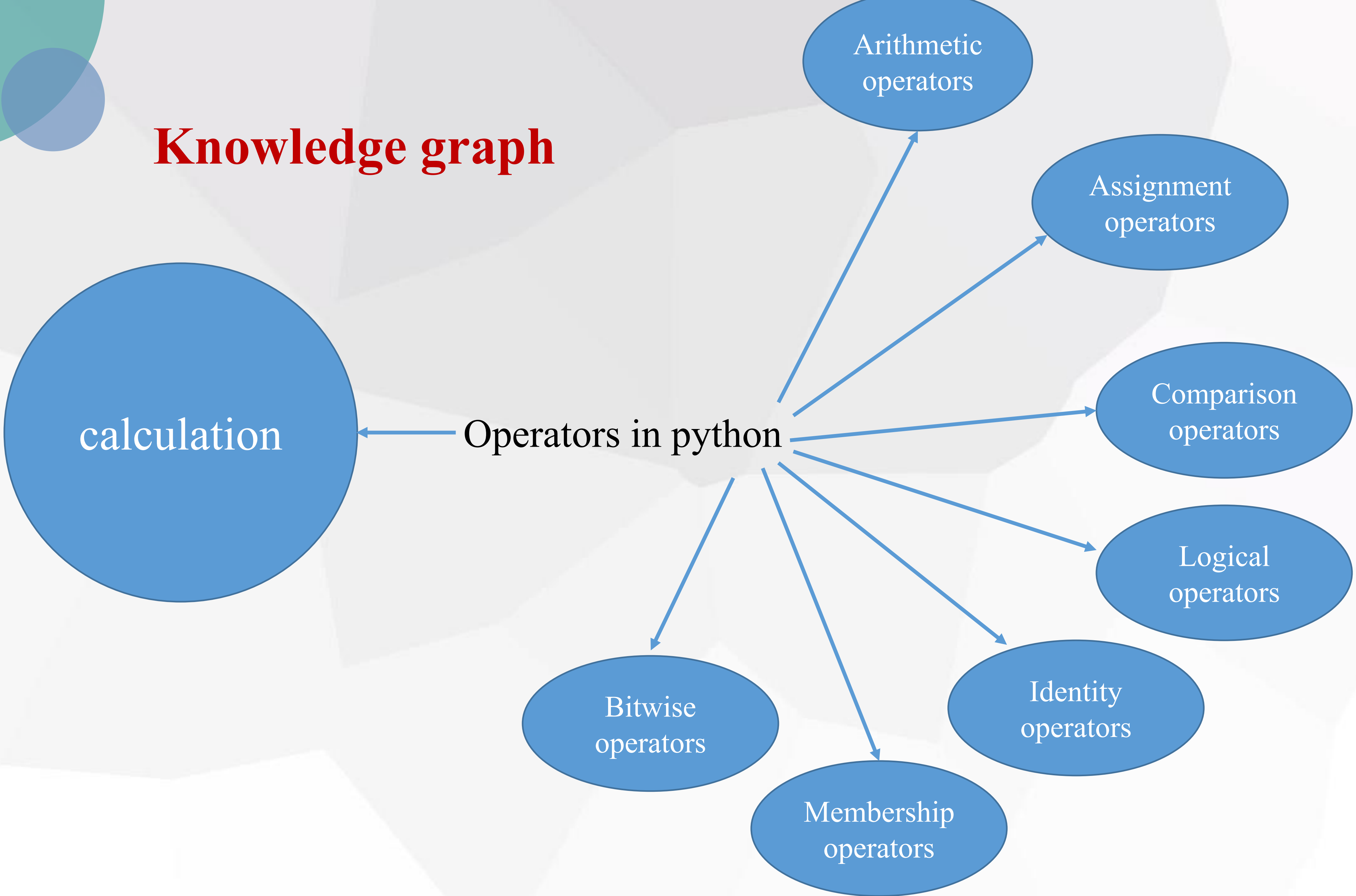
Python Operators

Knowledge graph



calculation

Operators in python

Arithmetic operators

Assignment operators

Comparison operators

Logical operators

Identity operators

Membership operators

Bitwise operators

Python Operators

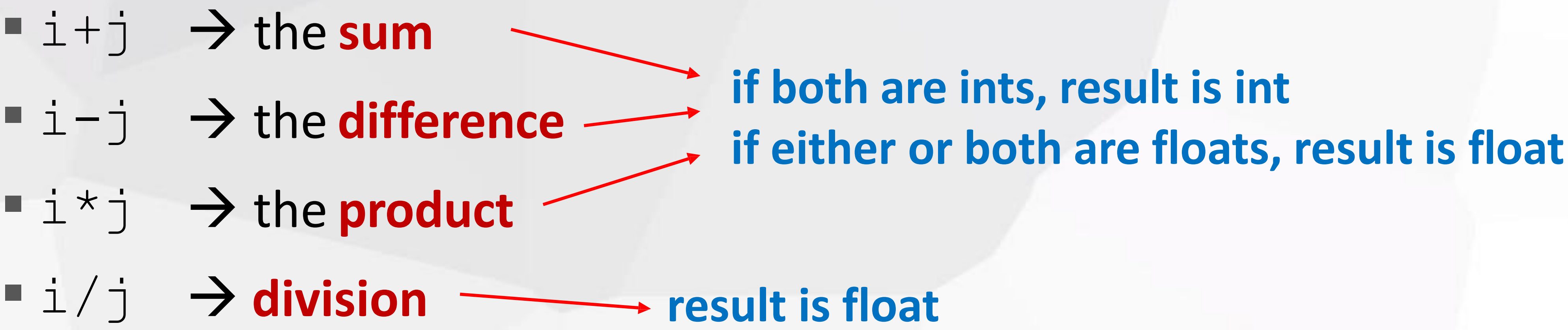
- Python divides the operators in the following groups:
  - **Arithmetic operators**
  - **Assignment operators**
  - **Comparison operators**
  - **Logical operators**
  - **Identity operators**
  - **Membership operators**
  - **Bitwise operators**

Python Operators

- Python divides the operators in the following groups:
  - **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 on ints and floats



- `i+j` → the **sum**
- `i-j` → the **difference**
- `i*j` → the **product**
- `i/j` → **division**

if both are ints, result is int
if either or both are floats, result is float

result is float

Python Operators

- Python divides the operators in the following groups:
  - **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

Python Operators

- Python divides the operators in the following groups:
  - **Comparison operators**

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

Python Operators

- Python divides the operators in the following groups:
  - ➢**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 Operators

- Python divides the operators in the following groups:
  - **Logical operators**

A	B	A and B	A or B
True	True	True	True
True	False	False	True
False	True	False	True
False	False	False	False

Python Operators

- Python divides the operators in the following groups:
    - **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

Python Operators

- Python divides the operators in the following groups:
  - ➢**Membership operators**
  - ➢

Operator	Description	Example
in	Returns True if a variable/value found in the sequence	10 in list1
not in	Returns True if a variable/value not found in the sequence	10 not in list1

单选题 1分

设置

此题未设置答案，请点击右侧设置按钮

Bitwise __________ outputs 1 if either of the bits is 1 and outputs 0 if both of the bits are 1.

A OR

B AND

C XOR

D NOT

提交

Python Operators

- Python divides the operators in the following groups:
  - **Bitwise operators**

Operator	Name	Description
&	AND	Sets to 1 if both bits are 1
|	OR	Sets to 1 if one of two bits is 1
^	XOR	Sets to 1 if only one of two bits is 1
~	NOT	Inverts all the bits
<<	left shift	Shift left from the right, shift 1 bit means multiply by 2
>>	right shift	Shift right from left and keep the sign, shift 1 bit means divided by 2

Python Operators

- Arithmetic Shift:

Both an arithmetic left shift and a logical left shift (<<) correspond to **a multiplication by 2** when there is no overflow.

x << y

Returns multiplying x by 2**y

```
print(10 << 2)
 40
```

Python Operators

- Arithmetic Shift:

With numbers in twos complement notation, right arithmetic and logic shift (<<) corresponds to **a division by 2**, with **floor truncation for odd numbers**

x >> y
Returns x by 2**y.

```
print(3 >> 2, 3>>1, -3>>1)
 0 1 -2
```