

~\OneDrive\Desktop\TAREAFINAL_VAZQUEZ_ZUBIMENDI_MATIAS_GONZALO\calculadora.py

```
1  from tkinter import *
2
3
4
5
6  class CalculadoraModel:
7      """Clase que representa el modelo de la calculadora."""
8      def __init__(self):
9          """Inicializa el modelo de la calculadora."""
10         self.resultado = ""
11
12
13
14
15
16
17     def calcular(self, operacion):
18         """
19         Calcula el resultado de la operación ingresada.
20
21         Parameters:
22             operacion (str): La operación a calcular.
23
24         Returns:
25             None
26         """
27         try:
28             self.resultado = str(eval(operacion))#eval evalúa la operacion y la resuelve
29             (OPERACION DE CADENA,STR A NUMERO INT)
30         except Exception as e:
31             self.resultado = "Error"
32
33
34
35
36
37  class CalculadoraView:
38      """Clase que representa la vista de la calculadora."""
39      def __init__(self, ventana_interfaz):
40          """Inicializa la vista de la calculadora."""
41          self.ventana_interfaz = ventana_interfaz
42          self.ventana_interfaz.title("Calculadora ALUMNO VAZQUEZ ZUBIMENDI MATIAS GONZALO.")
43      )
44
45      # color de fondo y transparencia
46      self.ventana_interfaz.configure(bg='#8a8685')
47      self.ventana_interfaz.attributes('-alpha', 0.9)
48
49      self.frame = Frame(self.ventana_interfaz, bg='#8a8685')
50      self.frame.pack()
```

```
51     self.label = Label(self.frame, text="Resultado:", bg='#8a8685', fg='white')
52     self.label.grid(row=0, column=0)
53
54     self.resultado_var = StringVar()
55     self.resultado_entry = Entry(self.frame, textvariable=self.resultado_var, state='
disabled')
56     self.resultado_entry.grid(row=0, column=1, columnspan=2)
57
58     self.botonnes = [
59         '7', '8', '9', '/',
60         '4', '5', '6', '*',
61         '1', '2', '3', '-',
62         '0', '.', '=', '+', 'C'
63     ]
64
65     self.row = 1
66     self.column = 0
67     for boton in self.botonnes:
68         self.crear_boton(boton)
69
70
71
72
73
74
75
76
77     def crear_boton(self, valor):
78         """Crea un botón en la interfaz gráfica."""
79         Button(
80             self.frame,
81             text=valor,
82             width=5,
83             command=lambda: self.boton_presionado(valor),
84             bg='#8a8685',
85             fg='white'
86
87         ).grid(row=self.row, column=self.column)
88         self.column += 1
89         if self.column > 3:
90             self.column = 0
91             self.row += 1
92
93
94
95
96
97
98
99
100
101     def boton_presionado(self, valor):
102         """
103         Maneja la acción cuando se presiona un botón en la calculadora.
104
```

```
105     Parameters:
106         valor (str): El valor del botón presionado.
107
108     Returns:
109         None
110     """
111     if valor == '=':
112         self.resultado_var.set(self.modelo.resultado)
113     elif valor == 'C':
114         self.resultado_var.set('') # Borrar todo el contenido
115     else:
116         self.resultado_var.set(self.resultado_var.get() + valor)
117
118
119
120
121
122 class CalculadoraController:
123     """Clase que representa el controlador de la calculadora."""
124     def __init__(self, ventana_interfaz):
125         """Inicializa el controlador de la calculadora."""
126         self.ventana_interfaz = ventana_interfaz
127         self.modelo = CalculadoraModel()
128         self.vista = CalculadoraView(ventana_interfaz)
129
130         # Conectar la vista con el controlador
131         self.vista.boton_presionado = self.boton_presionado
132
133
134
135
136     def boton_presionado(self, valor):
137         """
138         Maneja la acción cuando se presiona un botón en la calculadora.
139
140         Parameters:
141             valor (str): El valor del botón presionado.
142
143         Returns:
144             None
145         """
146         if valor == '=':
147             self.modelo.calcular(self.vista.resultado_var.get())
148             self.vista.resultado_var.set(self.modelo.resultado)
149         elif valor == 'C':
150             self.vista.resultado_var.set('') # Borrar todo el contenido
151         else:
152             self.vista.resultado_var.set(self.vista.resultado_var.get() + valor)
153
154
155
156
157
158
159 #PROBANDO MODULO
```

```
160 #print(__name__)
161
162 """
163
164 aqui se imprime __main__ porque se esta ejecutando el modulo calculadora.py en el modulo
    al que importemos
165 este script , se imprimira el nombre del modulo es decir calculadora
166
167 """
168
169
170 if __name__ == "__main__":
171     ventana_interfaz = Tk()
172     app = CalculadoraController(ventana_interfaz)
173     ventana_interfaz.mainloop()
174 else:
175     print("modulo calculadora importado, calculadora.py :)")#si se importa el modulo
    calculadora.py se imprime esto :)
176
```