|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A picture containing drawing, stop, room  Description automatically generated | Python Programming  Practical # 8 | | | |
|  |  |  |  | |
| **Name** | Sahil Shah | **Roll Number** | 21302C0022 | |
| **Subject/Course:** | Python Programming | **Class** | | SY BSc. IT |
| **Topic** | Importing our own module & Handling Exceptions | **Division** | | C |
|  |  |  |  | |
|  |  |  |  | |
| 1. Open a new file in IDLE (“New Window” in the “File” menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the “Control Flow and Functions” exercise into this file and save it. Now open a new file and save it in the same directory. | | | | |
| Program:  File 1 save as gemo  import math  def sphereArea(r):  return 4\*math.pi\*r\*\*2  def sphereVolume(r):  return 4\*math.pi\*r\*\*3/3  def sphereMetrics(r):  return sphereArea(r),sphereVolume(r)  def circleArea(r):  return math.pi\*r\*r  def squareArea(x):  return x\*\*2  File2 demo  import gemo  def pointyShapeVolume(x,h,square):  if square:  base=gemo.squareArea(x)  else:  base=gemo.circleArea(x)  return h\*base/3.0  print(pointyShapeVolume(4,2.6,True))  print(pointyShapeVolume(4,2.6,False))  Output Screenshot | | | | |
|  | | | | |
| 1. Write a program to implement exception handling. | | | | |
| Program:  try:  num=int(input("Enter the number"))  re=100/num  except(ValueError,ZeroDivisionError):  print("something is wrong")  else:  print("Result is : ",re)  Output Screenshot | | | | |
| 1. Configure the widget with various options like: bg=”red”, family=”times”, size=18 | | | | |
| Program:  import tkinter as tk  win=tk.Tk()  win.title("Practical 9A")  def redClick():  label.config(text="Helvetica Font")  label.config(bg="red")  label.config(font=("Helvetica",16))    def greenClick():  label.config(text="Cambria Font")  label.config(bg="green")  label.config(font=("Cambria",18))    def yellowClick():  label.config(text="Arial Font")  label.config(bg="yellow")  label.config(font=("Arial",14))  label=tk.Label(win,text="Practical 9A",bg='white')  label.pack()  B1=tk.Button(win,text="Red Click",relief='raised',command=redClick)  B1.pack(side="left")  B2=tk.Button(win,text="Green Click",relief='raised',command=greenClick)  B2.pack(side="left")  B3=tk.Button(win,text="Yellow Click",relief='raised',command=yellowClick)  B3.pack(side="left")  win.mainloop()  Output Screenshot  1.    2.    3.    4. | | | | |

|  |
| --- |
| 1. Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc. |
| Program Screenshot  from tkinter import \*  def swap():  if v.get():  e.pack\_forget()  mb.pack(anchor="w",side="right")  l2.config(text="use Menu below")  l2.config(bg="yellow")  l2.config(font=("Helvetica",16,"italic"))  else:  mb.pack\_forget()  e.pack(anchor="w",side="left")  l2.config(text="use Entry box below")  l2.config(bg="green")  l2.config(font=("Cambria",16,"bold"))  e.focus()  t=Tk()  v=IntVar(t)  c=Checkbutton(t,command=swap,text="select to use menu.",variable=v)  c.pack(anchor="w")  f1=Frame(t)  l1=Label(f1,text="select menu item of your choice:")  l1.pack(side="left")  l2=Label(f1,text="use entrybox below.",bg="green",font=("Cambria",16,"bold"))  l2.pack(side="top")  f=Frame(f1)  f.pack(side="left")  e=Entry(f,width=35)  mb=Menubutton(f,width=25,text="Veg",indicatoron=1,relief="sunken",anchor="w")  m=Menu(mb,tearoff=0);mb.configure(menu=m)  for s in "Veg nonVeg Chinese French".split():  m.add\_command(label=s,command=lambda s=s:mb.configure(text=s))  f.pack()  f1.pack()  Output Screenshot  1.  2.  3. |

|  |
| --- |
|  |