Arithmetic operators

In [1]:
```
a=10
b=20
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(a//b)
print(a%b)
print(a**b)
```

```
30
-10
200
0.5
0
10
100000000000000000000
```

Comparison(Relational Operator)

In [2]:
```
c=5
d=15
print(c==d)
print(c!=d)
print(c>d)
print(c<d)
print(c>=d)
print(c<=d)
```

```
False
True
False
True
False
True
```

Logical Operator

In [3]:
```
A=10
B=20
C=30
print(A<B and B>A)
print(A>B and C<A)
print(C>B and B>A)
print(B<A and A>C)
print(not(A>B))
print(not(B<C))
```

```
True
False
True
False
True
False
```

Assignment Operator

In [4]:
```
X=8
X+=8
print(X)
X-=4
print(X)
X*=2
print(X)
X/=2
print(X)
X//=2
print(X)
X**=2
print(X)
X%=2
print(X)
```

```
16
12
24
12.0
6.0
36.0
0.0
```

Binary,Octal,Hexadecimal,Decimal

In [5]:
```
n=10
print(n)
```

```
10
```

In [7]:
```
n='ob10011'
print(n)
```

```
ob10011
```

In [8]:
```
bin(19)
```

Out[8]:
```
'0b10011'
```

In [10]:
```
n=0o12
print(n)
```

```
10
```

In [12]:
```
oct(10)
```

Out[12]:
```
'0o12'
```

In [13]:
```
n=0XA23
print(n)
```

```
2595
```

In [14]:
```
hex(2595)
```

Out[14]:
```
'0xa23'
```

In [15]:
```
import sys
sys.version
```

Out[15]:
```
'3.13.5 | packaged by Anaconda, Inc. | (main, Jun 12 2025, 16:37:03) [MSC v.192
9 64 bit (AMD64)]'
```

In [16]:
```
import keyword
keyword.kwlist
```

Out[16]:
```
['False',
 'None',
 'True',
 'and',
 'as',
 'assert',
 'async',
 'await',
 'break',
 'class',
 'continue',
 'def',
 'del',
 'elif',
 'else',
 'except',
 'finally',
 'for',
 'from',
 'global',
 'if',
 'import',
 'in',
 'is',
 'lambda',
 'nonlocal',
 'not',
 'or',
 'pass',
 'raise',
 'return',
 'try',
 'while',
 'with',
 'yield']
```

In [17]:
```
len(keyword.kwlist)
```

Out[17]:
```
35
```

In [ ]: