:-

- []
- [1, 4]
- [1, 4, 9]
- [1, 4, 9, 16]
- [1, 4, 9, 16, 25]

Q1

a) program:

try:
to open

try:
obj = open("abc.txt", "w")
obj.write("Hello World")

Except IOError:
print("Hello World")

else:
print("Operation successful")

Output:
Hello World

b) program:

try:
a = int(input("Enter:"))

Except ValueError:
print("Value Error")

Except IOError:
print("Environment error")

else:
print("Operation successful")

Output:
Enter a= : 89
Operation Success

Enter a= : 2+3

This is

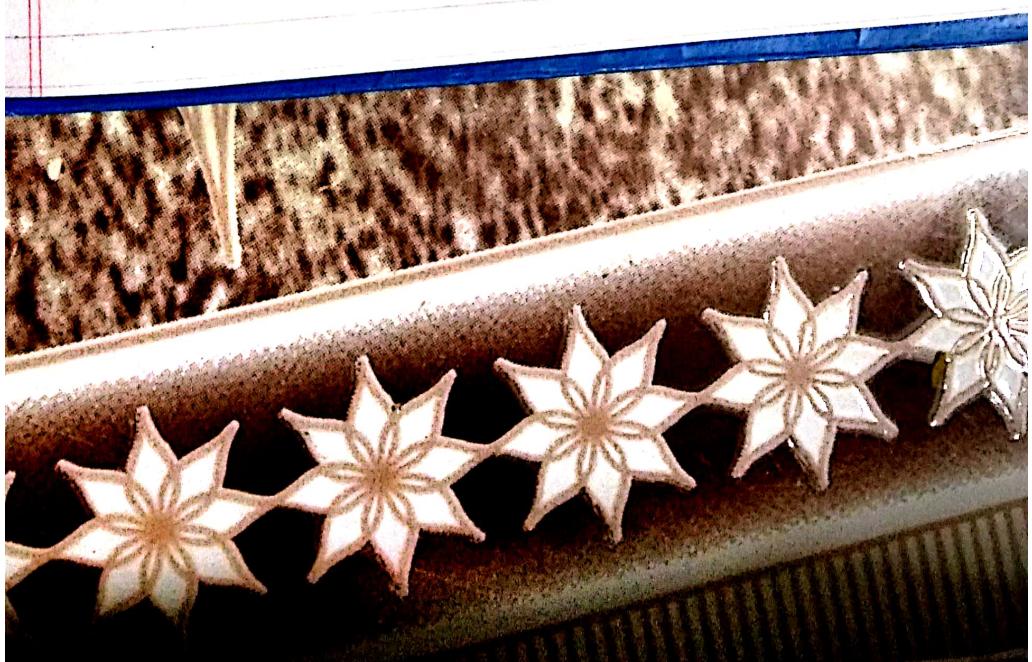
Value Error



Practical No. 3

A) Algorithm

- 1) Use the try block + to define the normal block of action.
- 2) Example Define a file object & open the file in the write mode and write some content onto the file.
- 3) Use an except block with ValueError as an exception and display the appropriate message to the user else display the message that the operation is carried out successfully.
- 4) Write program to demonstrate the use of value error in give program.
- ~~Step 1: Accept the user input of integer datatype in the try block~~
- ~~Step 2: Define the except block with ValueError as a keyword and display appropriate message.~~
- ~~Step 3: Define else block if none of the errors is occurred and display message accordingly~~
- Dr. M



Practical 4

Regular Expression.

Step1: Import re module declare pattern and define sequence use match method with declare arguments if arguments matched than print the same otherwise print pattern NOT FOUND!

Step2: Import re module declare pattern with backslash and meta character. Declare string value. use.findall() with arguments and print the same.

Step3: Import re module declare pattern with character use the split() and print the output

```
# match()
```

```
import re
```

```
pattern = r"fycs"
```

```
sequence = "fycs represent computer science stream"
```

```
if re.match(pattern, sequence):
```

```
    print("Matched Pattern Found!")
```

```
else:
```

```
    print("Not Found!")
```

```
>>> matched Pattern Found!
```

```
# numerical values (segregation)
```

```
import re
```

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

```
string = 'Hello 123, howdy 789, 45 hours'
```

```
output = re.findall(pattern, string)
```

```
print(output)
```

```
>>> ['123', '789', '45']
```

```
# split()
```

```
import re
```

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

```
string = "Hello 123, howdy 789, 45 hours"
```

```
output = re.split(pattern, string)
```

```
print(output)
```

```
>>> ['Hello', 'howdy', 'hours']
```

28.

nospace!

import re

string = 'abc def ghi'

Pattern = re.compile('st+')

replace = ""

V1 = re.sub(pattern, replace, string)

print(V1)

>>> abc def ghi

group()

import re

sequence = 'python is an interesting language'

V2 = re.search('Python', sequence)

print(V2)

V3 = V2.group()

print(V3)

>>> <sre.SRE_Match object at 0x02810f00>

Python

Verifying the given set of phone number.

import re

list1 = ['8004567891', '9145673210', '7865432981', '9876543210']

for value in list1:

if re.match(r'[8-9]\d{1}[\d{0-9}\d{9}]', value) and len(value) == 10:

print("Criteria matched for cell number!")

else:

print("Criteria failed!")

Step 6: Import os module declare list with numbers.
use the conditional statement here we have used up the
condition statement use of condition for checking first
is either 8 or 9 and next number are in range of 0 to
and check whether the entered numbers are equal to 10
if criteria matches Print all numbers matched otherwise
otherwise Print failed.

Dwji

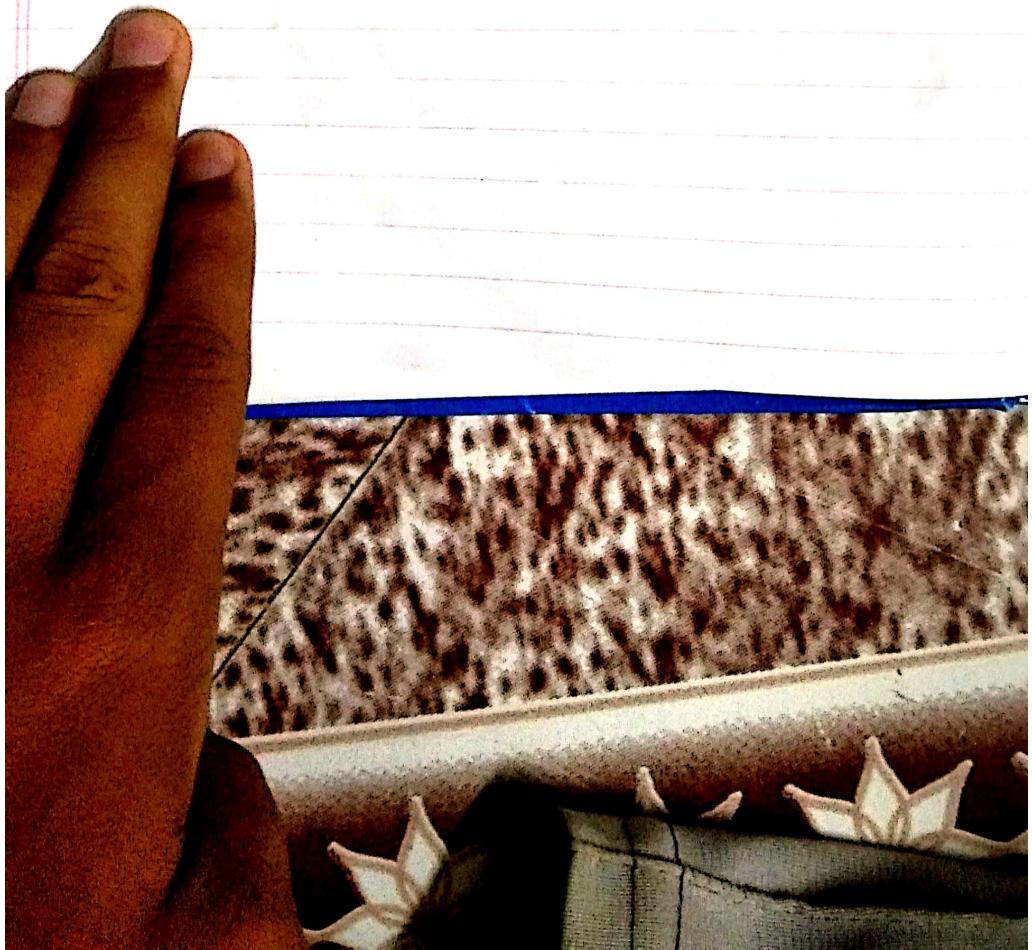
8.8

>>

Step 7: Import re module declare a string use the module with.findall() for finding the vowels in the string and declare the sum.

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

Step 9: Import re module enter a string use pattern to display only two elements of the particular string use.findall() declare two variable with initial value as zero use for condition and subsequently use the if condition check whether condition Satisfy add up the or else increment value and display the values subsequently.



- >> Criteria matched for cell number
- Criteria matched for cell number
- Criteria failed!
- Criteria matched for cell number

eff vowels

100% 50

9m700st 'plant is life overall'

8781

$$\text{output} = \text{BP}$$

print(*attr*)
→ [is, 'overall']

#host & domain

Important role

seq = 'abc .desc (@edu.com , xy2 @gmail.com)'

$$\text{tension} = \delta' [w_1 -] + [w_1 - \gamma]$$

```
output = 80.  
print(output)
```

```
>>> f'abc\tesc\edu.com\1xy2\gmail.com'
```

Counting of first 2 letters:

impost re.

S: 'M.R.O., M.S.b, m.s.c, M.R.E'

$$P = \sigma' [f m_S / m_0] +$$

$O = \text{sp. fundall}$ (P.S.).

print(0)

三〇

$$t = 0$$

for vino:

if ($\ell = \text{ms}$):

18

else:

$m = m + 1$

print("No of males is:", m)

print("No of females is:", f)

>>> [mr, ms, lms, mrs]

('No. of males is: 2')

('No. of females is: 2')

18

Practical 5

AIM : To make use of GUI application along with the basic pack method

Algorithm: Use steps

1. Use the tkinter library for importing the feature of text
2. Create a variable from a text variable & position it on parent window.
3. Use the pack() along the object created from text method & the parameter

① side = TOP , padx = 20 , pady = 40 , ipady = 50

Use the mainloop() method for triggering corresponding event repeat above step with a label method which following argument .

in tk. Toplevel window .

Hey there! my name is Abby

Label

Label.config(text="")

def sel 2():

selection = "Raj"

Label.config(text=selection)

def sel 3():

selection = "Pranay"

Label.config(text=selection)

def sel 4():

selection = "Sachin"

Label.config(text=selection)

root = Tk()

var = IntVar()

L1 = Label(root, text="Select only soft numbers")

L1.pack(side=TOP)

R1 = Radiobutton(root, text="1750", variable=var, value=0, command=sel1)

R1.pack(anchor=N)

R2 = Radiobutton(root, text="1742", variable=var, value=1, command=sel2)

R2.pack(anchor=N)

R3 = Radiobutton(root, text="1743", variable=var, value=2, command=sel3)

R3.pack(anchor=N)

R4 = Radiobutton(root, text="1744", variable=var, value=3, command=sel4)

R4.pack(anchor=N)

Label = Label(root)

Label.pack(side=BOTTOM)

root.mainloop()

Aim : To make use of radiobutton widget for selection of the one option.

Algorithm:

STEPS:

Use the tkinter method to import the relevant module.
Define a function which tells user about given selection mode from multiple option available.

Use the config method along with label method & call the variable as an argument within method.

Now define the parent window & define option using config
Now create object of radiobutton which will take following arguments

① Positioning or parent window .

② Text variable

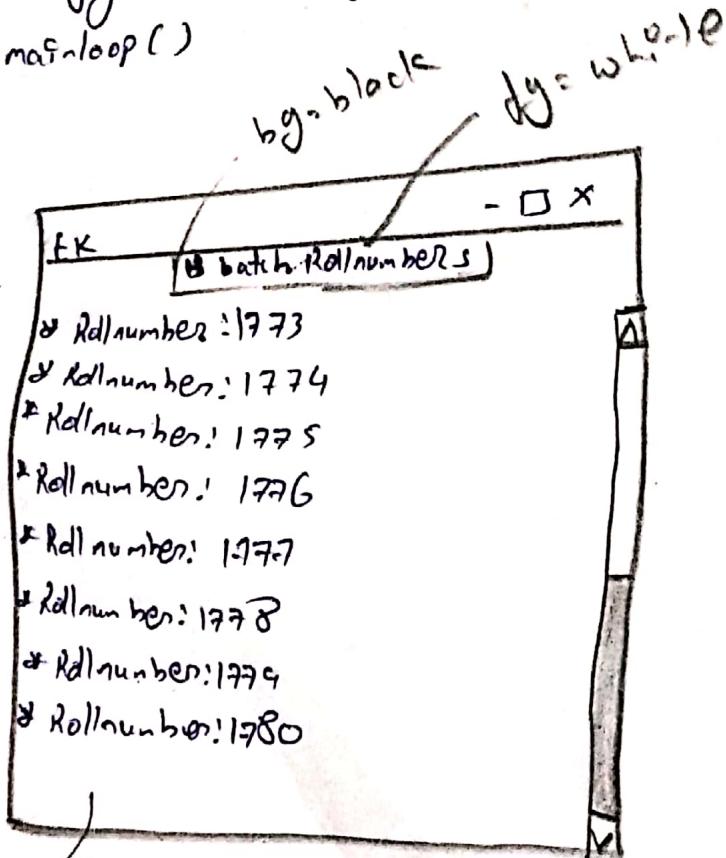
③ Define variable argument .

④ Corresponding value and assign to given function .

Now call the pack method for corresponding Radio object so & and specify argument as an anchor attribute .

Now define a label object and place it onto Parent window using pack method & finally we main loop method .

from tkinter import *
 root = Tk()
 root.geometry ('450x400')
 L = Label (root, text = "B Batch Roll numbers")
 L.pack ()
 scroll = Scrollbar (root)
 scroll . pack (side = RIGHT, fill = Y)
 myList = Listbox (root, yscrollcommand = scroll . set, bg = "light blue")
 for num in range (41, 81):
 myList . insert (END, "# Roll Number: " + str (num))
 myList . pack (side = LEFT, fill = BOTH)
 scroll . config (command = myList . yview)
 root . mainloop ()



bg = light blue

Ans: To make last scroll bar widget of the GUI application.

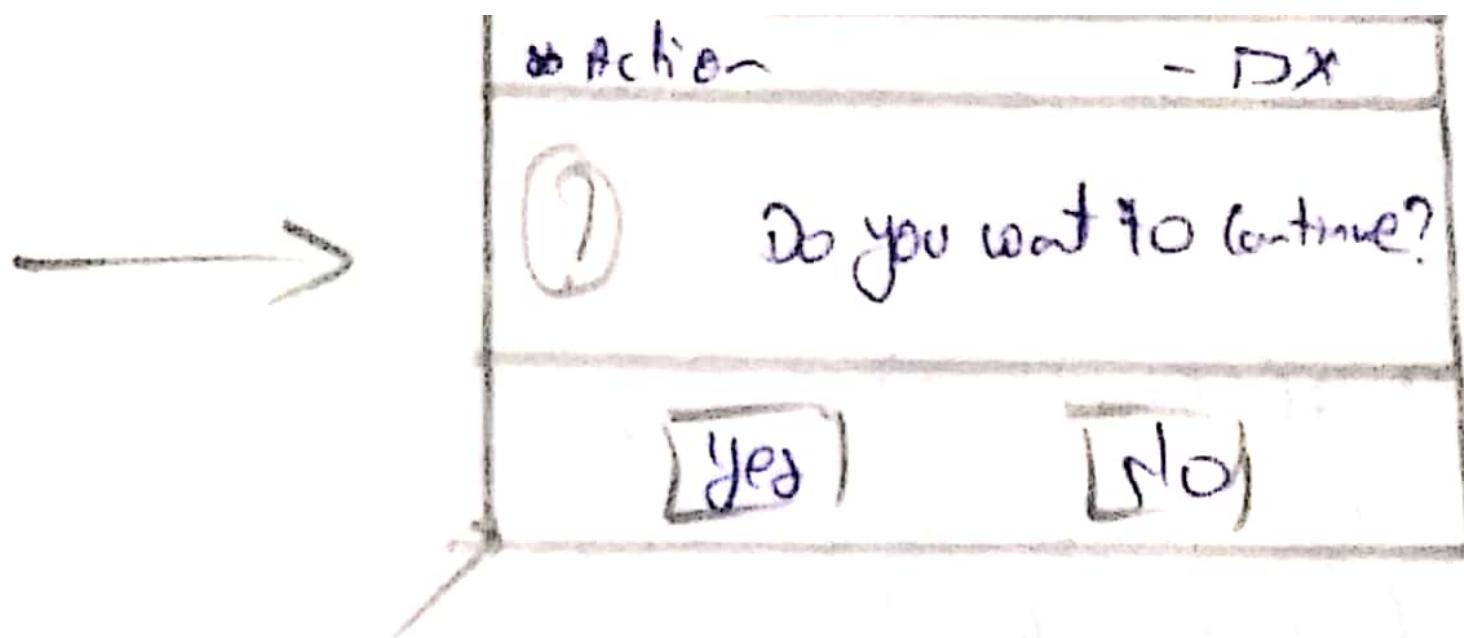
Algorithm:

- 1) Import Tkinter library to use scroll bar widget.
- 2) Create Object corresponding to scroll bar window & handle on this from Scrollbar & then its parent window should be created.
- 3) Create an object of Label method to provide a heading and place it on the parent window.
- 4) use pack method along with object of scroll bar created as argument add label.
- 5) Create an object of last bar method and place it on the parent with attributes you want demand.
- 6) Use for loop to insert values in the object list bar by using method.
- 7) Use config method along with scrollbar object & command option finally tell the mainloop method.

Dinesh

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

Step 4: Use Pack method and finally use the mainloop() method.



Ask user to
message box

Error	- DX
-------	------

```

from tkinter import *
root = Tk()
root.geometry("450x500")
root.config(bg = "light green")
root.title("window 1")
B1 = Button(root, text = "Next", command = main1)
# B1.grid()
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("450x500")
tos.config(bg = "purple")
tos.title("main window")

B3 = Button(tos, text = "Continue", command = main)
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("450x500")
    top.config(bg = "purple")
    top.title("window 2")

```



which will be used to exit from the window
onto current window

Step 4: Define another function which will use the quit method
to terminate the program.

Step 5: Now create an object of main window and use its
method like config, title, geometry etc.

Step 6: Define two buttons which will be placed on the main
window; one to traverse another window and the other
to terminate the program.

step 8: Finally call mainloop method

to start our application and to run it.

So, by this we can see that we have created a simple application which has a button and when we click on it, it will print "Hello World".

So, now we have successfully created our first application.

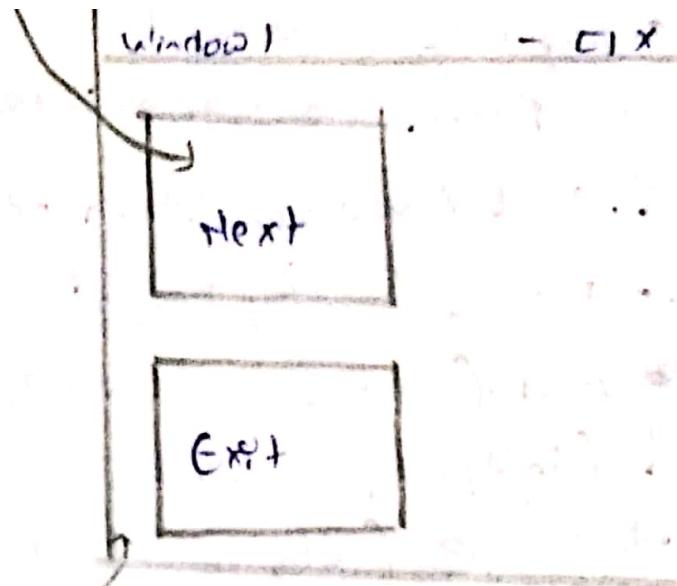
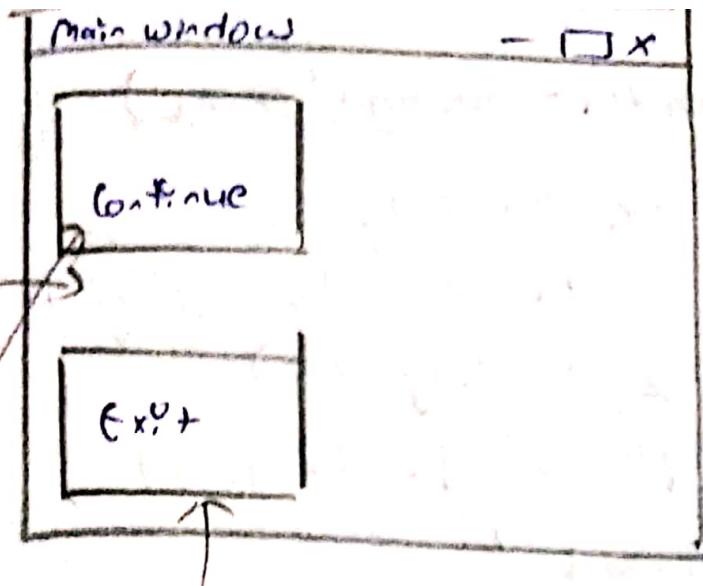
So, this is how we can create a simple application in Python.

So, this is how we can create a simple application in Python.

So, this is how we can create a simple application in Python.

So, this is how we can create a simple application in Python.

So, this is how we can create a simple application in Python.



Take
control
over
window 1

Terminate

ng = light green

the program

Window 2



Jm102

```
from tkinter import *
root = Tk()
root.title("Image")
root.geometry("500x500")
root.minsize = 0
root.config(bg = "white")
leftframe = frame (root, height = 150, width = 150, bg = "green")
leftframe.grid (row = 0, column = 0, padx = 20, pady = 30)
leftframe = frame (root, height = 150, width = 150, bg = "yellow")
rightframe = grid (row = 0, column = 2, padx = 20, pady = 30)
l1 = Label (leftframe, text = "Original", relief = RAISED)
photo = PhotoImage (file = "lighthouse.gif")
ori_image = photo.subsample (3, 5)
label (leftframe, image = ori_image).pack ()
root.mainloop()
```



I am so glad you took my

specimens and objects corresponding to the localities mentioned
and give the their naked sandy title, name and figure

got out a few objects for the men who had been
on the boat with Wright and who signed a document
stating the bulgarian letter.

Step: Use the grid method along the left from object and create another from object and repeat along the right from object.

copy (note a local copy and place it on left pane
 with attributes and with relief attribute as applied
 subsequently use the grid method with row, column value as
 no. 0.0 with zero external padding values

Step 6: We'll be photograph + record with job distribution
specified.

BD

Step 7: Use the subsample method with object of 1m
and give xy coordinates

Step 8: Use Label method & Position subsampling
grid method on left frame - similarly use the big
object and place original image on

Step 9: Finally digger the mar-top method

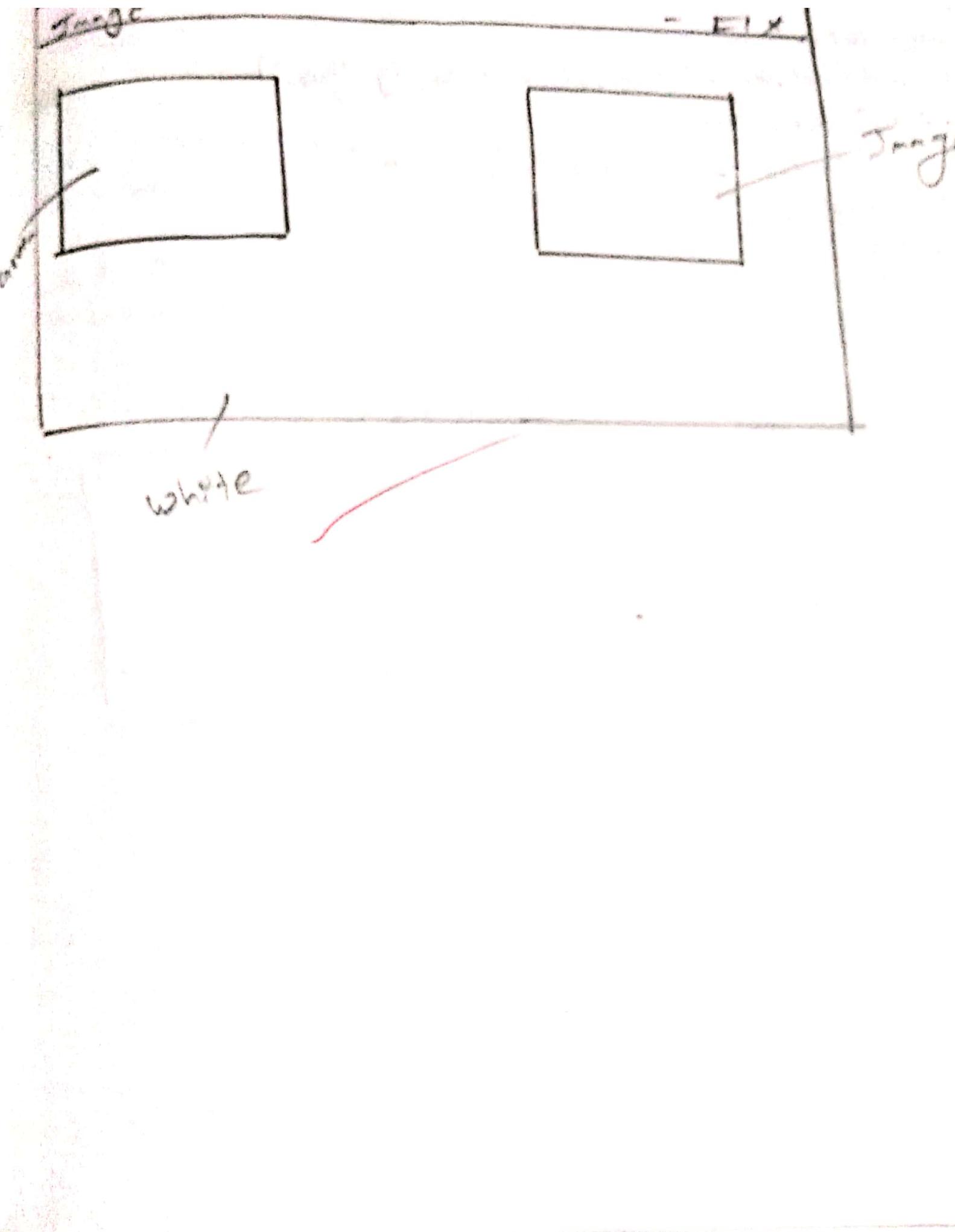
Mar-top method is used to find the position of the object in the frame. It is done by dividing the frame into small squares and then finding the position of the object in each square. The position of the object is then determined by the square in which it is located.

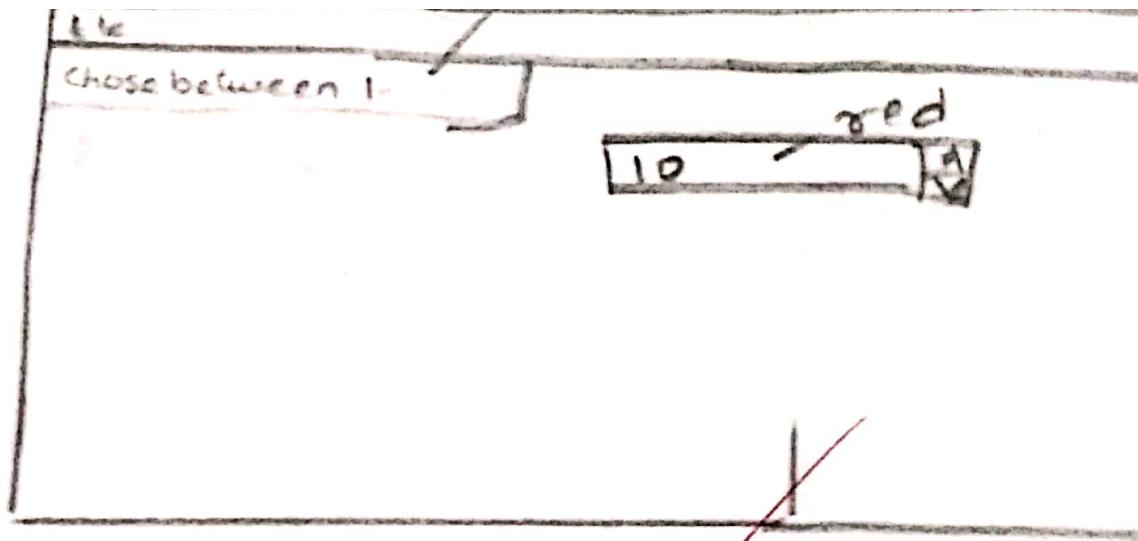
The mar-top method is used to find the position of the object in the frame. It is done by dividing the frame into small squares and then finding the position of the object in each square. The position of the object is then determined by the square in which it is located.

The mar-top method is used to find the position of the object in the frame. It is done by dividing the frame into small squares and then finding the position of the object in each square. The position of the object is then determined by the square in which it is located.

The mar-top method is used to find the position of the object in the frame. It is done by dividing the frame into small squares and then finding the position of the object in each square. The position of the object is then determined by the square in which it is located.







✓

Section

- Q1: Create an object from tv method and subsequently
create an object from the window method.
- Q2: Make the object you created onto the first window
and triggers the scrollEnd event.
- Q3: Use the tick method to provide the duration of
another method.
- Q4: Use the mainloop to terminate.

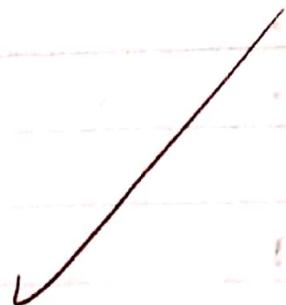
Done

Panned window with text otherwise
to embed the new object

Step 3: Similarly create a second Panned window object
to add it onto the first Panned window with orientation
specified.

Step 4: Create another Label object and place it on to the
second Panned window object and add it on to the second

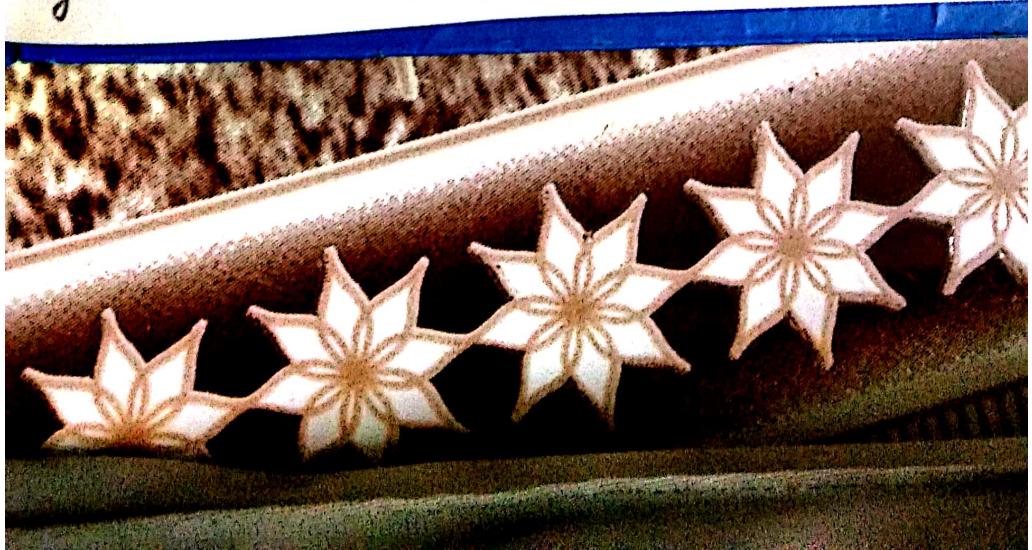
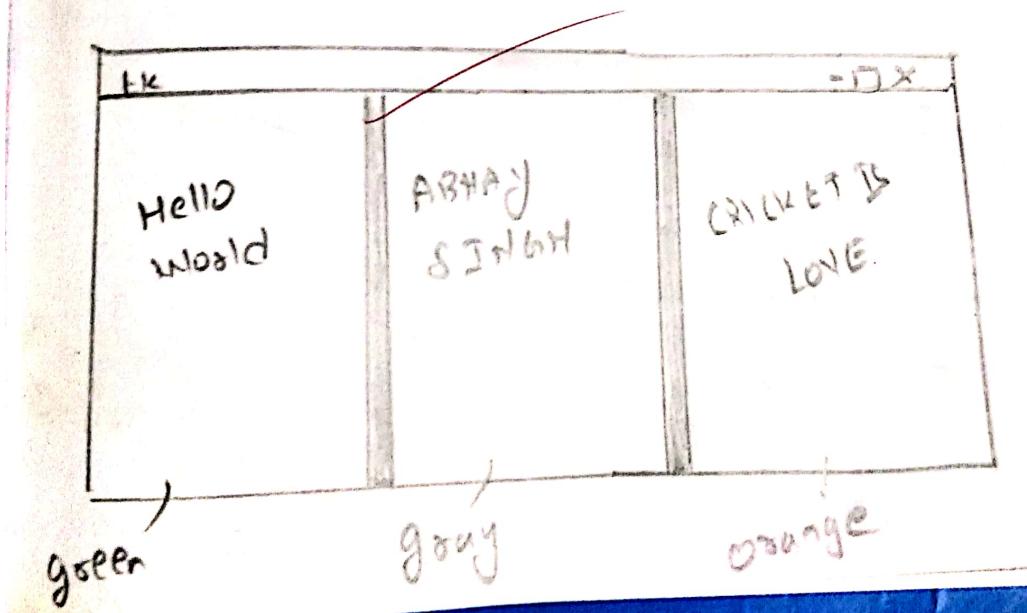
Step 5: Triggered the mainloop method.

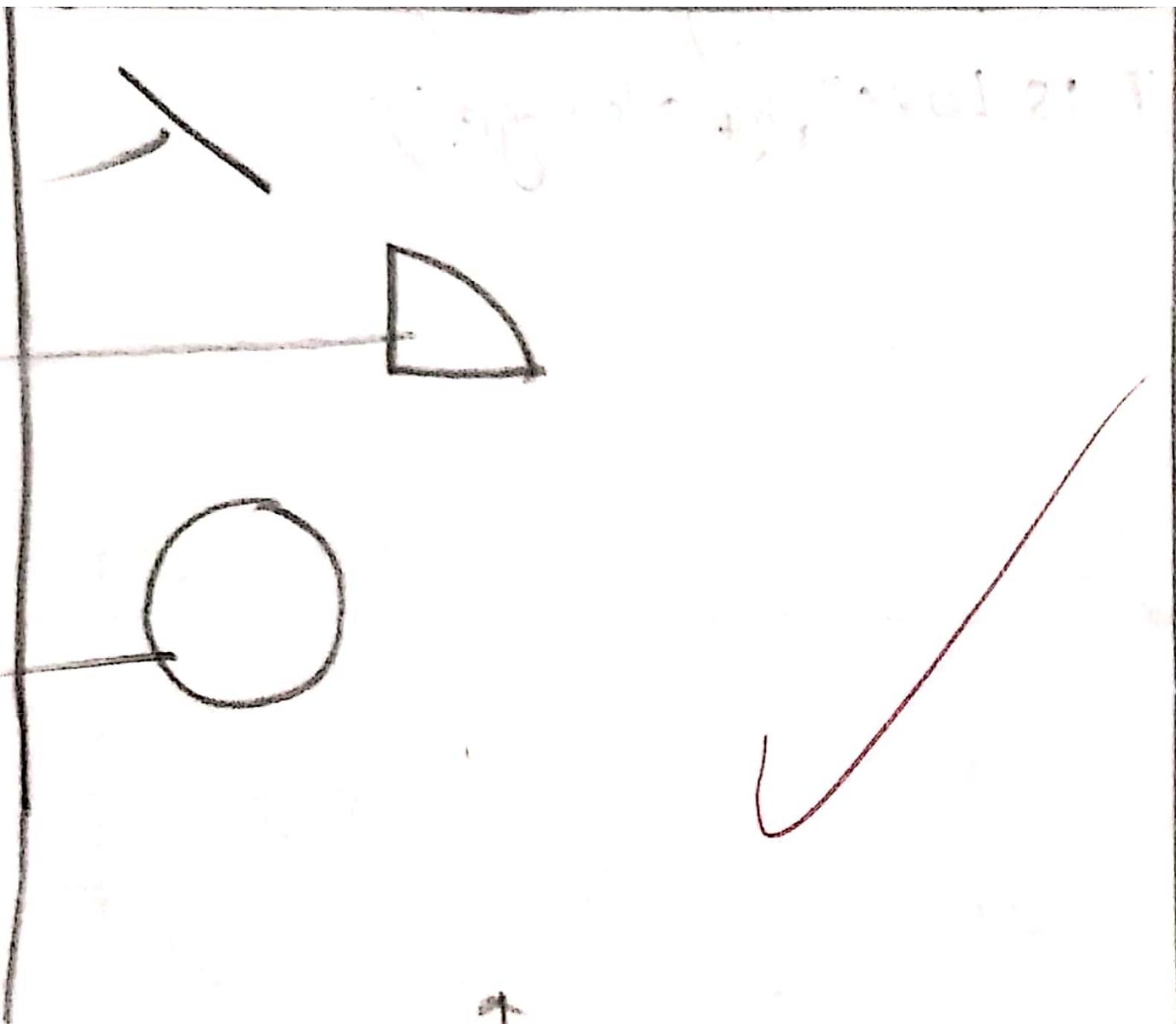


```

from tkinter import *
root = Tk()
p1 = PanedWindow(bg = "red")
p1.pack(fill = BOTH, expand = 1)
l1 = Label(p1, text = "Hello World", bg = "green")
p1.add(l1)
p1 = PanedWindow(p1, orient = VERTICAL, bg = "blue")
p1.add(p1)
l2 = Label(p1, text = "ABHAY SINGH", bg = "gray")
p1.add(l2)
p1 = PanedWindow(p1, orient = HORIZONTAL, bg = "pink")
l3 = Label(p1, text = "CRICKET IS LOVE", bg = "orange")
p1.add(p1)
p1.add(l3)
root.mainloop()

```





Contd.

Step 1: Use the tkinter method and create an object from the canvas method and use the attribute height, weight, bg colour and the parent window object.

Step 2: Use the method create oval, create line to create arc along with the canvas object created, and use the fill=orange value. Also use no file attribute to assign - various colour.

Step 3: Now call the pack method and mainloop method.

Ques 12

What do you mean by class and object?

Explain the difference between class and object.

Explain the difference between class and function.

Explain the difference between class and module.

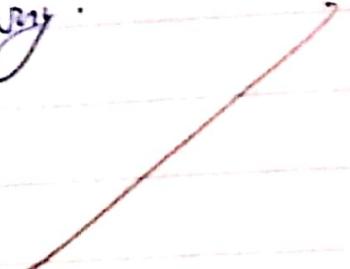
Explain the difference between class and package.

Explain the difference between class and interface.

Step 3: Use the object to create ~~for~~
given the corresponding & regular name for the website

Step 4: Check whether the given URL address with regular name
of the page is not equal to none. Then print the
URL found else print not found.

Step 5: Use the close method to terminate the
database library.



```
import dbm  
db = dbm.open("database", 'c')  
if db["https://www.google.com"] == None:  
    print("URL found")
```

else:

```
    print("URL Not found")
```

```
db.close()
```

wfwf:

URL found.

3A

import os, sqlite3

connection = sqlite3.connect ("student.db")

c1 = connection.cursor()

c1.execute ('create table student (Name, RollNo, DOB)')

c1.execute ('insert into student values ("Rakesh", 1840, 23-06-2002)')

c1.execute ('insert into student values ("Sachin", 1841, 25-04-1996)')

c1.execute ('insert into student values ("Manoj", 1842, 02-02-2000)')

connection.commit()

c1.execute ('select * from student')

c1.fetchall()

c1.execute ('Drop table student')

Output:

[('Rakesh', 1840, 23-06-2002), ('Sachin', 1841, 25-04-1996), ('Manoj',
1842, 02-02-2000)]

- Step 1: Import the corresponding library taking of database connection.
- Step 2: Now create connection object using sqlite library and connecting method for create the new database.
- Step 3: Now create the cursor object using cursor method from the Connection create in step 1.
- Step 4: Now use the cursor object use insert statement for entering the values coordinating into the different field considering the data type.
- Step 5: Now use the execute method for creating the table with the column name & respective datatype.
- Step 6: Use the commit method to complete the transaction use the connection object.
- Step 7: Use the execute statement along with the cursor object for ex step, accessing the value to the data-base using select query where clause.
- Step 8: Finally use the fetchall method for displaying the value for the table using the cursor object.

dtpp: use the execute method and the drop table system
for terminating the database finally use the close method.

JavaDB

```
b=Label(r,text='Class',font='Bold').grid(row=2)

w=Spinbox(from_=1701,to=1790)

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

b=Label(r,text='Roll Number',font='Bold').grid(row=3)

e1=Entry(r)

e2=Entry(r)

e3=Entry(r)

e1.grid(row=0,column=1)

e2.grid(row=1,column=1)

e3.grid(row=2,column=1)

var1=IntVar()

Checkbutton(text='MALE',variable=var1).grid(row=4,sticky=W)

var2=IntVar()

Checkbutton(text='FEMALE',variable=var2).grid(row=5,sticky=W)

Lb=Listbox(r)

Lb.insert(1, 'PYTHON')

Lb.insert(2, 'PROGRAM WITH C')

Lb.insert(3, 'LINUX')

Lb.insert(4, 'DATA STRUCTURE')
```

Student Information

First Name

Last Name

Class

Roll Number 170

MALE

FEMALE

PYTHON

PROGRAM WITH C

CLOUD

DATA STRUCTURE

This opens a new window

1213