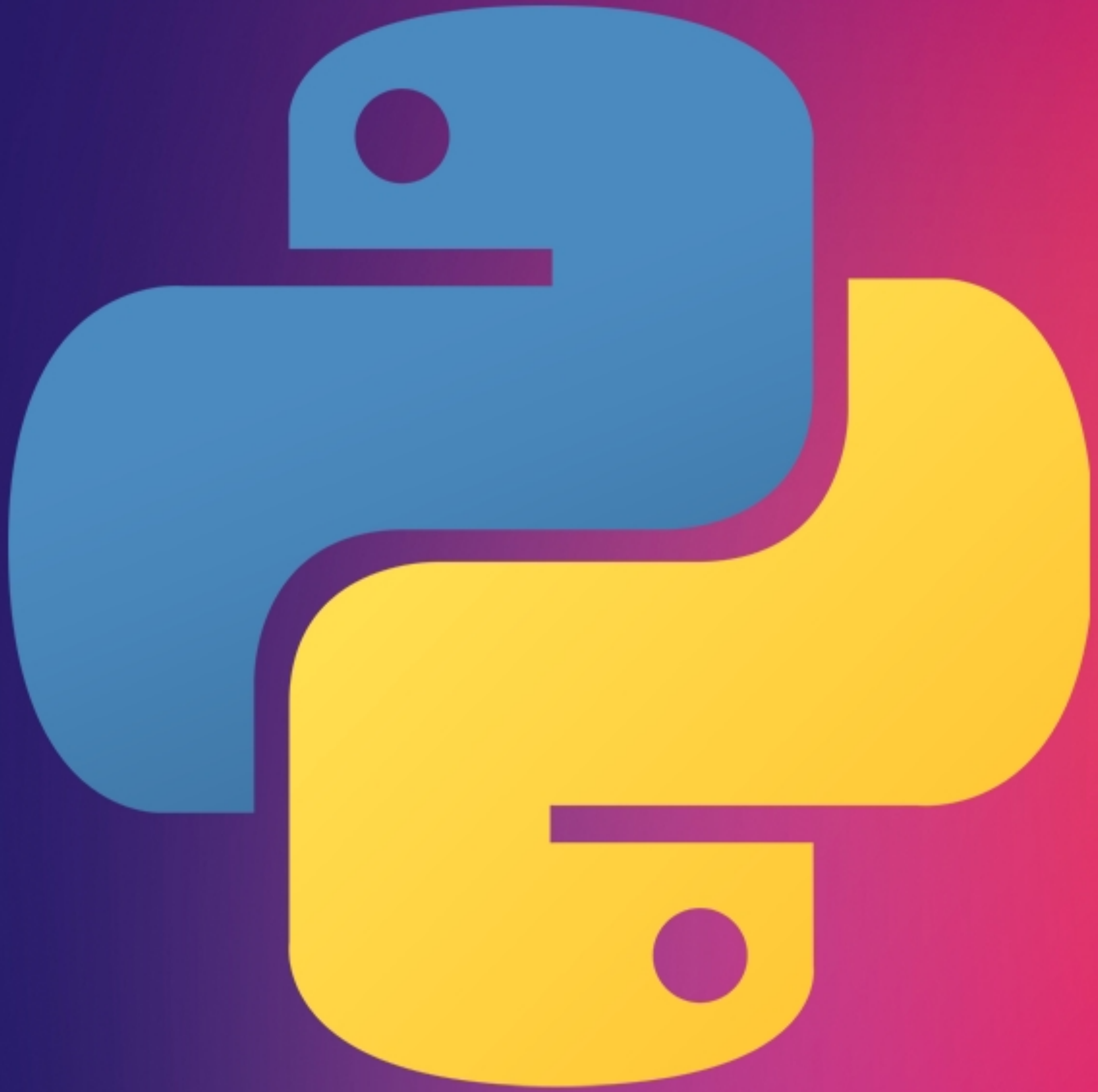
PYTHON

For Beginners

By Aman Guliani



RECAP

- Last Time we learned
  - Numbers & Math and Radom Lib.
  - String, string access
  - Boolean

LISTS

LISTS

- list is a collection of data types. So you can define a list of strings or numbers or mix them both.
- Elements can be changes within a list.
- Defined as list_var = [“String1”, “String2”, 2, 3]
- Lists are accessed same as numbers list_var[0] etc
- Lists have methods too, just like strings , imp ones
  - list.append(element)
  - list.insert(index, obj)
  - list.reverse()
  - list.sort() !!!!

SETS

SETS

- Set is similar to list, but it doesn’t allow duplicates
- Defined as set_var = {“String1”, “String2”, 2, 3}
- Sets cannot be accessed by index since they are unordered. So only using for.
- Once a set is created, you cannot change an element
- You can add value to it using add() or remove an element using remove()
- Used for generally removing duplicates.

DICTIONARY

DICTS

- One of the core data structures which is basis of all databases.
- Key -> Value.
- Define car_model = {“id”: “1”, “name”: “Mazda”, “model”: “Cx5”, “color”: “Red”}
- Accessed as car_model[“name”] or car_model.get(“name”)
- You can change value for a key car_model[“color”] = “blue”
- You can also add more keys car_model[“year”] = 2019
- Check if key exist in dict :
    - If “color” in car_model:
        - Change color

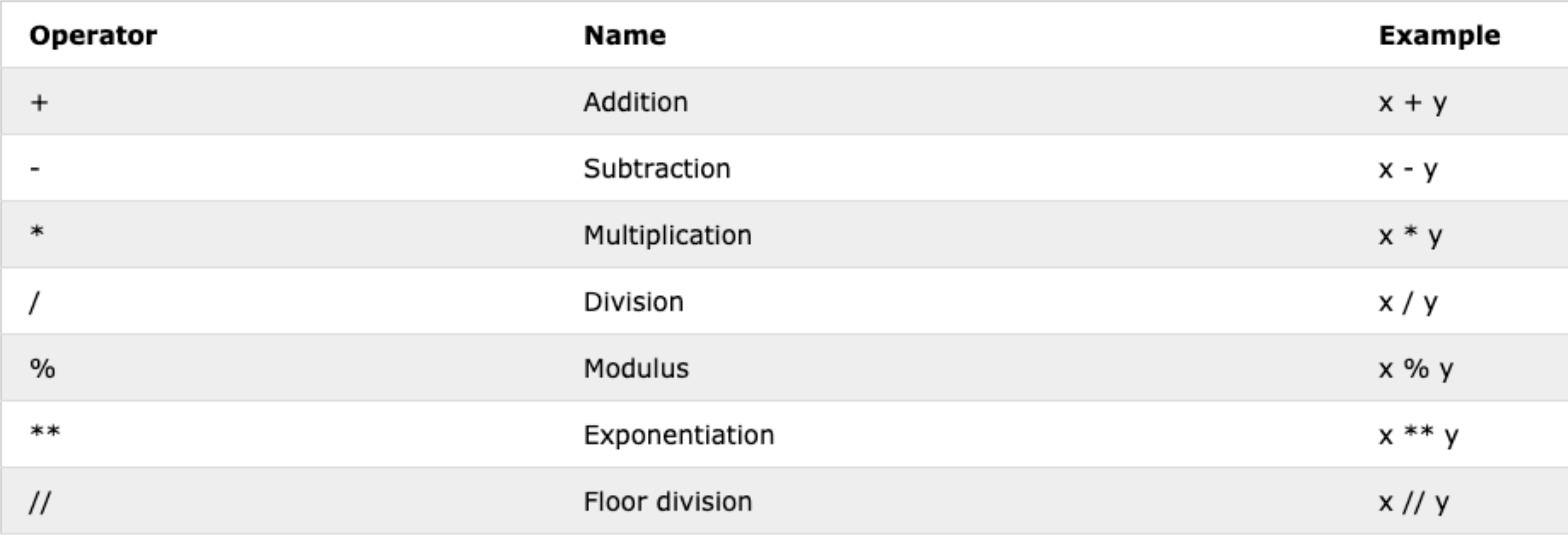
OPERATORS

OPERATORS

- Operators are used to perform operations on variables and values.

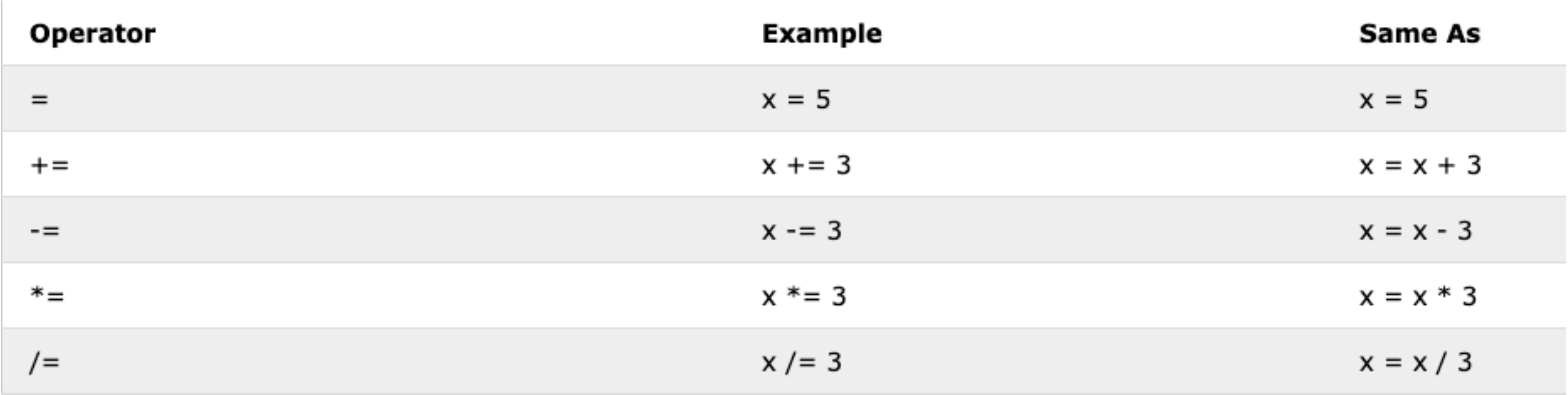
- Operators are divided into different groups
  - Arithmetic operators
  - Assignment operators
  - Comparison operators
  - Logical operators
  - Identity operators
  - Membership operators

ARITHMETIC OPERATORS – WE ALREADY KNOW THESE !!



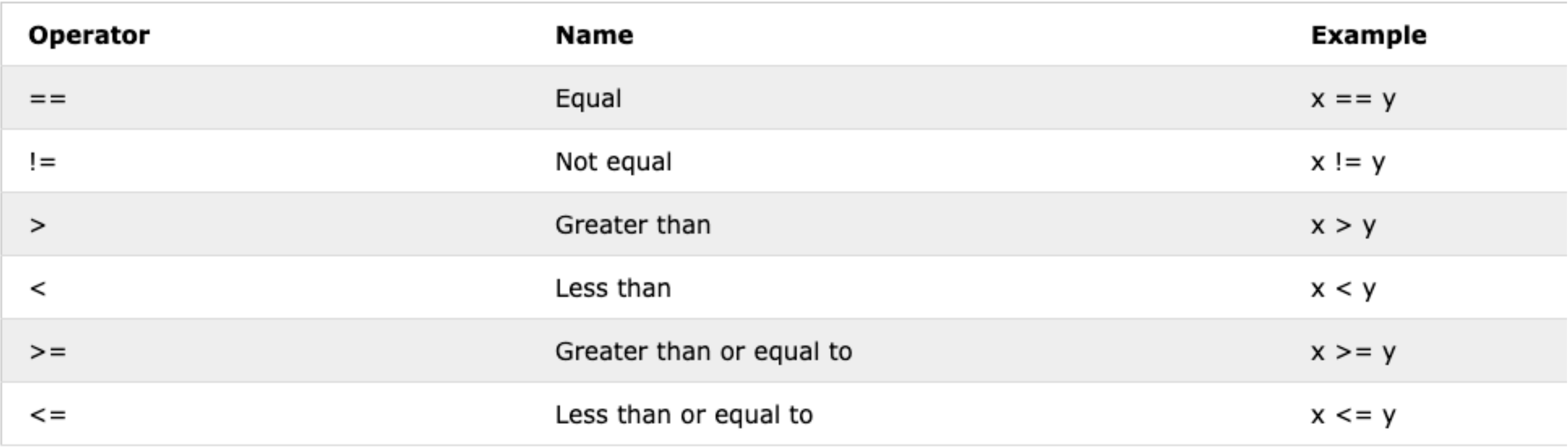
Operator	Name	Example
+	Addition	x + y
-	Subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
%	Modulus	x % y
**	Exponentiation	x ** y
//	Floor division	x // y

ASSIGNMENT OPERATIONS - SOUNDS FAMILIAR



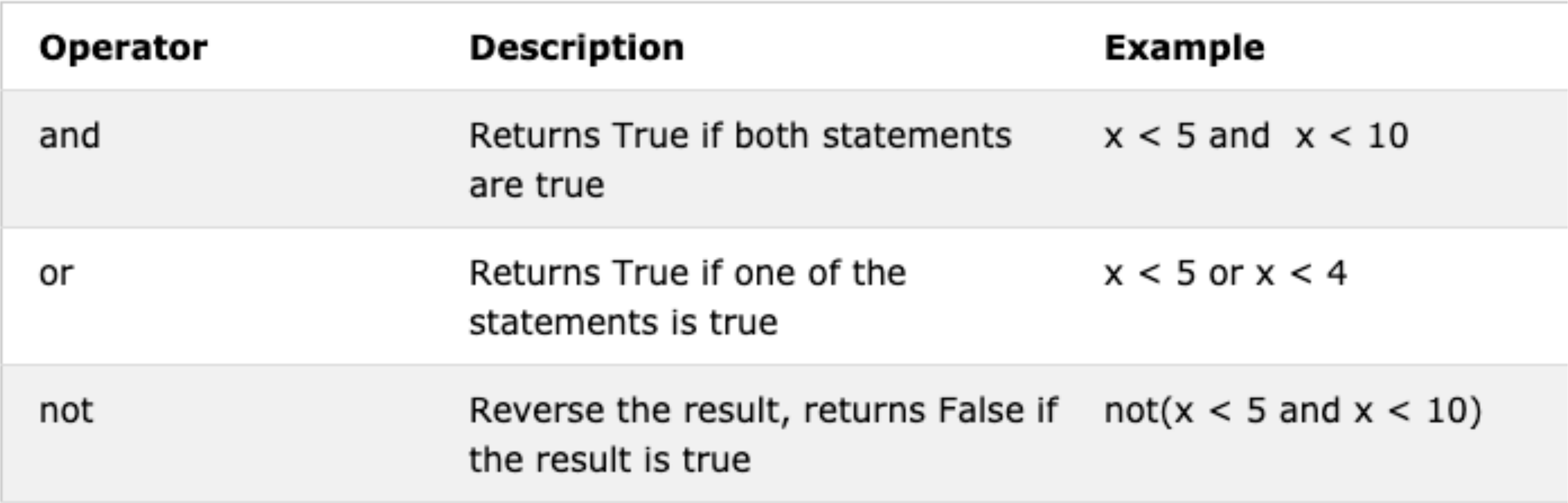
Operator	Example	Same As
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3

COMPARISON OPERATORS – DIDN'T WE LEARN ABOUT THESE TOO ?



Operator	Name	Example
==	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

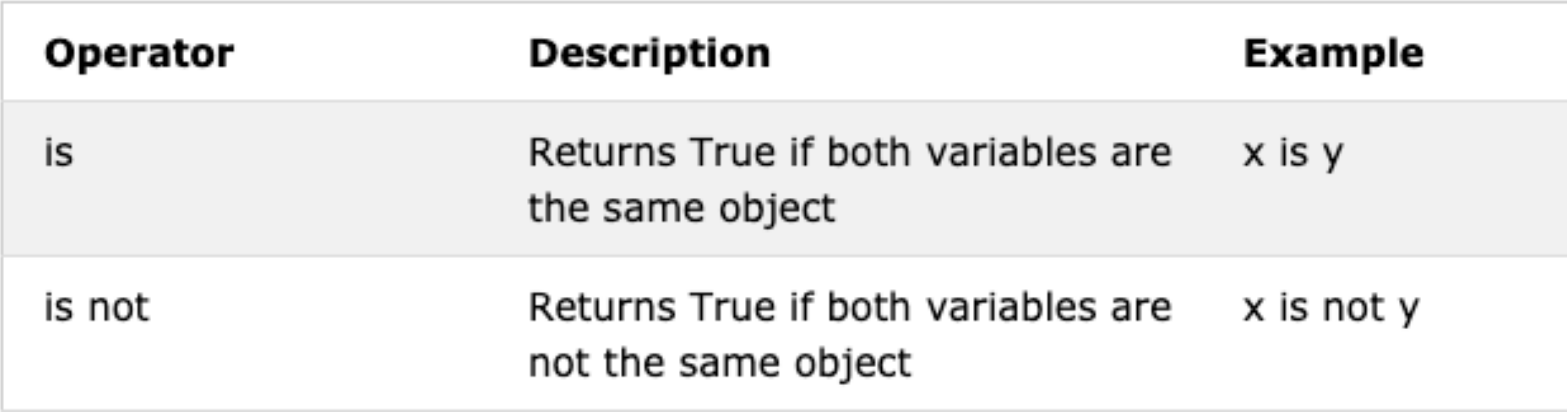
LOGICAL OPERATORS



Operator	Description	Example
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)

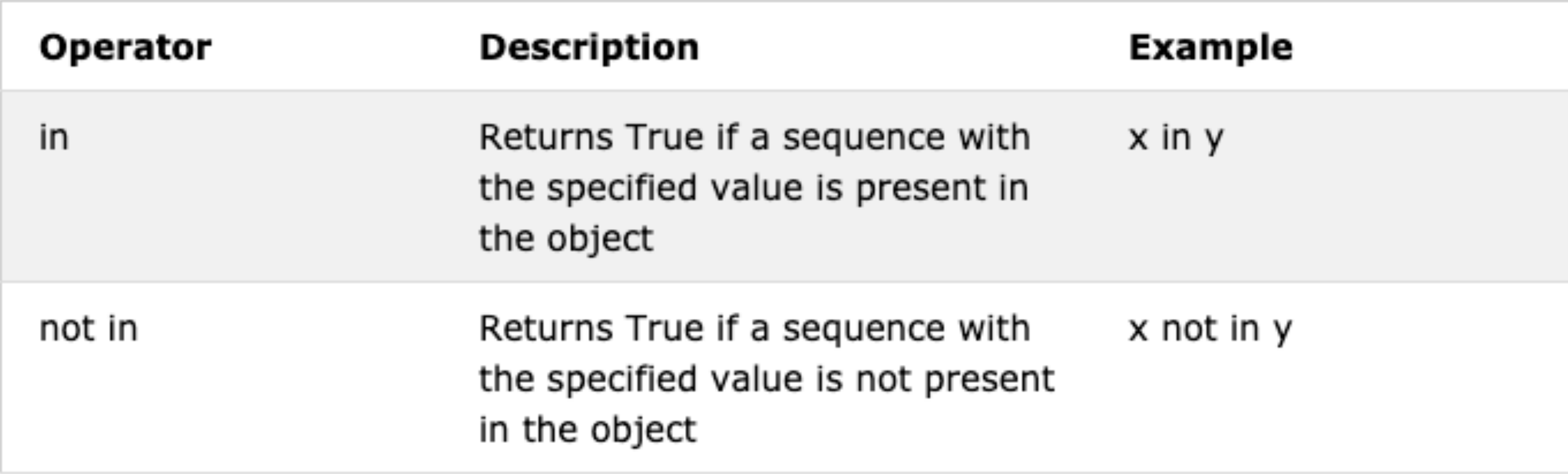
python™

IDENTITY OPERATORS



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

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

LOOPS

LOOPS

- For loop - used for iterating over a sequence
  - Syntax :
    - For var in range:
      - Do something
- While loop - execute a set of statements as long as a condition is true.
  - Syntax:
    - While condition:
      - Do something

python™