Practical No. 3 : Write simple Python program using operators: Arithmetic Operators, Logical Operators, Bitwise Operators

• Practical related questions

1. Describe ternary operator in Python.

➔ The ternary operator in Python is a conditional expression that allows you to evaluate something in a single line instead of using a full `if-else` statement.

Syntax:
<value_if_true> if <condition> else <value_if_false>

Example:
```
a = 10
b = 20
max_value = a if a > b else b
print("Maximum value is:", max_value)
```

Here, If the condition `a > b` is True, `a` is assigned to `max_value`.
If False, `b` is assigned.

Output:
Maximum value is: 20

2. Describe about different Bitwise operators in Python with appropriate examples.

➔ Bitwise operators are used to perform bit-level operations on integers. They operate on bits and perform bit-by-bit operations.
Here, are the bitwise operators that are used in python:

1) **& – Bitwise AND**

The bitwise AND operator performs a logical AND operation on each bit of two numbers. It returns 1 only if both bits are 1, otherwise it returns 0.

Example:

```
a = 10
b = 4
print("a & b =", a & b)
```

Output:

```
a & b = 0
```

2) **| – Bitwise OR**

The bitwise OR operator compares each bit of two numbers and returns 1 if 1 bit is 1 and other bit is 0. If both the bits are same it returns 0.

Example:

```
a = 10
b = 4
print("a | b =", a | b)
```

Output:

```
a | b = 14
```


3) **^ – Bitwise XOR**

The bitwise XOR (exclusive OR) operator returns 1 if the bits are different in the two numbers; otherwise, it returns 0.

Example:

```
a = 10
b = 4
print("a ^ b =", a ^ b)
```

Output:

```
a ^ b = 14
```

4) **~ – Bitwise NOT**

The bitwise NOT operator inverts all the bits of the number. It changes every 1 to 0 and every 0 to 1. In Python, it returns the negative of the number plus one (i.e., `~a = -a - 1`).

Example:

```
a = 10
b = 4
print("~a =", ~a)
```

Output:

```
~a = -11
```

5) << – Left Shift

The left shift operator shifts the bits of a number to the left by the specified number of positions. It adds zeros from the right and multiplies the number by 2^n.

Example:

```
a = 10
b = 4
print("a << 1 =", a << 1)
```

Output:

```
a << 1 = 20
```

6) >> – Right Shift

The right shift operator shifts the bits of a number to the right by the specified number of positions. It divides the number by 2^n.

Example:

```
a = 10
b = 4
print("a >> 1 =", a >> 1)
```

Output:

```
a >> 1 = 5
```

3. Describe about different Logical operators in Python with appropriate examples.

➔

1) **Operator: and**

Name: Logical AND

Returns `True` if both the conditions are `True`, otherwise returns `False`.

Example:

```
a = 5
b = 10
if a > 0 and b > 0:
    print("Both numbers are positive")
else:
    print("One or both numbers are not positive")
```

Output:

```
Both numbers are positive
```

2) **Operator:** `or`

Name: Logical OR

Returns True if at least one condition is True, otherwise returns False.

Example:

```
a = 5
b = -2
if a > 0 or b > 0:
    print("At least one number is positive")
else:
    print("Both numbers are not positive")
```

Output:

```
At least one number is positive
```

3) **Operator:** `not`

Name: Logical NOT

Reverses the result of the condition. If the condition is `True`, it returns `False`; if it is `False`, it returns `True`.

Example:

```
a = 5
if not a < 0:
    print("Number is not negative")
else:
    print("Number is negative")
```

Output:

```
Number is not negative
```

4. Write a program to find the square root of a number.

→

```
num = float(input("Enter a number: "))
square_root = num ** 0.5
print("Square root of", num, "is", square_root)
```

Output:

```
Enter a number: 16
Square root of 16.0 is 4.0
```

5. Write a program to convert bits to Megabytes, Gigabytes and Terabytes.

→

```
bits = int(input("Enter number of bits: "))
bytes_value = bits / 8
megabytes = bytes_value / (1024 * 1024)
gigabytes = bytes_value / (1024 * 1024 * 1024)
terabytes = bytes_value / (1024 * 1024 * 1024 * 1024)
print("Megabytes:", megabytes)
print("Gigabytes:", gigabytes)
print("Terabytes:", terabytes)
```

Output:

```
Enter number of bits: 83886080
Megabytes: 10.0
Gigabytes: 0.009765625
Terabytes: 9.5367431640625e-06
```

6. Write a program to swap the value of two variables.

→

```
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
a, b = b, a
print("After swapping:")
```

```
print("First number:", a)
print("Second number:", b)
```

Output:

```
Enter first number: 12
Enter second number: 32
After swapping:
First number: 32
Second number: 12
```

7. Write a program to calculate surface volume and area of a cylinder.

➔

```
pi = 3.14
radius = float(input("Enter radius of the cylinder: "))
height = float(input("Enter height of the cylinder: "))
surface_area = 2 * pi * radius * (radius + height)
volume = pi * radius * radius * height
print("Surface Area of Cylinder:", surface_area)
print("Volume of Cylinder:", volume)
```


Output:

```
Enter radius of the cylinder: 5
Enter height of the cylinder: 10
Surface Area of Cylinder: 471.00000000000006
Volume of Cylinder: 785.0
```