Operators

**Q.1 Write a program for arithmetic operators.**

**Code:**

# Taking two numbers as input from the user

num1 = float(input("Enter the first number: "))

num2 = float(input("Enter the second number: "))

# Performing arithmetic operations

addition = num1 + num2

subtraction = num1 - num2

multiplication = num1 \* num2

division = num1 / num2 # Normal division

floor\_division = num1 // num2 # Floor division

modulus = num1 % num2 # Remainder

exponentiation = num1 \*\* num2 # num1 raised to the power of num2

# Displaying the results

print("\n--- Arithmetic Operations ---")

print(f"Addition: {num1} + {num2} = {addition}")

print(f"Subtraction: {num1} - {num2} = {subtraction}")

print(f"Multiplication: {num1} \* {num2} = {multiplication}")

print(f"Division: {num1} / {num2} = {division}")

print(f"Floor Division: {num1} // {num2} = {floor\_division}")

print(f"Modulus: {num1} % {num2} = {modulus}")

print(f"Exponentiation: {num1} \*\* {num2} = {exponentiation}")

**Input:**

Enter the first number: 10

Enter the second number: 3

**Output:**

--- Arithmetic Operations ---

Addition: 10.0 + 3.0 = 13.0

Subtraction: 10.0 - 3.0 = 7.0

Multiplication: 10.0 \* 3.0 = 30.0

Division: 10.0 / 3.0 = 3.3333333333333335

Floor Division: 10.0 // 3.0 = 3.0

Modulus: 10.0 % 3.0 = 1.0

Exponentiation: 10.0 \*\* 3.0 = 1000.0

**Explanation:**

1. **Addition (+)**: Adds two numbers.

2. **Subtraction (-)**: Subtracts the second number from the first.

3. **Multiplication (\*)**: Multiplies two numbers.

4. **Division (/)**: Performs normal division and gives a float result.

5. **Floor Division (//)**: Performs division and rounds down to the nearest whole number.

6. **Modulus (%)**: Finds the remainder of the division

7. **Exponentiation (\*\*)**: Raises the first number to the power of the second.

**Q.2 Write a program for assignment operators.**

**Code:**

# Initializing a variable

num = 10

print(f"Initial value of num: {num}")

# Using assignment operators

num += 5 # Add and assign

print(f"After num += 5: {num}")

num -= 3 # Subtract and assign

print(f"After num -= 3: {num}")

num \*= 2 # Multiply and assign

print(f"After num \*= 2: {num}")

num /= 4 # Divide and assign

print(f"After num /= 4: {num}")

num %= 3 # Modulus and assign

print(f"After num %= 3: {num}")

num \*\*= 2 # Exponentiation and assign

print(f"After num \*\*= 2: {num}")

num //= 3 # Floor division and assign

print(f"After num //= 3: {num}")

**Output:**

Initial value of num: 10

After num += 5: 15

After num -= 3: 12

After num \*= 2: 24

After num /= 4: 6.0

After num %= 3: 0.0

After num \*\*= 2: 0.0

After num //= 3: 0.0

**Explanation:**

1. **+=**: Adds the value on the right to the variable and assigns the result.

2. **-=**: Subtracts the value on the right from the variable and assigns the result.

3. **\*=**: Multiplies the variable by the value on the right and assigns the result.

4. **/=**: Divides the variable by the value on the right and assigns the result.

5. **%=**: Takes the modulus of the variable with the value on the right and assigns the result.

6. **\*\*=**: Raises the variable to the power of the value on the right and assigns the result.

7. **//=**: Performs floor division and assigns the result.

**Q.3Write a program for Bitwise operators.**

**Code:**

# Initializing two numbers

a = 5 # Binary: 0101

b = 3 # Binary: 0011

# Bitwise AND

result\_and = a & b # Binary: 0101 & 0011 = 0001

print(f"{a} & {b} = {result\_and} (Bitwise AND)")

# Bitwise OR

result\_or = a | b # Binary: 0101 | 0011 = 0111

print(f"{a} | {b} = {result\_or} (Bitwise OR)")

# Bitwise XOR

result\_xor = a ^ b # Binary: 0101 ^ 0011 = 0110

print(f"{a} ^ {b} = {result\_xor} (Bitwise XOR)")

# Bitwise NOT

result\_not\_a = ~a # Binary: ~0101 = -(0101 + 1) = -6

print(f"~{a} = {result\_not\_a} (Bitwise NOT)")

# Left Shift

result\_left\_shift = a << 2 # Binary: 0101 << 2 = 10100 (Shift left by 2 bits)

print(f"{a} << 2 = {result\_left\_shift} (Left Shift)")

# Right Shift

result\_right\_shift = a >> 2 # Binary: 0101 >> 2 = 0001 (Shift right by 2 bits)

print(f"{a} >> 2 = {result\_right\_shift} (Right Shift)")

**Output:**

5 & 3 = 1 (Bitwise AND)

5 | 3 = 7 (Bitwise OR)

5 ^ 3 = 6 (Bitwise XOR)

~5 = -6 (Bitwise NOT)

5 << 2 = 20 (Left Shift)

5 >> 2 = 1 (Right Shift)

**Explanation:**

**1. & (AND)**: Sets a bit to 1 if both corresponding bits are 1.

**2.| (OR)**: Sets a bit to 1 if at least one corresponding bit is 1.

**3.^ (XOR)**: Sets a bit to 1 if the corresponding bits are different.

**4.~ (NOT)**: Inverts all bits (1 becomes 0, and 0 becomes 1). Negative due to two's complement representation.

**5.<< (Left Shift)**: Shifts the bits to the left by the specified number of positions, adding 0s on the right.

**6.>> (Right Shift)**: Shifts the bits to the right by the specified number of positions, discarding bits on the right.

**Q.4 Write a program to calculate greatest of three numbers.**

**Code:**

# Input: Taking three numbers from the user

num1 = float(input("Enter the first number: "))

num2 = float(input("Enter the second number: "))

num3 = float(input("Enter the third number: "))

# Determine the greatest number

if num1 >= num2 and num1 >= num3:

greatest = num1

elif num2 >= num1 and num2 >= num3:

greatest = num2

else:

greatest = num3

# Output: Display the greatest number

print(f"The greatest number among {num1}, {num2}, and {num3} is {greatest}.")

**Input:**

Enter the first number: 12

Enter the second number: 45

Enter the third number: 30

**Output:**

The greatest number among 12.0, 45.0, and 30.0 is 45.0.

**Explanation:**

1. **if condition**: Checks whether the first number is greater than or equal to the other two.

2. **elif condition**: Checks whether the second number is greater than or equal to the other two

3. **else**: If neither of the above conditions is true, the third number is the greatest.

**Q.5 Calculate the area of a circle.**

**Code:**

import math # Import the math module to use the constant pi

# Input: Taking the radius from the user

radius = float(input("Enter the radius of the circle: "))

# Calculate the area of the circle

area = math.pi \* radius \*\* 2

# Output: Display the area

print(f"The area of the circle with radius {radius} is {area:.2f}.")

Input:

Enter the radius of the circle: 5

**Output:**

The area of the circle with radius 5.0 is 78.54.

**Explanation:**

**1.math.pi**: Provides the value of π\piπ to ensure the calculation is accurate.

**2.radius \*\* 2**: Squares the radius to calculate the area.

**3.:.2f**: Formats the output to display the area rounded to 2 decimal places.

**Q.6 Calculate the area of a triangle**

**Code:**

# Input: Taking the base and height from the user

base = float(input("Enter the base of the triangle: "))

height = float(input("Enter the height of the triangle: "))

# Calculate the area of the triangle

area = 0.5 \* base \* height

# Output: Display the area

print(f"The area of the triangle with base {base} and height {height} is {area:.2f}.")

**Input:**

Enter the base of the triangle: 10

Enter the height of the triangle: 5

**Output:**

The area of the triangle with base 10.0 and height 5.0 is 25.00.

**Explanation:**

1. **Formula**: The area of the triangle is calculated by multiplying the base by the height and dividing by 2.

2. **Formatting**: The area is displayed with 2 decimal points for precision.

**Q.7 Calculate the area of a rectangle**.

**Code:**

def calculate\_area(length, width):

return length \* width

length = float(input("Enter the length of the rectangle: "))

width = float(input("Enter the width of the rectangle: "))

area = calculate\_area(length, width)

print(f"The area of the rectangle is: {area} square units")

**Input:**

Enter the length of the rectangle: 5

Enter the width of the rectangle: 3

**Output:**

The area of the rectangle is: 15.0 square units

**Q.8 Calculate the area of a square.**

**Code:**

# Function to calculate the area of a square

def calculate\_area(side):

return side \*\* 2 # The formula for the area of a square is side^2

# Taking user input for side length

side = float(input("Enter the side length of the square: ")) # Converts input to float for precision

# Calling the function to calculate the area

area = calculate\_area(side)

# Displaying the result

print(f"The area of the square is: {area} square units")

**Input:**

Enter the side length of the square: 4

**Output:**

The area of the square is: 16.0 square units