PYTHON IS FOR EVERYONE

Tutorial 3:

PYTHON PROGRAMMING - BASIC OPERATIONS IN GOOGLE COLAB

Jeff Gentry

@www.linkedin.com/in/jefferycharlesgentry

Objectives

- Understand basic arithmetic operations in Python.
- Learn about operator precedence.
- Practice performing operations with variables.

Arithmetic Operations

Python supports several basic arithmetic operations:

1. **Addition (+)**
2. **Subtraction (-)**
3. **Multiplication (*)**
4. **Division (/)**
5. **Floor Division (//): Divides and returns the largest integer less than or equal to the result.**
6. **Modulus (%): Returns the remainder of a division.**

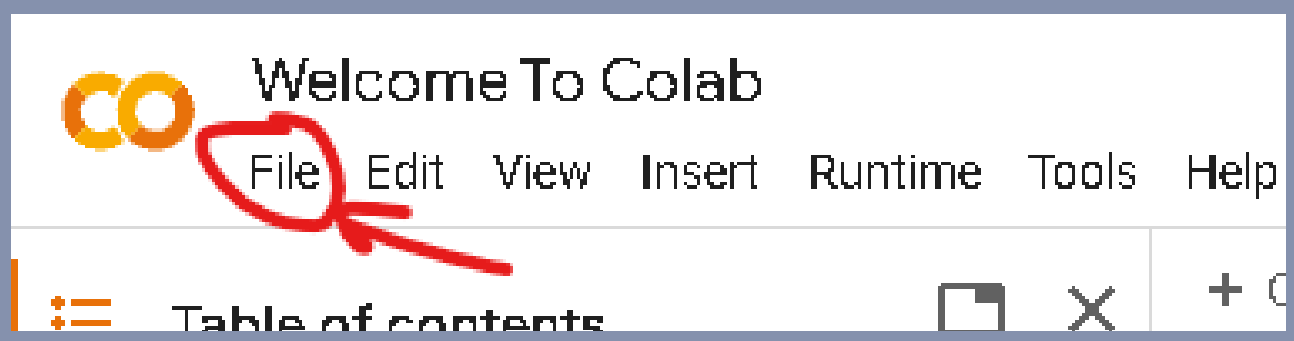
Exponentiation (**): Raises a number to the power of another.

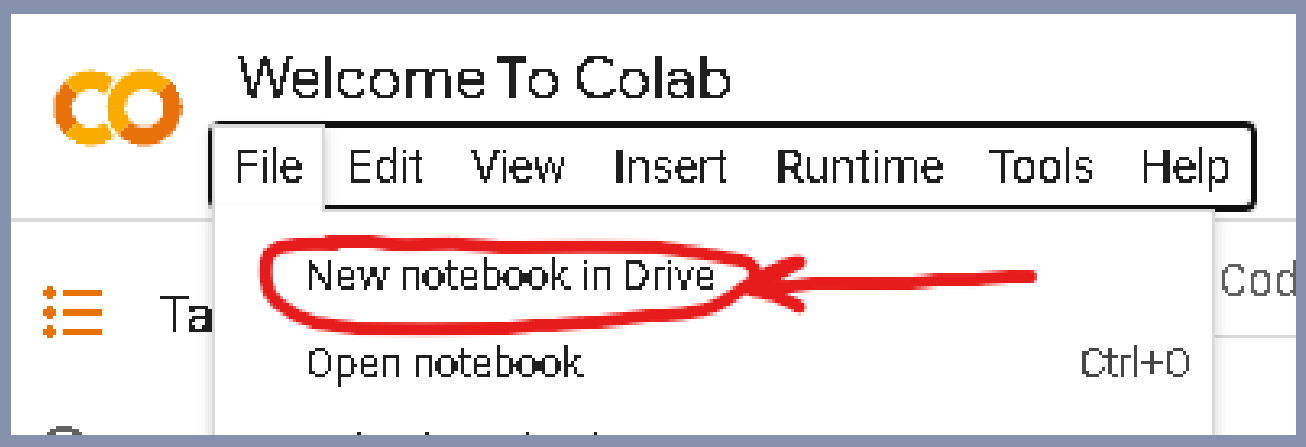
Performing Basic Operations

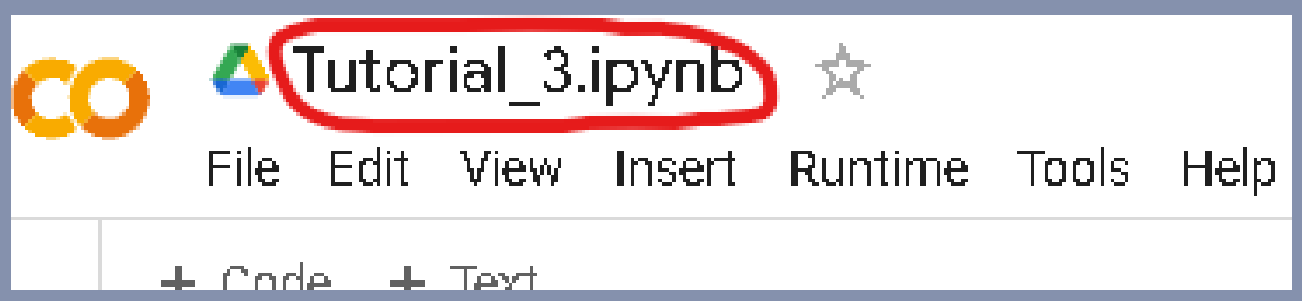
Let’s perform some basic arithmetic operations in Google Colab.

- Define a few variables
- Addition
- Subtraction
- Multiplication
- Division
- Floor Division
- Modulus
- Exponentiation

Create a New Notebook







Write Python Code

Add Code Cell:

- You should see a code cell (a box with In [ ]:) where you can write your code.
- If you don’t see a code cell, click on + Code in the toolbar.

Basic Operations

Type the Following Code:

```python
# Define two numbers
a = 10
b = 3

# Addition
sum_result = a + b
print("Sum:", sum_result)

# Subtraction
difference = a - b
print("Difference:", difference)

# Multiplication
product = a * b
print("Product:", product)

# Division
quotient = a / b
print("Quotient:", quotient)

# Floor Division
floor_division = a // b
print("Floor Division:", floor_division)

# Modulus
remainder = a % b
print("Remainder:", remainder)

# Exponentiation
power = a ** b
print("Power (a^b):", power)
```

Basic Operations Results

```
Sum: 13
Difference: 7
Product: 30
Quotient: 3.3333333333333335
Floor Division: 3
Remainder: 1
Power (a^b): 1000
```

Operator Precedence

Operator precedence determines the order in which operations are performed in an expression. In Python, the order of operations is as follows (from highest to lowest precedence):

1. Parentheses ()
2. Exponentiation **
3. Multiplication *, Division /, Floor Division //, Modulus %
4. Addition +, Subtraction -

You can use parentheses to change the order of operations.

Precedence Examples

```python
# Without parentheses
result1 = 10 + 2 * 3
print("Without parentheses:", result1)  # Output: 16

# With parentheses
result2 = (10 + 2) * 3
print("With parentheses:", result2)  # Output: 36
```

```
Without parentheses: 16
With parentheses: 36
```

Practice Exercises

1. **Basic Calculations: Create two variables, x and y, and perform all basic arithmetic operations on them. Print the results.**
2. **Calculate Area: Write a program that calculates the area of a rectangle. Use variables for length and width, and print the area.**
3. **Order of Operations: Create an expression that uses all types of operations and parentheses. Print the result to see how operator precedence affects the outcome.**

Example Program

```python
# Program to calculate the area of a rectangle

# Get user input for length and width
length = float(input("Enter the length of the rectangle: "))
width = float(input("Enter the width of the rectangle: "))

# Calculate the area
area = length * width

# Print the area
print(f"The area of the rectangle is: {area}")
```

```
Enter the length of the rectangle: 5
Enter the width of the rectangle: 6
The area of the rectangle is: 30.0
```

Conclusion

In this tutorial, you learned about basic arithmetic operations in Python and how to use them with variables. You also explored operator precedence and how it affects calculations. Mastering these operations is essential for performing more complex calculations in your programs.

Next Steps

In tutorial 4, we can explore control structures by using conditional statements, create decision making programs and then practice writing conditional statements.

FOLLOW ME

for more tips you didn’t know you needed

Jeff Gentry

@www.linkedin.com/in/jefferycharlesgentry