Lesson-4: Operators & Expressions

Types Of Operators:

Based on number of operands
- Unary
- Binary
- Ternary

Based on type of operations

- Arithmetic operators
- Assignment operators
- Unary Minus operator
- Relational operators
- Logical operators
- Bitwise operators
- Membership operators
- Identity operators

Precedence: Associativity

**	(RL)
-	(LR)
*, /, //, %	(LR)
+, -	(LR)
<, >, <=, >=, !=, ==	(LR)
not	(LR)
and	(LR)
or	(LR)

Program: 1   Output:

```
a = 5
b = 3
c = 1

print( -a )
print( a + b )
print( a - b )
print( a * b )
print( a / b )
print( a % b )
print( a // b )
print( a ** b )
```

```
-5
8
2
15
1.6666666666666667
2
1
125
```

Keep in mind:

```
int  / int     = float
int  / float   = float
float / int    = float
float / float  = float
```

```
int  // int          = truncated int
int  // float        = truncated float
float // int         = truncated float
float // float       = truncated float
```

Difference between "Coercion" and "Type Conversion"

1) **Coercion**:   2 + 4.5 —> 2.0 + 4.5 —> 6.5 (Automatic Conversion)
2) **Conversion**: float(2) + 4.5 —> 2.0 + 4.5 —> 6.5 (Forced Conversion)
                   2 + int(4.5) —> 2 + 4 —> 6 (Forced Conversion)

Guess The Output:

>>> print(7/5)   >>> print(7/5.0) >>> print(7.0/5.0)

>>> print(7//5) >>> print(7//5.0) >>> print(7.0//5.0)

>>> print(0 % 5)

>>> print(3 * 2 + 1 ** 2 - 1 / 2 // 3 - 1 % 2 + 1)   **Output**: 7.0

Program: 2   **Output:**

```
a = b = 5
print(a, b)

a = 1; b = 2
print(a, b)

a, b = 1, 2
print(a, b)

n = 10
print(-n)
```

```
5 5
1 2
1 2
-10
```

Keep in mind:

x **and** y (if x is False, it returns x, otherwise it returns y)
x **or** y (if x is False, it returns y, otherwise it return x)
not x   (if x is False, it returns True, otherwise False)

NOTE: All non-zero values are considered 'True' and zero value is considered 'False'

Program: 3

```
print( 100 and 200 )
print( 0 and 200 )
print( 200 and 0 )
print( 100 or 200 )
print( 0 or 200 )
print( 200 or 0 )
print( not 10 )
print( not 0 )
print(True and 100 )
```

Ouput:

```
200
0
0
100
200
200
False
True
100
```

Program: 4

```
print(1 < 2)
print(1 < 2 < 3)
print(4 > 2 >=2 > 1)
print(1 < 2 > 3 < 4 )
print(1 > 4 == 3 < 4 != 3)
```

Output:

```
True
True
True
False
False
```

Program: 5

```
a = 9
b = 2
c = 0
print( a << b )
print( a >> b )
print( a & b )
print( a | b )
print( a ^ b )
print( ~c )
```

Output

```
36
2
0
11
11
-1
```

Program: 6

```
print(2 + 3 < 4 / 2 and 4 // 5 == 5 % 2 or 3 - 1)
```

Output:

```
2
```

Program: 7

```
a = 5
b = 3
c = 1

print( 2 + 3 * 5 )
print( (2 + 3) * 5 )
print( 2 * 3 * 5 )
print( 2 * (3 * 5) )
print( 2 ** 2 ** 3 )
print( 2 + 3 / 1 * 2 % 3 – 1 )
```

Output:

```
17
25
30
30
256
1.0
```

Guess The Output:

>>> int (10)   >>> int ('10')   >>> int ('10.8')

>>> float(10)   >>> float('10')   >>> float('10.8')

>>> eval (input ("Enter an expression:\n"))

>>> int(0b1010)   >>> float(0b1010)   >>> int ("0xA", 16)

>>> bin (10)   >>> hex (10)   >>> oct (10)

Program 8:

```
l = [ "Subhash", "Charan", "Amitabh"]
print ("Subhash" in l)
print ("Subhash" not in l)
print ("Akshay" not in l)
```

Output:

```
True
False
True
```

Program 9:

```
a = 10
b = 10

print( a is b )
print( a is not b )

a = 11

print( a is b )
print( a is not b )
```

Output:

```
True
False
False
True
```

Program 10:

```
print( 1 / 0 )
```

Output:

Run-Time Error
Zero Division Error

Program 11:

```
n = 10
print( n != 0 or 1 / 0 )
```

Output:

True

Guess The Output:

>>> 10 == 20   >>> 10 != 20   >>> 10 <= 20   >>> '2' < '9'

>>> '12' > '9'   >>> '1' > '000' >>> "99" > "11"   >>> 'heaven' > 'heau'

Programming Assignments:

1. WAP to display the powers of 2, one per line. (2 power 1, 2 power 2 and so on till 2 power 10).
2. WAP to enter a number and print 2 raised the number you entered.
3. WAP to enter a base and a power and find the base raised to that power.
4. WAP that allows user to enter 4 binary digits and convert it into decimal number.
5. WAP to print ascii values of "Hello World".
6. WAP to convert a degree from fahrenheit to celcius.
7. WAP to enter two float point values and displays the result of the first number divided by the second, with exactly six decimal places displayed in scientific notation.
8. WAP to enter an uppercase letter or a lowercase letter and display the corresponding UNICODE encoding.
9. WAP that prompts the user for a certain number of cities for the Travelling Salesman problem, and displays the total number of possible routes that can be taken.