



SEGUNDA PARTE

```
#iniciamos importando las librerias necesarias para el proyecto
from tkinter import *
from tkinter import messagebox
import psycopg2

#INSERTAR DATOS EN LA TABLA DE
def table_insert(sub,des,total):
    try:
        conn = psycopg2.connect(user="_postgres",password="_miguel071299",host="_localhost",port="_5432",database="_Miguel_Orrellana")
        cur = conn.cursor()
        cur.execute("""INSERT INTO public."servicios"(subtotal,descuento,total)VALUES(%s,%s,%s)"""%(sub,des,total))
        conn.commit()
        conn.close()
    except:
        print("Conexión fallida")

#LEER DATOS DE LA BASE DE DATOS
def datos_tabla():
    try:
```

```
        cur.execute( SELECT *FROM servicios )
        rows = cur.fetchall()
        cur.close()
        conn.close()
    except:
        print("Conexión fallida")
    return rows

#SALIR
def salir():
    ventanaP.quit()

def resultado():
    pres = [[50,35,70],[40,25,55],[25,10,25]]
    pres_op = [pres[classss.get()][a] for a in range(3)]
    presios.config(text="Comida: Q{}.00 Bebida: Q{}.00 Pelicula: Q{}.00".format(pres_op[0],pres_op[1],pres_op[2]))
    cant_com = int(cantidad_com.get())
    cant_beb = int(cantidad_beb.get())
    cant_pel = int(cantidad_pel.get())
    return pres_op[classss.get()],cant_com,cant_beb,cant_pel,op1.get(),op2.get(),op3.get()
```

```

def calculat():
    try:
        pres_tr,cass_tra,cant_comtr,cant_bebtr,cant_peltr,opp1,opp2,opp3=despliege()
        sub = pres_tr[0]*cant_comtr+pres_tr[1]*cant_bebtr+pres_tr[2]*cant_peltr
        if cant_comtr+cant_bebtr+cant_peltr>=10 and cass_tra>0:
            des = 10
            total = sub*0.90
        elif cass_tra==0 and opp1>0 and opp2>0 and opp3>0:
            des = 5
            total = sub*0.95
        else:
            des = 0
            total = sub
        resultados.config(text="sub: Q{}.00   descuento {}%   total: Q{}".format(sub,des,total))
        table_insert(sub,des,total)
    except:
        print("Cantidad de información muy grande")
#FUNCION QUE DESPLIEGA LOS DATOS DE LA BASE DE DATOS EN UN TEXTO
def estado():
    try:
        datos_guar = datos_tabla()

```

```

    except:
        print("Algo salió mal en la conexión")

#FUNCION QUE BORRA LA BENTANA QUE MUESTRA LOS DATOS DE LA BASE DE DATOS
def borrar():
    comentario.delete(1.0,END)

#-----Construcción del programa-----
ventanaP = Tk()
#-----INICIO VARIABLES INERFASE GRÁFICA-----
classs=IntVar()#variable usada radio_button
op1=IntVar()# --
op2=IntVar()# --|-->Variables check button
op3=IntVar()# --
cantidad_com=StringVar()
cantidad_beb=StringVar()
cantidad_pel=StringVar()

```

```

check_uno = Checkbutton(ventana,text="Comida",variable=op1,onvalue=1,offvalue=0,command=resultado)
check_uno.grid(row=7,column=1,padx=4,pady=4)
check_dos = Checkbutton(ventana,text="Bebida",variable=op2,onvalue=2,offvalue=0,command=resultado)
check_dos.grid(row=8,column=1,padx=4,pady=4)
check_tres = Checkbutton(ventana,text="Película",variable=op3,onvalue=3,offvalue=0,command=resultado)
check_tres.grid(row=9,column=1,padx=4,pady=4)
spin_un = Spinbox(ventana,from_=0,to=150,state='readonly',width=3,textvariable=cantidad_com)
spin_un.grid(row=7,column=2,padx=4,pady=4)
spin_un = Spinbox(ventana,from_=0,to=150,state='readonly',width=3,textvariable=cantidad_beb)
spin_un.grid(row=8,column=2,padx=4,pady=4)
spin_un = Spinbox(ventana,from_=0,to=150,state='readonly',width=3,textvariable=cantidad_pel)
spin_un.grid(row=9,column=2,padx=4,pady=4)
precios = Label(ventana)
precios.grid(row=10,column=0,columnspan=4,padx=4,pady=4)

ventanaP.mainloop()

```

```

text_texto = Label(ventana,text="Listado")
text_texto.grid(row=0,column=1,columnspan=2)

comentario = Text(ventana,width=39,height=9)
comentario.grid(row=1,column=1,columnspan=2,rowspan=4)
scrollvertical = Scrollbar(ventana,command=comentario.yview)
scrollvertical.grid(row=1,column=3,sticky="nsew",rowspan=4)
comentario.config(yscrollcommand=scrollvertical.set)

resultados = Label(ventana)
resultados.grid(row=5,column=0,columnspan=4,padx=4,pady=4)
clase_ = Label(ventana,text="Clases")
clase_.grid(row=6,column=0,padx=4,pady=4)
servicio = Label(ventana,text="Servicios")
servicio.grid(row=6,column=1,padx=4,pady=4)
cantidad = Label(ventana,text="Cantidad")
cantidad.grid(row=6,column=2,padx=4,pady=4)
#-----Despliegue de Radio button-----
Radio_uno = Radiobutton(ventana,text="1° Clase",variable=classs,value=0,command=resultado)

```

FUNCIONAMIENTO

Calcular

Limpiar

Salir

Clases	Servicios	Cantidad
<input checked="" type="radio"/> 1° Clase	<input checked="" type="checkbox"/> Comida	2
<input type="radio"/> 2° Clase	<input type="checkbox"/> Bebida	0
<input type="radio"/> 3° Clase	<input checked="" type="checkbox"/> Película	2

Comida: Q50.00 Bebida: Q35.00 Película: Q70.00

Acciones

Reporte

Calcular

Limpiar

Salir

Listado

sub: Q240.00 descuento 0% total: Q240

Base de datos

pgAdmin

File

Object

Tools

Help

Browser

Servers (1)

PostgreSQL 12

Databases (4)

Miguel Orellana

Tarea 2

postgres

prueba

Login/Group Roles

Tablespaces

Dashboard

Properties

SQL

Statistics

Dependencies

Miguel Orellana/postgres@PostgreSQL 12

Query Editor

Query History

1

2

3

select * from servicios;

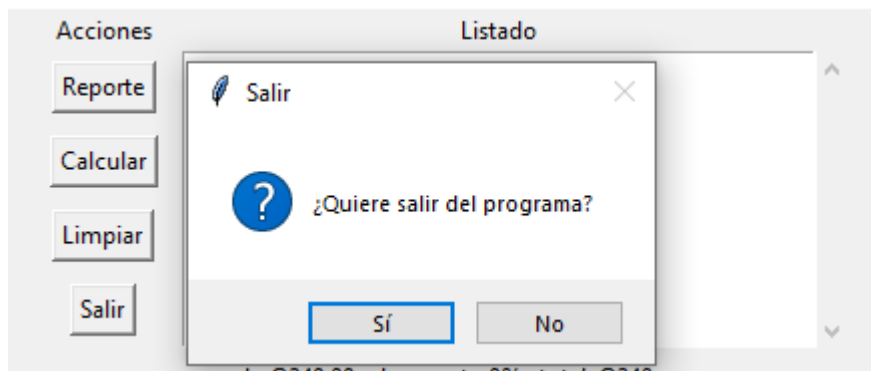
Data Output

Explain

Messages

Notifications

	subtotal numeric	descuento numeric	total numeric
1	275	5	261.25
2	240	0	240



SALIR