**CALCULADORA DE PYTHON**

**INTEGRANTES:** Joaquin Bayon , Ignacio Ledesma , Juan Copa y Agustin Neira  **MATERIA:** Tecnología de control

**CURSO:** 7mo B

**PROYECTO:** Calculadora con Interfaz Gráfica, investigando librerías externas para darle una apariencia más cercana a la realidad. Aplicando conceptos de funciones, clases y excepciones. Agregando un historial de la misma.

**REGLAS:** Respetar los roles designados, para obtener un ritmo de trabajo más apropiado. Los roles pueden ser variables, pero cada uno debe mantener uno de los roles por clase. Deberán realizarse entregas parciales del proyecto. Entre estas entregas parciales haremos al menos una exposición previa cuando estén por el 50% del proyecto, en esta detallaran los detalles técnicos como han conseguido resolver el problema. Posteriormente se realizará una evaluación individual por grupo para analizar el proceso de aprendizaje propio y su incidencia en el proyecto, buscando la detección de participantes pasivos del grupo, dándose como posibilidad en el peor de los casos frente al no compromiso en reiteradas ocasiones, el desaprobar individualmente la materia. Muestra en vivo del funcionamiento de la aplicación y explicación en detalle de lo realizado

**Introducción**

**Objetivo del Proyecto:**

El presente proyecto tiene como objetivo desarrollar una calculadora científica completa utilizando Python y la biblioteca Tkinter para la interfaz gráfica de usuario. La calculadora está diseñada para realizar operaciones matemáticas básicas y avanzadas, proporcionando una herramienta funcional y visualmente atractiva para cálculos científicos.

**Importancia de la Calculadora en Programación:**

Las calculadoras representan uno de los proyectos fundamentales en el aprendizaje de programación, ya que integran conceptos esenciales como interfaces gráficas, manejo de eventos, procesamiento de expresiones matemáticas y gestión de errores. Este tipo de aplicaciones demuestra la capacidad de Python para crear herramientas prácticas y funcionales.

**Python y Tkinter como Herramientas de Desarrollo:**

Python se presenta como el lenguaje ideal para este proyecto debido a su sintaxis clara y bibliotecas robustas. Tkinter, como biblioteca estándar de Python para interfaces gráficas, permite crear aplicaciones de escritorio multiplataforma sin dependencias externas, facilitando el desarrollo y distribución de la aplicación.

**Descripción del Programa**

**Funcionalidades Implementadas:**

La calculadora científica desarrollada incluye las siguientes operaciones:

**Operaciones Básicas:**

**-** Suma, resta, multiplicación y división

- Operaciones con paréntesis

- Cambio de signo

- Función de borrado (backspace y clear)

**Operaciones Científicas:**

- Funciones trigonométricas (sin, cos, tan)

- Logaritmos (log natural y log base 10)

- Potencias (x², x³, x⁻¹)

- Constantes matemáticas (π, e)

- Funciones exponenciales

**Características Adicionales:**

- Modo de ángulos (grados/radianes)

- Historial de cálculos

- Interfaz gráfica intuitiva

- Manejo de errores

**Método de Interacción**

La aplicación utiliza una interfaz gráfica desarrollada con Tkinter que simula el diseño de una calculadora científica real. La interacción se realiza mediante botones clickeables organizados en una cuadrícula de 9 filas por 6 columnas, con un campo de entrada para mostrar las expresiones y resultados.

**Estructura General del Código:**

El programa está estructurado usando programación orientada a objetos con una clase principal `Calculator` que hereda de `tk.Tk`. Esta arquitectura permite una organización clara del código y facilita el mantenimiento y la extensión de funcionalidades.

**Implementación Técnica**

**Arquitectura del Sistema:**

**Clase Principal (Calculator):**

```python

class Calculator(tk.Tk):

def \_\_init\_\_(self):

super().\_\_init\_\_()

# Configuración inicial

self.expression = ''

self.deg\_mode = True

self.history = []

```

La clase encapsula toda la funcionalidad de la calculadora, incluyendo:

- Gestión del estado de la expresión matemática

- Control del modo de ángulos (grados/radianes)

- Mantenimiento del historial de cálculos

**Interfaz Gráfica**

**Configuración Visual:**

- Ventana principal de 400x600 píxeles

- Esquema de colores oscuro (#2E2E2E) similar a calculadoras profesionales

- Campo de entrada de 360x60 píxeles con justificación a la derecha

- Botones organizados en cuadrícula con colores diferenciados

**Distribución de Botones:**

Los botones están organizados estratégicamente con funciones específicas:

- Fila 1: Funciones de modo y navegación

- Filas 2-4: Funciones científicas y trigonométricas

- Filas 5-8: Números y operaciones básicas

**Funciones Principales**

Función append():

Gestiona la entrada de caracteres a la expresión matemática actual.

Función calculate():

Implementa un sistema seguro de evaluación de expresiones utilizando:

- Diccionario de funciones seguras

- Evaluación restringida con `eval()` controlado

- Manejo de errores con try-except

Funciones Trigonométricas Personalizadas:

```python

def sin(x):

return math.sin(math.radians(x)) if self.deg\_mode else math.sin(x)

```

Estas funciones adaptan automáticamente los cálculos según el modo de ángulos seleccionado.

Gestión del Historial

La funcionalidad de historial incluye:

- Almacenamiento de cálculos en lista interna

- Ventana emergente para visualización

- Opción de limpieza del historial

- Formato legible "expresión = resultado"

**Validación y Manejo de Errores**

**Sistema de Seguridad:**

- Uso de diccionario restringido para `eval()`

- Eliminación de `\_\_builtins\_\_` para prevenir accesos no autorizados

- Captura de excepciones con mensaje de error genérico

**Tipos de Errores Manejados:**

- Errores de sintaxis en expresiones

- División por cero

- Valores fuera de dominio para funciones matemáticas

- Operaciones inválidas

**Pruebas y Resultados**

**Casos de Prueba Realizados**

**Operaciones Básicas:**

- 5 + 3 = 8

- 10 - 4 = 6

- 7 \* 6 = 42

- 15 / 3 = 5

**Operaciones Científicas:**

- sin(30°) = 0.5 (en modo grados)

- log(10) = 2.302585... (logaritmo natural)

- 2³ = 8

- √16 = 4

**Manejo de Errores:**

- División por cero: Muestra "Error"

- Expresiones inválidas: Muestra "Error"

- Funciones fuera de dominio: Muestra "Error"

**Comportamiento del Programa**

**Rendimiento:**

- Respuesta inmediata a clics de botones

- Cálculos instantáneos para operaciones simples

- Gestión eficiente de memoria para el historial

**Usabilidad:x**

- Interfaz intuitiva similar a calculadoras físicas

- Retroalimentación visual clara

- Funciones de corrección (backspace, clear)

**Funcionalidades Verificadas**

Interfaz Gráfica: Todos los botones responden correctamente

Cálculos Básicos: Operaciones aritméticas funcionan sin errores

Funciones Científicas: Trigonometría y logaritmos calculan correctamente

Modo de Ángulos: Conversión automática entre grados y radianes

Historial: Almacenamiento y visualización de cálculos previos

Manejo de Errores: Recuperación elegante de errores sin crash