Arithmetic Operators

Operator	Name	Example
+	Addition	x + y
-	Subtraction	x – y
*	Multiplication	x * y
/	Division	x / y
%	Modulus	x % y
**	Exponentiation	x ** y
//	Floor division	x // y

Arithmetic Operators(Continued)

- **Addition :** Adds values on either side of the operator.
  example:

```
>>> x = 5
>>> y = 2
>>> print(x + y)
Output = 7
```

- **Subtraction :** Subtracts right hand operand from left hand operand.
  example:

```
>>> x = 5
>>> y = 2
>>> print(x - y)
Output = 3
```

- **Multiplication :** Multiplies values on either side of the operator.
  example:

```
>>> x = 3
>>> y = 2
>>> print(x * y)
Output = 6
```

Arithmetic Operators(Continued)

- **Division :** Divides left hand operand by right hand operand.
  example:

```
>>> x = 10
>>> y = 4
>>> print(x / y)
Output = 2.5
```

- **Modulus :** This operator is also known as remainder operator

example 1:

```
>>> x = 10
>>> y = 5
>>> print(x % y)
Output = 0
```

example 2:

```
>>> x = 3
>>> y = 4
>>> print(x / y)
Output = 3
```

Arithmetic Operators(Continued)

- **Exponentiation :** Performs exponential (power) calculation on operators. Power operator.
  example:

```
>>> x = 3
>>> print(x**3)
Output = 27
```

- **Floor division :** The real floor division operator is // . It returns floor value for both integer and floating point arguments.
  example :

```
>>> 5 // 2 result 2
>>> -5 // 2 result -3
>>> 2.0 // 2 result 1.0
>>> -5.0 // 2 result -3.0
```

AI উত্তর

In math, the floor value is the largest integer that is less than or equal to a number, and the ceiling value is the smallest integer that is greater than or equal to a number.

Python Assignment Operators

Operator	Example	Same As
=	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

\

Identity Operators

Operator	Description	Example
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

- **(is) :**

Example 1 :

```
>>> x = 10
>>> y = 10

>>> print(x is y)
output : True
```

Example 2:

```
>>> x = 10
>>> y = 20

>>> print(x is y)
output : False
```

- **(is not) :**

Example 1:

```
>>> x = 10
>>> y = "Hello world"

>>> print( x is not y)
Output : True
```

Example 2:

```
>>> x = 10
>>> y = "10"

>>> print( x is y)
Output : False
```

Python Membership Operators

Operator	Description	Example
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 y

- **in :**

Example 1:

```
>>> x = ["apple", "banana"]
>>> print("Mango" in x)

Output : False
```

Example 2 :

```
>>> x = "Hello world"
>>> print('H' in x)

Output : True
```

- **not in :**

Example 1:

```
>>> x = "Hello world"
>>> print('H' not in x)

Output : False
```

Example 2 :

```
>>> x = ["apple", "banana"]
>>> print("Mango" not in x)

Output : True
```

Logical Operators

- And
- Or
- not

Comparison or Relational Operators

- == (equal)
- != (not equal)
- > (greater than)
- < (less than)
- >= (greater than or equal)
- <= (less than or equal)