**Python Assignment:**

1.Python basic syntax:

Python uses indentation to indicate a block of code.

Example

if 5 > 2:

print("Five is greater than two!")

2.Python Operators:

Operators are used to perform operations on variables and values.

Python divides the operators in the following groups:

Arithmetic operators

Assignment operators

Comparison operators

Logical operators

Identity operators

Membership operators

Bitwise operators

Python Arithmetic Operators

Arithmetic operators are used with numeric values to perform common mathematical operations:

Operator Name Example Try it

+ Addition x + y

- Subtraction x - y

\* Multiplication x \* y

/ Division x / y

% Modulus x % y

\* Exponentiation x \* y

// Floor division x // y

Python Assignment Operators

Assignment operators are used to assign values to variables:

Operator Example Same As Try it

= x = 5 x = 5

+= x += 3 x = x + 3

-= x -= 3 x = x - 3

\*= x \*= 3 x = x \* 3

/= x /= 3 x = x / 3

%= x %= 3 x = x % 3

//= x //= 3 x = x // 3

\*= x \*= 3 x = x \*\* 3

&= x &= 3 x = x & 3

|= x |= 3 x = x | 3

^= x ^= 3 x = x ^ 3

>>= x >>= 3 x = x >> 3

<<= x <<= 3 x = x << 3

Python Comparison Operators

Comparison operators are used to compare two values:

Operator Name Example Try it

== Equal x == y

!= Not equal x != y

> Greater than x > y

< Less than x < y

>= Greater than or equal to x >= y

<= Less than or equal to x <= y

Python Logical Operators

Logical operators are used to combine conditional statements:

Operator Description Example Try it

and Returns True if both statements are true x < 5 and x < 10

or Returns True if one of the statements is true x < 5 or x < 4

not Reverse the result, returns False if the result is true not(x < 5 and x < 10)

Python Identity Operators

Identity operators are used to compare the objects, not if they are equal, but if they are actually the same object, with the same memory location:

Operator Description Example Try it

is Returns True if both variables are the same object x is y

is not Returns True if both variables are not the same object x is not y

Python Membership Operators

Membership operators are used to test if a sequence is presented in an object:

Operator Description Example Try it

in Returns True if a sequence with the specified value is present in the object x in y

not in Returns True if a sequence with the specified value is not present in the object x not in

Python Bitwise Operators

Bitwise operators are used to compare (binary) numbers:

Operator Name Description

& AND Sets each bit to 1 if both bits are 1

| OR Sets each bit to 1 if one of two bits is 1

^ XOR Sets each bit to 1 if only one of two bits is 1

~ NOT Inverts all the bits

<< Zero fill left shift Shift left by pushing zeros in from the right and let the leftmost bits fall off

>> Signed right shift Shift right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off

3.