

PROJECT, NO: 02

Scientific Calculator.

Bano Qabil 2.0

Date = 08 / 02 / 2024 .

Student Name :> Muhammad Ishaq .

Email = muhammadishaqpak801@gmail.com

GitHub Link = <https://github.com/muhammadiishaq/BanoQabil-2.0-Python-Course>

Scientific Calculator Overview .:

This Python-based scientific calculator, developed by Muhammad Ishaq, offers a comprehensive set of mathematical functions in a command-line interface. Users can perform basic arithmetic operations like addition, subtraction, multiplication, and division, alongside advanced functions such as exponentiation, square root calculation, and trigonometric operations like sine, cosine, and tangent. The calculator includes error handling for division by zero and invalid input. Its interactive design allows for easy navigation, and users can gracefully exit the program with a simple "OFF" command. With its user-friendly interface and versatile functionality, the calculator serves as a handy tool for both educational and practical mathematical tasks.

Explanation of the code and logic design :

- **Creator:**

Developed by Muhammad Ishaq.

- **User Interface :**

The code begins with importing the **math** library and displaying initial information about the creator of the calculator.

It presents a list of available operations to the user, indicating the options for performing arithmetic calculations and various scientific functions

- **Functionality Overview :**

The code defines separate functions for each mathematical operation supported by the calculator, such as addition, subtraction, multiplication, division, exponentiation, square root, angle conversion, and trigonometric functions (sine, cosine, tangent).

Each function is accompanied by a docstring that provides a description of the function, its parameters, and its return value, enhancing code readability and maintainability.

```
Created By : Muhammad Ishaq "
***** SCIENTIFIC CALCULATOR *****

"Welcome to Simple Scientific Calculator"

Choice Operations:

1. Addition (+)
2. Subtraction (-)
3. Multiplication (x)
4. Division (/)
5. Power (^)
6. Square Root (√)
7. Degree (deg)
8. Radian (rad)
9. Sine (sin)
10. Cosine (cos)
11. Tangent (tan)

Type On or Off for using calculator :
```

- **User Interaction Loop :**

The code enters a continuous loop where users can input their choice of operation and operands.

It prompts users to turn the calculator "ON" or "OFF" to control its functionality.

Within the loop, the code validates user input, ensuring it falls within the specified range of available operations.

- **Operation Execution :**

Depending on the user's selection, the code executes the corresponding function to perform the desired mathematical operation.

For arithmetic operations, the code takes two operands as input and returns the result.

For scientific functions, the code accepts a single operand and calculates the result accordingly.

- **Error Handling :**

The code incorporates error handling mechanisms to address potential issues, such as division by zero and invalid user input.

It provides informative error messages to guide users and prevent unexpected program termination.

```
Type On or Off for using calculator :on
Enter your Operators number upto (1-11) :5
Enter your First number :46
Enter your second number :dfsg

Error! : Enter inputs in digits!.

Type On or Off for using calculator :
```

- **Calculator Open or Shutdown :**

Users can choose to turn on the calculator by entering "on, ON" when prompted.
Users can choose to turn off the calculator by entering "of, OFF" when prompted.
Upon shutdown, the code displays a farewell message before exiting the loop and terminating the program.

```
Type On or Off for using calculator :off
```

```
👉 Thanks for using calculator.
```

```
👋 Bye Bye.
```

```
Process finished with exit code 0
```

• Summary :

In summary, the code implements a scientific calculator with a user-friendly interface, comprehensive functionality, and robust error handling, providing an efficient tool for performing various mathematical tasks.

• Picture Of The Project :

```
try:
    choice = int(input("Enter your Operators number upto (1-11) :"))

    # If choice are this range after enter this block
    if choice in [1, 2, 3, 4, 5]:
        # user input select number.
        x = float(input("Enter your First number :"))
        y = float(input("Enter your second number :"))
        # Condition for select operation
        if choice == 1:
            result = add(x, y)
        elif choice == 2:
            result = sub(x, y)
        elif choice == 3:
            result = multi(x, y)
        elif choice == 4:
            result = divide(x, y)
        elif choice == 5:
            result = power(x, y)
        print("Answer: ", result)

    # List for select operation.
    elif choice in [6, 7, 8, 9, 10, 11]:
        x = float(input("Enter your number :"))
        if choice == 6:
            if x <= 0:
                print("Error! ⚠️ : Cannot calculate square root of a negative number.")
            result = square_root(x)
        elif choice == 7:
            result = degree(x)
        elif choice == 8:
            result = radians(x)
        elif choice == 9:
            result = sin(x)
        elif choice == 10:
            result = cos(x)
        elif choice == 11:
            result = tan(x)

# Student Name = Muhammad Ishaq
# RollNo = 8
# Email = muhammadishaqpak01@gmail.com

# Python Library Maths.
import math

print()
print("Created By -> 🐧 \ Muhammad Ishaq \""\n")
print("***** SCIENTIFIC CALCULATOR ***** \n")
print("❤️ Welcome to Simple Scientific Calculator ❤️ \n")

# Select operators
print("\nChoice Operations:\n", '\n' * 1, "1. Addition (+)", '\n' * 2, "2. Subtraction (-)",
      '\n' * 3, "3. Multiplication (*)", '\n' * 4, "4. Division (/)", '\n' * 5, "5. Power (^)",
      '\n' * 6, "6. Square Root (v)", '\n' * 7, "7. Degree (deg)", '\n' * 8, "8. Radian (rad)",
      '\n' * 9, "9. Sine (sin)", '\n' * 10, "10. Cosine (cos)", '\n' * 11, "11. Tangent (tan)")

# Adding function
def add(x, y):
    """
    Add two numbers.

    Parameters:
    x (float): The first number.
    y (float): The second number.

    Returns:
    float: The sum of x and y.
    """
    return x + y

# Subtraction function
def sub(x, y):
    """
    Subtract two numbers.

    Parameters:
    x (float): The first number.
    y (float): The second number.

    Returns:
    float: The difference of x and y.
    """
    return x - y

def square_root(x):
    """
    Calculate the square root of x.

    Parameters:
    x (float): The number to calculate the square root of.

    Returns:
    float: The square root of x.
    """
    return math.sqrt(x)

def degree(x):
    """
    Convert degrees to radians.

    Parameters:
    x (float): The angle in degrees.

    Returns:
    float: The angle in radians.
    """
    return math.radians(x)

def radians(x):
    """
    Convert radians to degrees.

    Parameters:
    x (float): The angle in radians.

    Returns:
    float: The angle in degrees.
    """
    return math.degrees(x)

def sin(x):
    """
    Calculate the sine of x.

    Parameters:
    x (float): The angle in radians.

    Returns:
    float: The sine of x.
    """
    return math.sin(x)

def cos(x):
    """
    Calculate the cosine of x.

    Parameters:
    x (float): The angle in radians.

    Returns:
    float: The cosine of x.
    """
    return math.cos(x)

def tan(x):
    """
    Calculate the tangent of x.

    Parameters:
    x (float): The angle in radians.

    Returns:
    float: The tangent of x.
    """
    return math.tan(x)

Created By -> 🐧 * Muhammad Ishaq *

***** SCIENTIFIC CALCULATOR *****

❤️ Welcome to Simple Scientific Calculator ❤️

Choice Operations:

1. Addition (+)
2. Subtraction (-)
3. Multiplication (*)
4. Division (/)
5. Power (^)
6. Square Root (v)
7. Degree (deg)
8. Radian (rad)
9. Sine (sin)
10. Cosine (cos)
11. Tangent (tan)

Type On or Off for using calculator :on
Enter your Operators number upto (1-11) :1
Enter your First number :45
Enter your second number :5
Answer:> 50.0

Type On or Off for using calculator :off

👉 Thanks for using calculator.
👋 Bye Bye.

Process finished with exit code 0
```

