operators

Different categories of binary operators in python are :

- ⦿ Arithmetic
- ⦿ Relational
- ⦿ Logical
- ⦿ Augmented assignment
- ⦿ Membership **(in)**
- ⦿ Identity **(is)**
- ⦿ Bitwise ***(not in curriculum)***

Arithmetic operator

Operator	**Integer eg**	**Float eg**
+ addition	4+7 is 11 -3+8 is 5	
- subtraction	20-5 is 15	Behavior is same as integer
* multiplication	4*7 is 28 -3*8 is -24	except for, result
/ division	20/5 is 4.0 20/ 3 is 6.666666666666667 -21/4 is -5.25	which is floating point
// integer division OR Floor division	20//5 is 4 20//3 is 6	32.0//3.0 is 10.0

	-21//4 is -6	
% remainder OR modulo	4%2 is 0 -5%2 is 1	“
** power OR exponentiation	2**2 is 4 -2**2 is -4 2**-2 is 0.25 -2**-2 is -0.25	Same as integer, except for result

Relational operator

- Operators to compare two values, and return a boolean value (True, False)
- Behave same for integer, floating point and string data

Operator	Purpose
<	Checks if value on, LHS is smaller than RHS
>	Checks if value on, LHS is larger than RHS
<=	Checks if value on, LHS is smaller than or equal to RHS
>=	Checks if value on, LHS is larger than or equal ton RHS
==	Checks if value on, LHS is equal to RHS
!=	Checks if value on, LHS is not equal to RHS

- For checking value of a variable in range, say x in range 8 to 15 both inclusive, we may write

8<=x<=15

Note : Floating point should not be compared for equality as they are stored with a precision of 17 significant digits only.

Logical operators

- Used for combining two conditions
- Return a boolean value

Operator	Purpose
and	If condition on both side of operator is true, then return true, false otherwise
or	If condition on any side of operator is true, then return true, false otherwise
not	It inverts the boolean value of expression / variable

Assignment operator

- = is the operator used for assigning a value.
- `>>>int var  = 35+x/y`
    - In this statement, processing happens from right to left
    - First the expression on RHS (35+x/y) will get evaluated and then only LHS will looked into.
    - Same idea is used for lvalue and rvalue.
    - lvalue is an address, which will exist after execution of statement, as it is a variable which can be modified
    - rvalue might be a temporary value, appearing on RHS of assignment operator.

Agumented assignment operator

- Also known as compound assignment operator

Operator	Behaviour
+=	Add and assign
-=	Subtract and assign
*=	Multiply and assign
/=	Divide and assign
//=	Floor divide and assign
%=	Modulus and assign
**=	Exponent and assign

Operator precedence

- Except for exponent operator, all are left to right associative
- Highest to lowest

Operator
**
*,/,//,%

+,-
Relational, membership and identity operators
Not
And
Or

Expressions

- A valid combination of value, variable(s), operator(s) and function call is known as expression
- An expression needs to be evaluated thus it is usually assigned to a variable or appears in a print statement.
- Examples of expressions are:

Expression	operands	Type	Will evaluate to
x+y*2+34	variables, constants	arithmetic	a numeric value
a>b	variables	relational	a Boolean value
"Great"!="great"	string constants	relational	a Boolean value
a+b > c+d	variables	mixed	a Boolean value
x!=y and z>4	variables, constants	mixed	a Boolean value
4*(45+60)	constants	arithmetic	a numeric value