

# CS & IT ENGINEERING

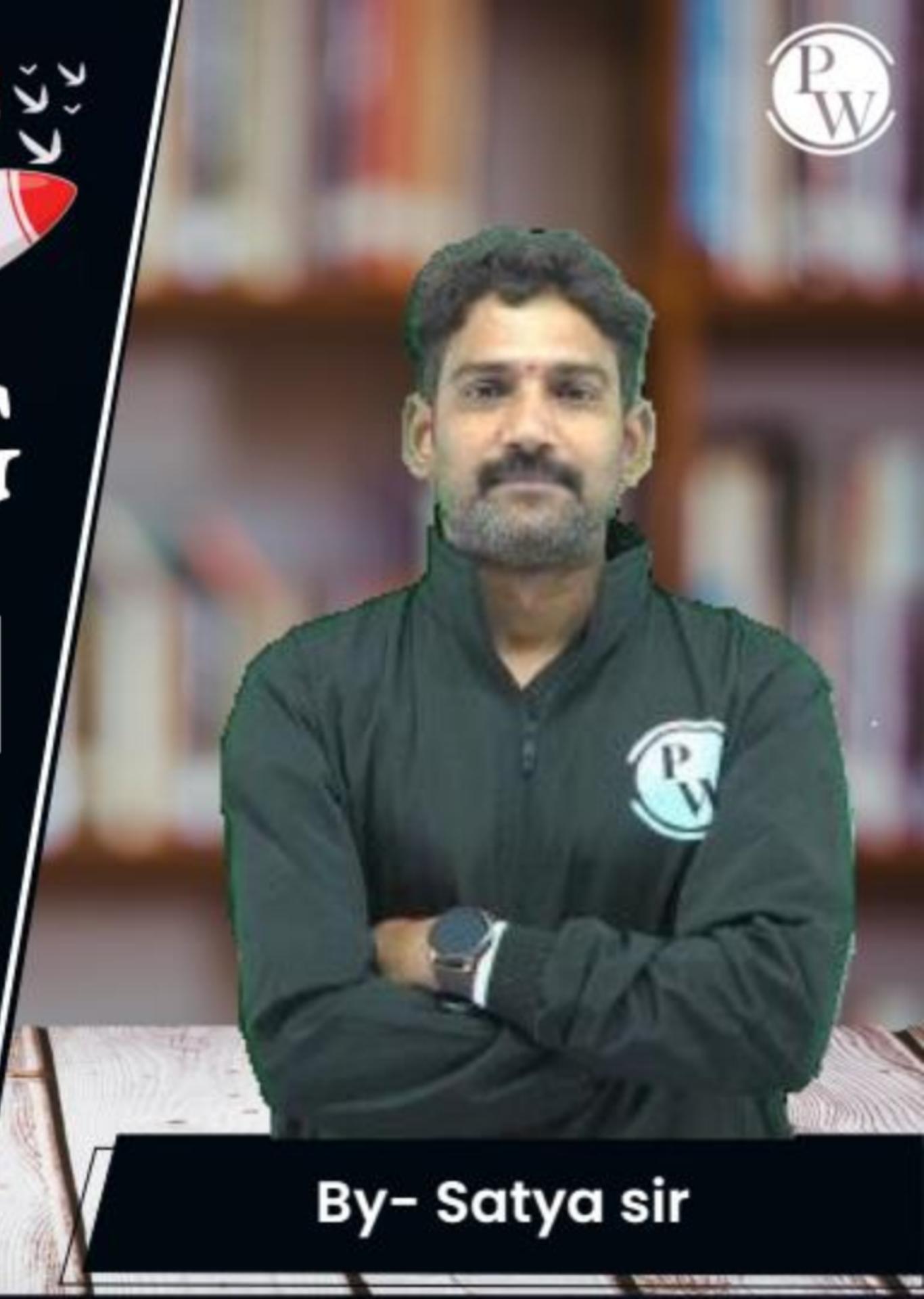


Python For Data Science

Basics Of Python

Lecture No.- 04

By- Satya sir



# Recap of Previous Lecture



## Operators

- 1) Arithmetic
- 2) Logical
- 3) Bitwise and shift
- 4) Comparison (or) Relational
- 5) Assignment
- 6) Identity
- 7) Membership

# Topics to be Covered



- Logical operators
- Bitwise & Shift
- Comparison & Assignment
- Arithmetic
- Operator Precedence & Associativity\*





## Topic : Operators in Python



Logical operators : Perform operation on Truth Values (True/False)

→ and, or, not

Ex:  $a=7$

$b=5$

$c=12$

A	B	A and B	A or B	not A
F	F	F	F	T
F	T	F	T	T
T	F	F	T	F
T	T	T	T	F

$$d = \frac{a > b}{\text{True}} \text{ and } \frac{b < c}{\text{True}} \quad d = \text{True}$$

$$e = \frac{c > a}{\text{True}} \text{ or } \frac{b > c}{\text{False}} \quad e = \text{True}$$

Print(d,e)

# True, True



## Topic : Operators in Python



Ex:  $a = 1$

$b = 0$

$c = -2$

$x = a + b \text{ or } b - 1$

$y = c + 1 \text{ and } a - 2$

$\text{Print}(x, y)$

A) True, True

B) True, False

C) -1, -1

D) 2, -1

[+ve] Non-Zero == True  
only zero == False  
None == False

When input to logical and, or operators is a number Then op also will be number [Decision making Number]

$x = 2 \text{ or } -1 \Rightarrow \text{True or } 1 \quad x = 2$

$y = -1 \text{ and } -1 \Rightarrow y = -1$



# Topic : Operators in Python



Ex:2

and > or [Left to right]  
Precedence Associativity

$x=1$

$y=0$

$c=4$

$d=-2$

$i = x \text{ or } y \text{ and } d \text{ or } c$

$j = c \text{ and } x \text{ or } y \text{ and } d$

$k = (y \text{ and } d) \text{ and } x \text{ or } c$

$\text{Print}(i+j+k)$

$$1+1+4=6$$

- and operation

⇒ If first i/p is True, Second i/p will be result.

⇒ If first i/p is False, First i/p(0) zero will be result.

- or operation

⇒ If first i/p is True, First i/p will be result

⇒ If first i/p is False, Second i/p will be result.



## Topic : Operators in Python



Ex: 3

i = 'GATE'

j = False

k = 'FALSE' (String)

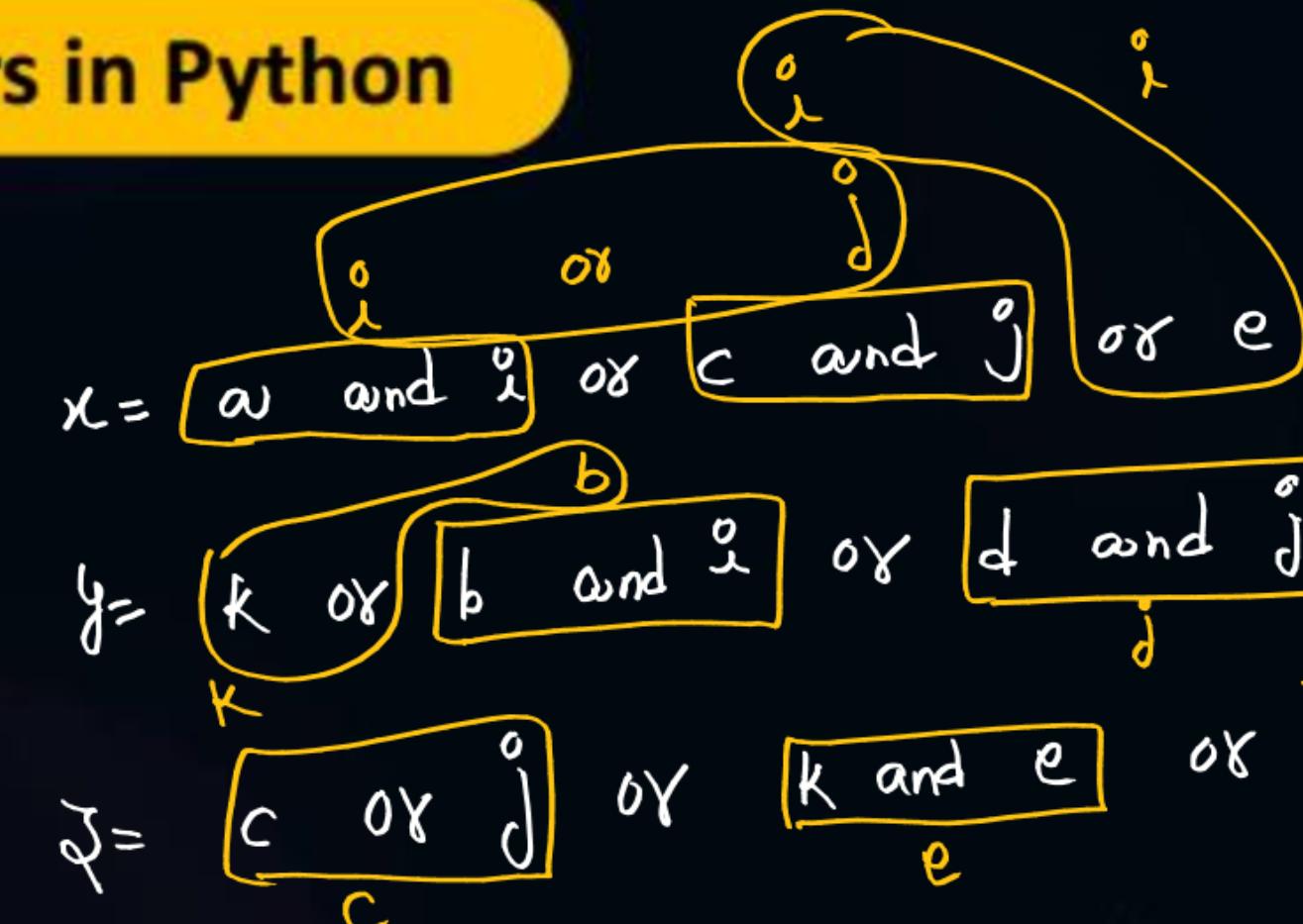
a = 1

b = 0

c = -1

d = True

e = None



Print(x, y, z)

≠ GATE FALSE -1

x = i  $\Rightarrow$  x = 'GATE'

y = k or j  $\Rightarrow$  k  $\Rightarrow$  y = 'FALSE'

z = c or b  $\Rightarrow$  c  $\Rightarrow$  z = c

$\Rightarrow$  z = -1



# Topic : Operators in Python



## Bitwise and Shift operators

: Perform operation on binary digits (bits: 1/0)

Operator	Meaning
&	Bitwise AND
	Bitwise OR
~	Bitwise NOT
^	Bitwise XOR
<<	Left shift
>>	Right shift

$$\boxed{X << n == X * 2^n}$$
$$\boxed{X >> n == X // 2^n}$$

A	B	$A \& B$	$A \mid B$	$A \wedge B$	$\sim A$
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0



## Topic : Operators in Python

( &gt; &lt; | )

Ex: 1     $a = 23$

$b = 14$

$c = 11$

$i = a \& b | c$      $i = 15$

$j = b \& c | a$      $j = 31$

$k = i \wedge j$      $k = 16$

Print(i, j, k)

# 15, 31, 16

$a = 23 \Rightarrow 10111$

$b = 14 \Rightarrow 01110$

$a \& b \Rightarrow \frac{01110}{00110} = 6$

$| c = \frac{01011}{01111} = 15$

$15 = 01111$

$31 = 11111$

$\overline{10000} = 16$

$b = 14 \Rightarrow 01110$

$c = 11 \Rightarrow 01011$

$b \& c = \frac{01011}{01010} = 10$

$a = \frac{10111}{11111} = 31$

&, A, |

Ex: 2

$$\begin{array}{r} 2 \\ \times 2 \\ \hline 2 \end{array} \quad \begin{array}{r} 2 \\ \times 2 \\ \hline 2 \end{array} \quad \begin{array}{r} 2 \\ \times 2 \\ \hline 2 \end{array}$$

$$Val_1 = 27 = 32 + 16 + 8 + 1 = 111001$$

$$Val_2 = 29 = 11101$$

$$Val_3 = 14 = 1110$$

$$Val_4 = 41 = 101001$$

$$i = Val_1 \wedge \underline{Val_2 \wedge Val_3} ; Val_4 \stackrel{i=61}{\circ}$$

$$j = \underline{Val_1 \wedge Val_2} \wedge Val_3 ; Val_4 \stackrel{j=63}{\circ}$$

$$k = Val_1 ; \underline{Val_2 \wedge Val_3} \wedge Val_4$$

$$k=61$$

$$\text{Point}(i, j, k) \# 61, 63, 61$$

K  $\Rightarrow 29 \& 14 = 11101$

$$\begin{array}{r} 01110 \\ \hline 01100 \end{array}$$

$$\begin{array}{r} 001100 \\ -101001 \\ \hline 100101 \end{array}$$

$$\begin{array}{r} 100101 \\ -111001 \\ \hline 111101 = 61 \end{array}$$

$$i = 29 \& 14 = 11101$$

$$\begin{array}{r} 01110 \\ \hline 001100 \end{array}$$

$$57 = \frac{111001}{110101}$$

$$41 = \frac{101001}{111101} = 61$$

$$\begin{array}{r} 29 \\ 14 - 1 \\ 7 - 0 \\ 3 - 1 \\ 1 - 1 \end{array}$$

$$j \Rightarrow 57 = 111001$$

$$\begin{array}{r} 011101 \\ \hline 011001 \end{array}$$

$$14 = \frac{001110}{010111}$$

$$41 \Rightarrow \frac{101001}{111111}$$

## Shift operators ( $<<$ , $>>$ )

$\text{Val} = \text{Number} << \text{Positions} \Rightarrow \text{Val} = \text{Number} * 2^{\text{Positions}}$

$\text{Val} = \text{Number} >> \text{Positions} \Rightarrow \text{Val} = \text{Number} / 2^{\text{Positions}}$

Ex:

$$i = 15$$

$$j = i << 1 \quad j = 15 \times 2^1 = 30$$

$$k = i << 2 = 15 \times 2^2 = 60$$

$$l = i << 3 = 15 \times 2^3 = 120$$

$$15 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \xrightarrow{<1 \Rightarrow} = 30$$

$$i = 15$$

$$j = i >> 1 \quad j = 15 / 2^1 = 7$$

$$k = i >> 2$$

$$l = i >> 3$$

$$\begin{aligned} j &= 15 / 2^1 & k &= 15 / 2^2 \\ &= 7 & &= 15 / 4 \\ & & &= 3 \end{aligned}$$

$$\begin{array}{c} 00001111 \\ \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \\ 00000111 \\ \hline 7 \end{array}$$

$$\begin{aligned} j &= 15 / 2^3 & k &= 15 / 2^2 \\ &= 15 / 8 & &= 1 \end{aligned}$$



# Topic : Operators in Python



Please Practice

<b>Operators</b>	<b>Associativity</b>
() Highest precedence	Left - Right
**	Right - Left
+x , -x, ~x	Left - Right
*, /, //, %	Left - Right
+, -	Left - Right
<<, >>	Left - Right
&	Left - Right
^	Left - Right
	Left - Right
Is, is not, in, not in, <, <=, >, >=, ==, !=	Left - Right
Not x	Left - Right
And	Left - Right
Or	Left - Right
If else	Left - Right
Lambda	Left - Right
=, +=, -=, *=, /= Lowest Precedence	Right - Left



## Topic : Operators in Python



H/W Questions

①

a = 'True'

b = 'False'

c = False

d = True

i = 10

j = 0

x = a and j or i

y = c or d and b

x = \_\_\_\_

y = \_\_\_\_

②

a = 13

b = 27

c = 34

d = 71

x = a & b & d

y = b | c & a

z = c | d & a

x = \_\_\_\_

y = \_\_\_\_

z = \_\_\_\_



## 2 mins Summary



- Logical operators

and, or, not

- Bitwise operators : &, |,  $\wedge$ , ~

Shift operators : <<, >>

To be Contd...



THANK - YOU