**Aim:** Write a program to perform different arithmetic operations on numbers in python.

**IDE:**

Arithmetic operations are fundamental to programming, and Python provides straightforward operators to perform these calculations. Let's revisit these basic arithmetic operations, which you've likely encountered in your math classes, and see how they can be used in Python.

**Types of Arithmetic Operators in Python**

Arithmetic operators in Python are fundamental tools used for performing basic mathematical operations. Here are the primary types of arithmetic operators:

* Addition
* Subtraction
* Multiplication
* Division
* Modulus
* Exponentiation
* Floor Division

Let's take a closer look at each of these operators to understand them better.

**Addition**

The addition operator in Python is “+”. It is used to add or sum two values.

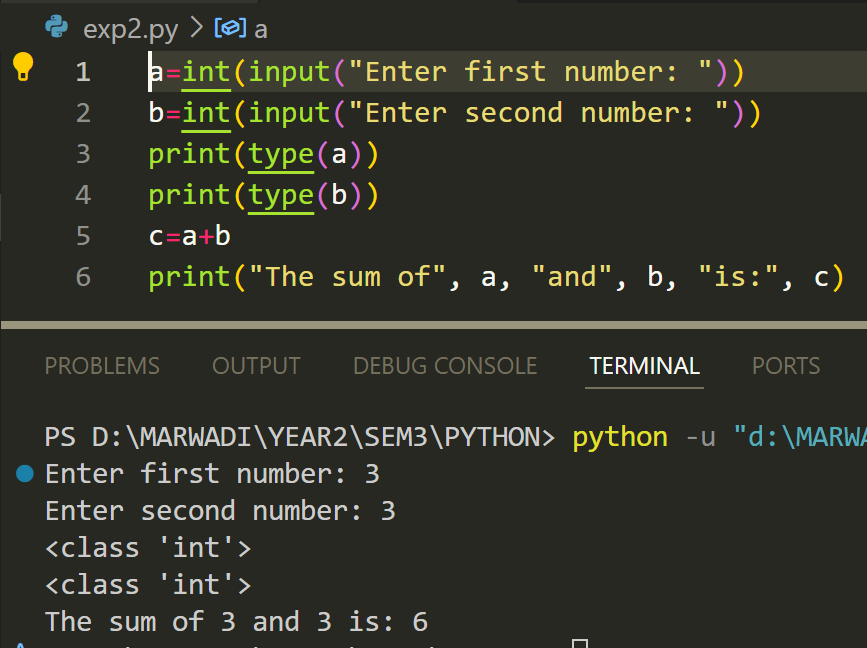
**Python Code:**

num1, num2 = 10, 30

sum= num1+num2

print("The sum of",num1,"and",num2,"is:",sum)

**Output:**

****

**Subtraction**

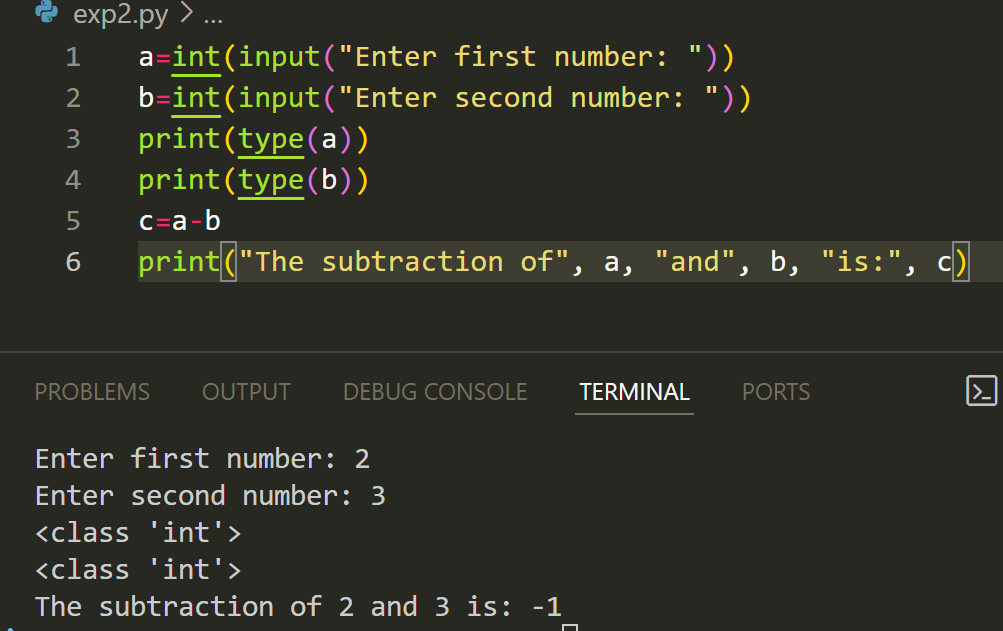
The subtraction operator in Python is “-”. It is used to subtraction or difference two values.

num1, num2 = 10, 30

sub= num1-num2

print("The subtraction of",num1,"and",num2,"is:",sub)

output:



**Multiplication**

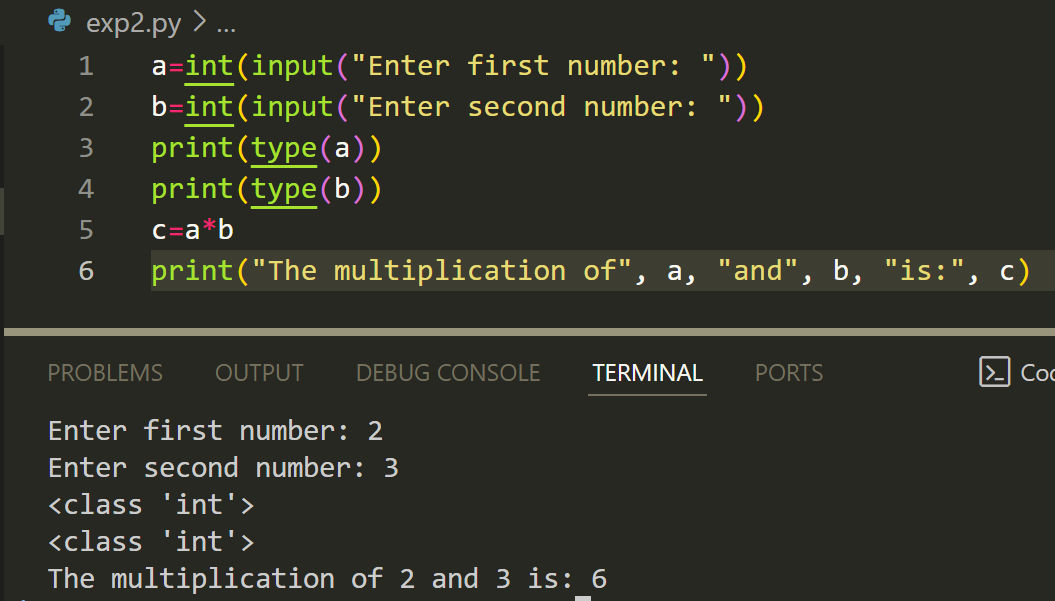
The Arithmetic Operator in Python for multiplication is “\*”. With this operator, we can find the product of two values.

num1, num2 = 10, 30

product= num1\*num2

print("The product of",num1,"and",num2,"is:",product)

Output:



**Division**

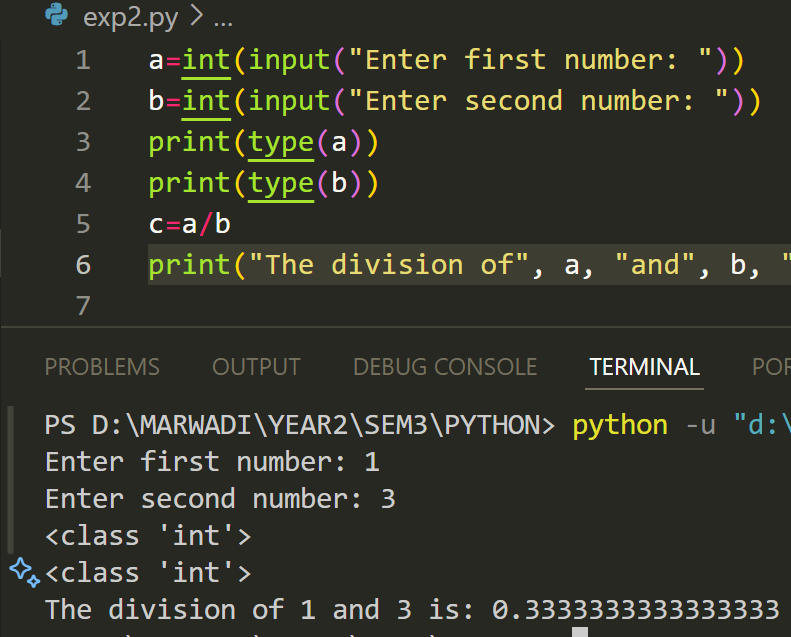
The “/” operator is the division operator in Python. We can find the quotient when the first operand is divided by the second.

num1, num2 = 10, 30

div = num1/num2

print("The division of",num1,"and",num2,"is:",div)

Output:



**Modulus**

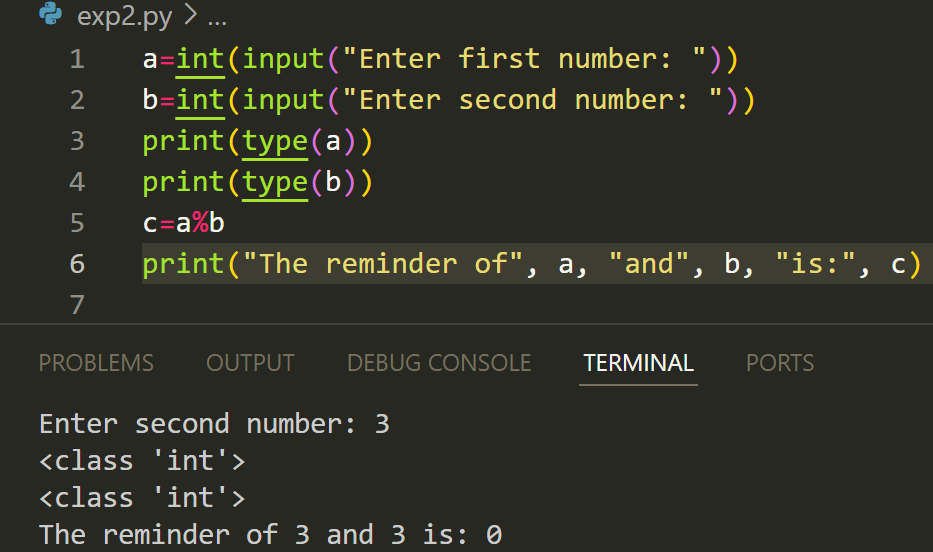
The “%” operator is the division operator in Python. Using this, we can find the remainder when the first operand is divided by the second.

num1, num2 = 10, 30

rem = num1%num2

print("The reminder of",num1,"and",num2,"is:",rem)

output:



**Exponentiation**

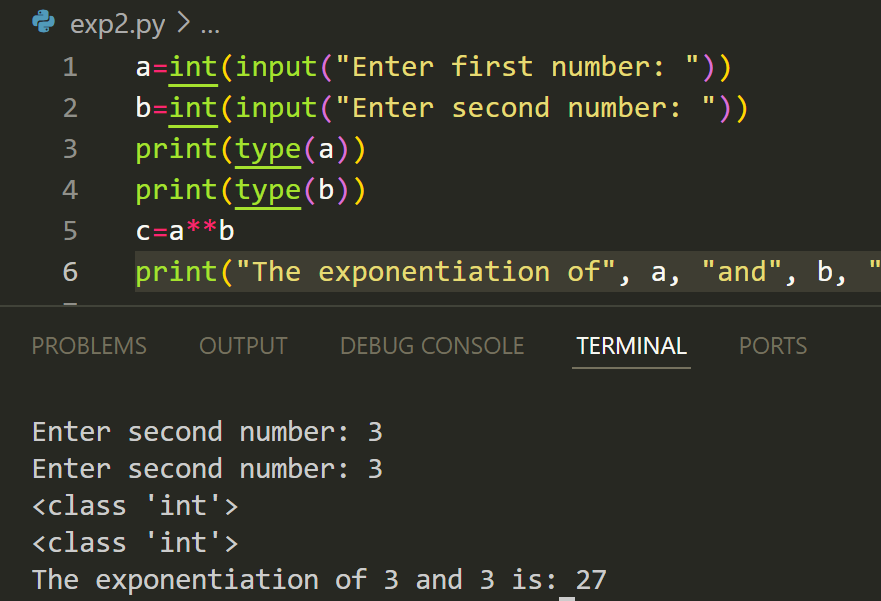
The exponentiation operator in Python is denoted by “\*\*”. It is used to raise the power of the first operand to the power of the second.

num1, num2 = 10, 3

exp = num1\*\*num2

print("The exponentiation of",num1,"and",num2,"is:",exp)

Output:



**Floor Division**

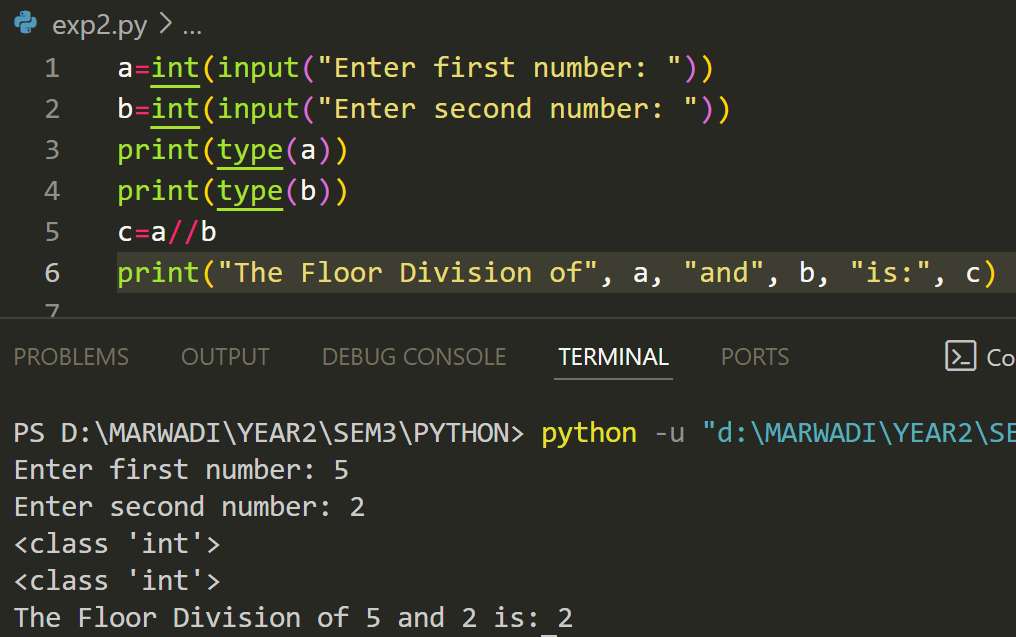
It is denoted by “//” in Python. We use it to find the floor of the quotient when the first operand is divided by the second.

num1, num2 = 10, 3

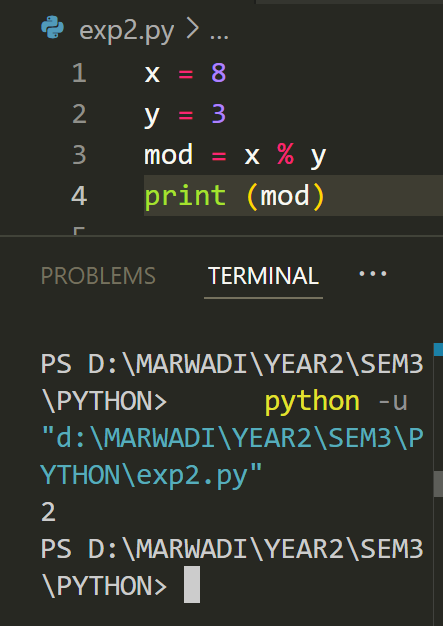
floordiv = num1//num2

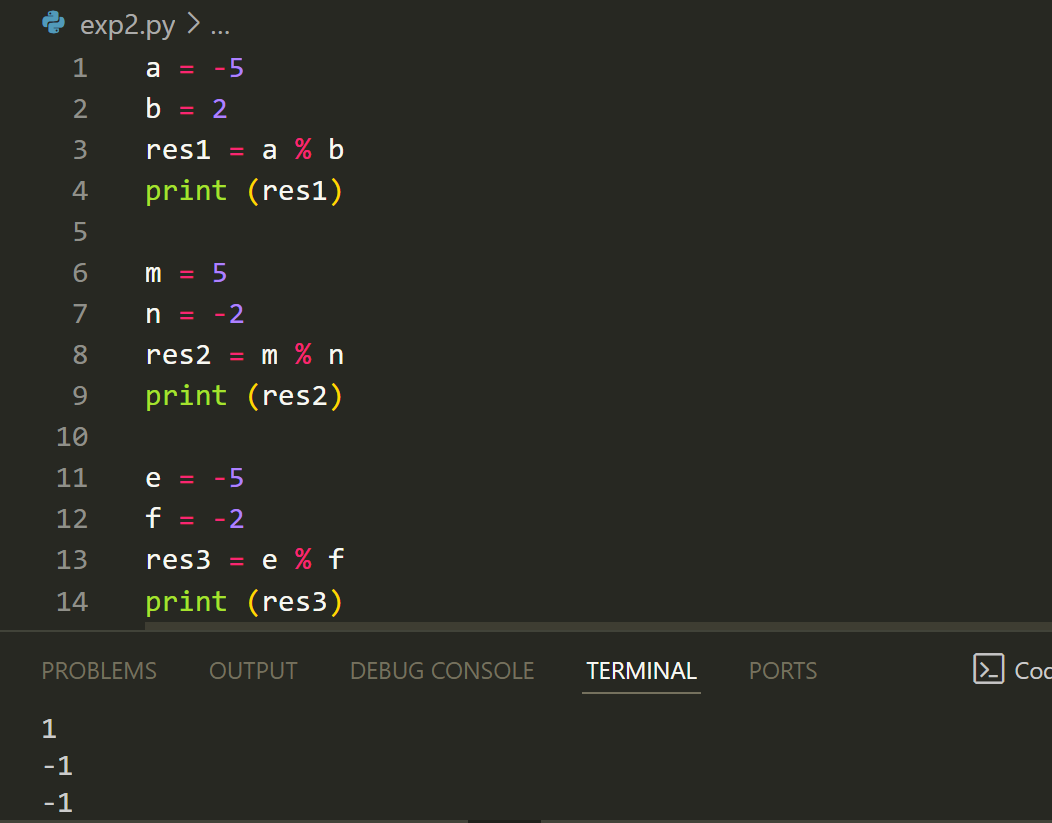
print("The Floor Division of",num1,"and",num2,"is:",floordiv)

Output:



**Task:**





**Order of precedence of Arithmetic operators in Python**

Arithmetic Operators in Python follow a basic order of precedence. When more than one operator is used, they are executed according to this order:

Operator Purpose

() Parentheses

\*\* Exponent

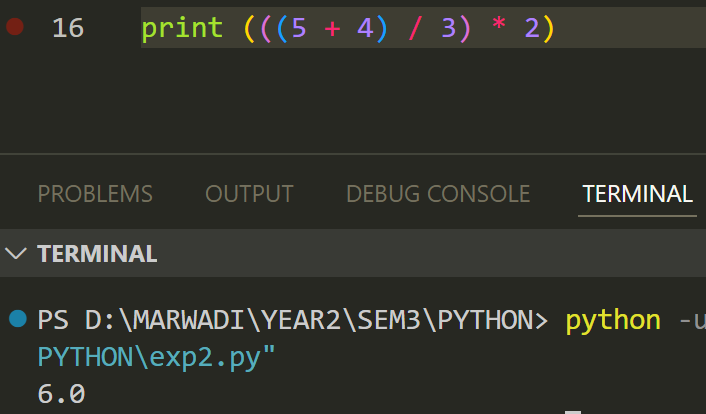
%, \*, /, // Modulos, Multiplication, Division and Floor division

+, - Addition and Subtraction

The operator listed at the top of the table will be executed first.

print (((5 + 4) / 3) \* 2)

Output



x = 3

y = 4

z = 6

print(x\*y//z)

print(x\*(y//z))

Output:

A screen shot of a computer program

AI-generated content may be incorrect.

x = 2

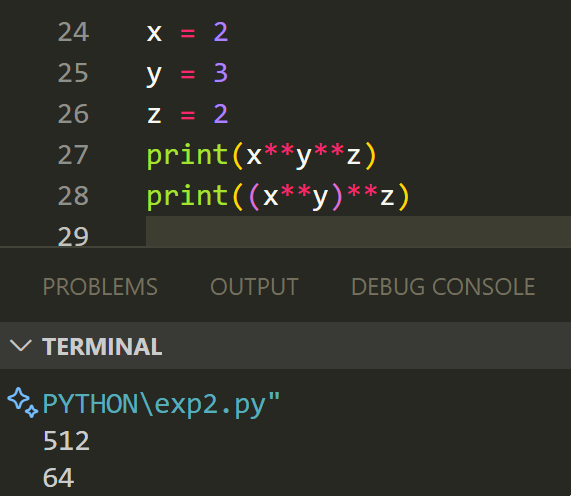
y = 3

z = 2

print(x\*\*y\*\*z)

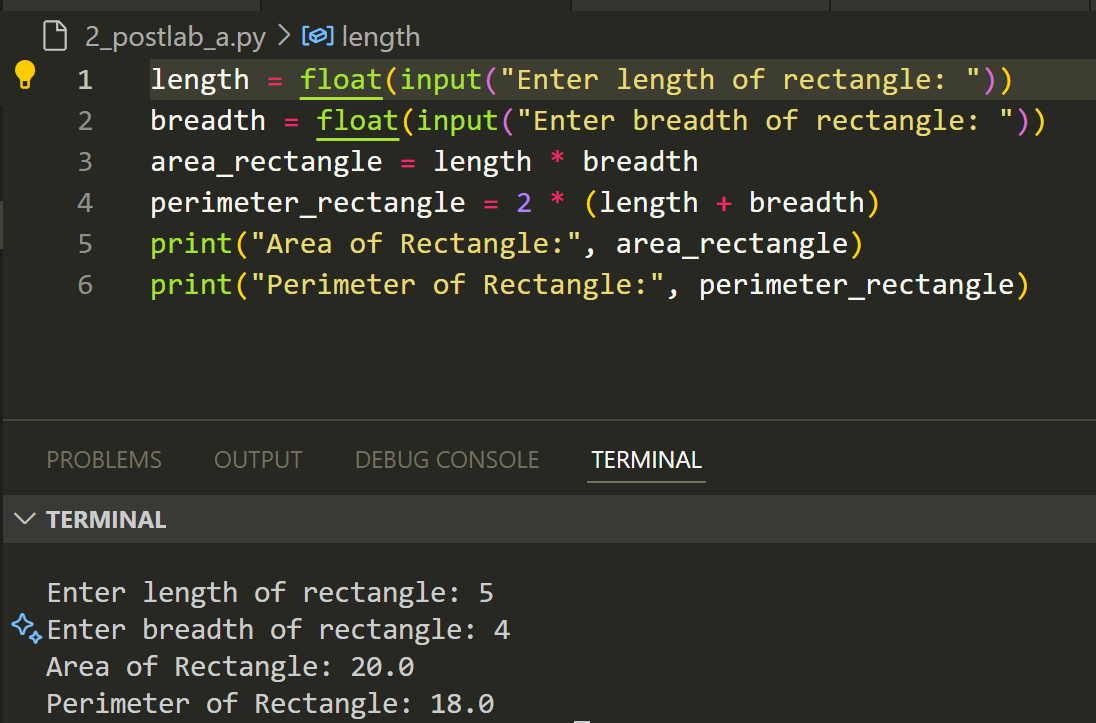
print((x\*\*y)\*\*z)

Output

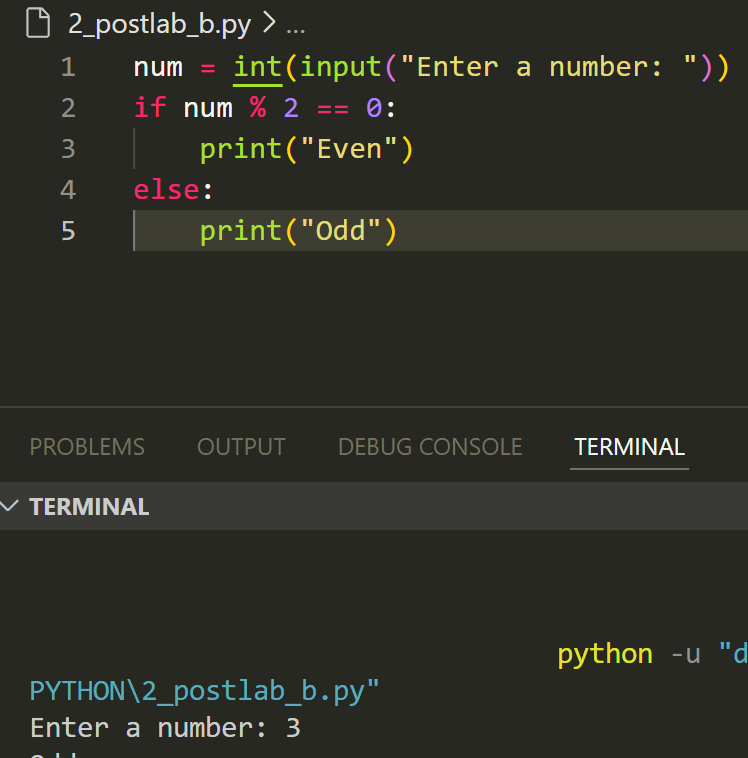


**Post Lab**

Write a python code for calculating the Area and Perimeter of a Rectangle



Write a python code for testing if a number is even or odd



Write a python code for calculate the area and volume of the Cube.

A screenshot of a computer program

AI-generated content may be incorrect.

Write a python code to solve the equation z = (x+y)\*(x-y)

A screenshot of a computer program

AI-generated content may be incorrect.

Write a python code to solve the equation z = (x+y)\*(x+y)-2xy; write a comment on it.

A screenshot of a computer program

AI-generated content may be incorrect.

Write a python code for Converting Celsius to Fahrenhit

A screen shot of a computer program

AI-generated content may be incorrect.