Python Operators

Assignment Questions

pw skills

1. Calculate the sum, difference, product, and quotient of two numbers.

```
# Define the numbers

num1 = 10

num2 = 5

# Calculate the sum

sum_result = num1 + num2

# Calculate the difference

difference_result = num1 - num2
```

```
# Calculate the product
product_result = num1 * num2
# Calculate the quotient
quotient_result = num1 / num2


# Display the result
print("The sum of",num1, "and", num2, "is:",
sum_result)
print(“the difference b\w ", num1, "and", num2"is:",
difference_result)
print("The product of", num1, "and", num2, "is:",
product_result)
print("The quotient of", num1, "and", num2, "is:",
quotient_result)
```

2. Perform various assignment operations on a variable.

```
# Define a variable

x = 10

# Perform various assignment operations

x += 5   #(addition assignment)

print("After addition:", x)

 # Output: 15

x -= 3  # (subtraction assignment)

print("After subtraction:", x) # Output: 12

x *= 2   # (multiplication assignment)

print("After multiplication:", x)

# Output: 24

x /= 4   # (division assignment)

print("After division:", x)

# Output: 6.0

x //= 2  # (floor division assignment)

print("After floor division:", x)
```

```
# Output: 3.0

x **= 2   # (exponentiation assignment)

print("After exponentiation:", x)

# Output: 9.0

x %= 5  #(modulus assignment)

print("After modulus operation:", x)

# Output: 4.0

x &= 1   # (bitwise AND assignment)

print("After bitwise AND operation:", x)

 # Output: 0

x |= 2   #(bitwise OR assignment)

print("After bitwise OR operation:", x)

# Output: 2

x ^= 1   # (bitwise XOR assignment)

print("After bitwise XOR operation:", x)

# Output: 3
```

```
x <<= 2   # (bitwise left shift assignment)

print("After bitwise left shift operation:", x)

 # Output: 12

x >>= 1   #(bitwise right shift assignment)

print("After bitwise right shift operation:", x)

 # Output: 6
```

3. Compare two numbers and print the results.

```
# Define the numbers

num1 = 15

num2 = 10

# Compare the numbers

if num1 == num2:
    print(num1, "is equal to", num2)

elif num1 != num2:

    print(num1, "is not equal to", num2)

if num1 < num2:

    print(num1, "is less than", num2)

elif num1 > num2:

    print(num1, "is greater than", num2)

if num1 <= num2:

    print(num1, "is less than or equal to", num2)

elif num1 >= num2:

    print(num1, "is greater than or equal to", num2)
```

4. Check conditions using logical operators.

```
# Define the variables
x = 10
y = 15
z = 20

# Check conditions using logical operators
```

```
if x < y and y < z:
    print("Both conditions are true: x < y and y < z")

if x < y or x < z:
    print("At least one condition is true: x < y or x < z")

if not(x > z):
    print("The condition x > z is not true")
```

5. Check the identity of variables.

```
# Define variables

a = [1, 2, 3]

b = [1, 2, 3]

c = a

# Check identity using is operator

print("a is b:", a is b)

 # Output: False

print("a is c:", a is c)
```

```
# Output: True
# Check identity using is not operator
print("a is not b:", a is not b)
# Output: True
print("a is not c:", a is not c)
# Output: False (same memory location)
```

6.Perform bitwise operations and any two integers.

```
# Define two integers
num1 = 10
num2 = 7
# Bitwise AND
result_and = num1 & num2
print("Bitwise AND:", result_and)
# Bitwise OR
result_or = num1 | num2
print("Bitwise OR:", result_or)
```

```
# Bitwise XOR
result_xor = num1 ^ num2
print("Bitwise XOR:", result_xor)
# Bitwise NOT (Unary)
result_not_num1 = ~num1
print("Bitwise NOT of num1:", result_not_num1)


# Bitwise LEFT SHIFT
result_left_shift = num1 << 1
print("Bitwise LEFT SHIFT of num1 by 1:", result_left_shift)
# Bitwise RIGHT SHIFT
result_right_shift = num1 >> 1
print("Bitwise RIGHT SHIFT of num1 by 1:", result_right_shift)
```

7. Use unary operators to change the sign of a number.

```
# Define a number

num = 10

# Change the sign using unary negation

operator

num = -num

# Print the result

print("Number after changing sign:", num)
```

8. Use the ternary operator to assign values based on conditions.

```
# Define variables
```

```
x = 10 , y = 20

# Assign value based on condition

using ternary operator

result = "x is greater than y" if x > y else

"x is less than or equal to y"

# Display the result

print(result)
```

Data Science With Generative AI Course