



TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE CUAUTLA

INGENIERÍA EN SISTEMAS

TOPICOS AVANZADOS DE PROGRAMACIÓN

SEMESTRE ENERO JUNIO-2026

DOCENTE: CABALLERO ALFARO ARÍSTIDES

UNIDAD 1

MARTÍNEZ GALLARDO SANTIAGO

4TO SEMESTRE

GRUPO: 1

25 DE FEBRERO DEL 2026

El presente proyecto consiste en el desarrollo de una calculadora básica con interfaz gráfica utilizando el lenguaje de programación Python y la librería Flet, la cual permite crear aplicaciones visuales modernas de manera sencilla y estructurada. El programa inicia importando la librería Flet con el alias ft, lo que facilita la escritura del código. Posteriormente, se define la función principal llamada main, que recibe como parámetro un objeto page, el cual representa la ventana de la aplicación. Dentro de esta función se configuran las propiedades principales de la interfaz, como el título de la ventana, el ancho, la altura, la desactivación del redimensionamiento y el espaciado interno, estableciendo así un entorno visual fijo y ordenado para el usuario.

Se declara una variable llamada expresion, inicializada como una cadena vacía, cuya función es almacenar los números y operadores que el usuario va presionando. Esta variable es fundamental, ya que actúa como memoria temporal de la operación matemática. Para manipular esta variable dentro de las funciones internas se utiliza la palabra clave nonlocal, lo que permite modificar su valor desde funciones anidadas dentro de main. El programa define varias funciones que controlan el comportamiento de la calculadora. La función agregar_numero se ejecuta cada vez que el usuario presiona un botón numérico; esta función obtiene el valor del botón presionado, lo concatena a la variable expresion, actualiza el texto del display y refresca la página con page.update(). De manera similar, la función agregar_operador permite añadir símbolos como suma o resta a la expresión, siguiendo el mismo procedimiento de actualización.

La función calcular es la encargada de resolver la operación matemática. Utiliza la función integrada eval() de Python para evaluar la cadena almacenada en expresion, convirtiendo el resultado nuevamente en texto para mostrarlo en pantalla. Para evitar que el programa se detenga ante un error, se implementa una estructura try-except, la cual muestra la palabra “Error” en caso de que la expresión no sea válida y reinicia la variable. Además, se incorpora la función limpiar, asociada al botón “C”, que restablece la expresión a una cadena vacía y coloca el valor “0” en el display, permitiendo comenzar una nueva operación.

En cuanto a la parte visual, se crea un objeto Text que funciona como pantalla o display de la calculadora, el cual se coloca dentro de un contenedor con color de fondo y borde definido para mejorar su presentación. Posteriormente, se diseñan los botones numéricos mediante una función llamada boton_numero, la cual genera contenedores de color azul con texto blanco y evento on_click asociado. Estos botones se organizan en filas y columnas utilizando las estructuras Row y Column, formando una cuadrícula ordenada del 1 al 9. De forma similar, se crea la función

boton_operador, que genera botones verdes para las operaciones y los coloca en una fila independiente. Finalmente, todos los elementos se agregan a la página mediante una columna principal que organiza verticalmente el display, las etiquetas de sección, los botones numéricicos, un divisor visual y los botones de operación.

El programa concluye con la estructura condicional `if __name__ == "__main__":`, que ejecuta la aplicación llamando a `ft.app(target=main)` cuando el archivo se ejecuta directamente. En términos generales, este proyecto aplica conceptos de programación estructurada y programación orientada a eventos, ya que cada botón ejecuta una función específica al ser presionado. También integra manejo de errores, diseño de interfaces gráficas y uso de funciones reutilizables para evitar la repetición de código. En conclusión, la práctica permitió comprender cómo desarrollar una aplicación funcional con interfaz gráfica en Python, organizando correctamente sus componentes visuales y controlando la lógica de operación mediante funciones, logrando así una calculadora básica completamente operativa y estructurada.

