Operators

Arthemetic operator

In [1]:
```
x=5
y=10
```

In [2]:
```
x,y
```

Out[2]: (5, 10)

In [3]:
```
x
y
```

Out[3]: 10

In [4]:
```
x+y
```

Out[4]: 15

In [5]:
```
x-y
```

Out[5]: -5

In [6]:
```
x*y
```

Out[6]: 50

In [7]:
```
x/y
```

Out[7]: 0.5

In [8]:
```
x//y
```

Out[8]: 0

In [9]:
```
x%y
```

Out[9]: 5

In [10]:
```
x**y
```

Out[10]: 9765625

In [12]:
```
100**2
```

Out[12]: 10000

In [ ]:

Assignment Operators

In [14]:
```
x+=2
```

In [15]:
```
x
```

Out[15]: 7

In [ ]:
```

```

In [18]:
```
x=10
x
```

Out[18]: 10

In [19]:
```
x+2
```

Out[19]: 12

In [21]:
```
x+=2
```

In [22]:
```
x
```

Out[22]: 14

In [23]:
```
x-=2
x
```

Out[23]: 12

In [24]:
```
x*=3
```

In [25]:
```
x
```

Out[25]: 36

In [26]:
```
x*2
```

Out[26]: 72

In [27]:
```
x/=2
```

In [28]:
```
x
```

Out[28]: 18.0

In [29]:
```
x//=2
```

In [30]:
```
x
```

Out[30]: 9.0

In [31]:
```
y=20
y//=2
```

In [32]:
```
y
```

Out[32]: 10

In [ ]:
```

```

In [33]:
```
y=40.5
y//=2
y
```

Out[33]: 20.0

In [ ]:
```

```

unary operator

In [35]:
```
n=7
```

In [36]:
```
m=-(n)
```

In [37]:
```
m
```

Out[37]: -7

In [ ]:
```

```

In [38]:
```
5
```

Out[38]: 5

In [39]:
```
-5
```

Out[39]: -5

In [ ]:
```

```

In [34]:
```
~7
```

Out[34]: -8

In [ ]:
```

```

Relational operators

In [40]:
```
a=5
b=6
```

In [41]:
```
a<b
```

Out[41]: True

In [42]:
```
a>b
```

Out[42]: False

In [44]:
```
a==b
```

Out[44]: True

In [45]:
```
a<=b
```

Out[45]: True

In [46]:
```
a>=b
```

Out[46]: True

In [47]:
```
a=5
```

In [48]:
```
b=10
```

In [49]:
```
a<=b
```

Out[49]: True

In [50]:
```
a>=b
```

Out[50]: False

In [51]:
```
a!=b
```

Out[51]: True

In [ ]:
```

```

Logial operator

In [52]:
```
a=5
b=4
```

In [53]:
```
a==b
```

Out[53]: False

In [54]:
```
a!=b
```

Out[54]: True

In [55]:
```
a
```

Out[55]: 5

In [56]:
```
b
```

Out[56]: 4

In [60]:
```
b=5
```

In [61]:
```
b
```

Out[61]: 5

In [62]:
```
a==b
```

Out[62]: True

In [64]:
```
a
```

Out[64]: 5

In [66]:
```
b=7
```

In [67]:
```
b
```

Out[67]: 7

In [ ]:

In [ ]:

In [68]:
```
a<8 and b<6
```

Out[68]: False

In [69]:
```
a<8 or b<6
```

Out[69]: True

In [70]:
```
a>8 and b>6
```

Out[70]: False

In [71]:
```
a>8 or b>6
```

Out[71]: True

In [72]:
```
x=False
x
```

Out[72]: False

In [73]:
```
not x
```

Out[73]: True

In [ ]:

unary operator

In [74]:
```
25
```

Out[74]: 25

In [75]:
```
bin(25)
```

Out[75]: '0b11001'

In [77]:
```
int(0b11001)
```

Out[77]: 25

In [ ]:

In [78]:
```
oct(25)
```

Out[78]: '0o31'

In [79]:
```
int(0o31)
```

Out[79]: 25

In [80]:
```
0x9
```

Out[80]: 9

In [81]:
```
0xa
```

Out[81]: 10

In [82]:
```
0xb
```

Out[82]: 11

In [83]:
```
0xc
```

Out[83]: 12

In [84]:
```
0xd
```

Out[84]: 13

In [85]:
```
0xe
```

Out[85]: 14

In [86]:
```
0xf
```

Out[86]: 15

In [87]:
```
0xg
```

```
  Cell In[87], line 1
    0xg
     ^
SyntaxError: invalid hexadecimal literal
```

In [ ]:
```

```

Bitwise operator

complement(~)

In [88]:
```
print(bin(12))
print(bin(13))
```

```
0b1100
0b1101
```

In [89]:
```
12
```

Out[89]: 12

In [90]:
```
~12
```

Out[90]: -13

In [91]:
```
~10
```

Out[91]: -11

In [92]:
```
12 & 13
```

Out[92]: 12

In [95]:
```
35^40
```

Out[95]: 11

left shift <<: We gain the bit <<

right shift >>: We loose the bit >>

In [96]:
```
12 << 2
```

Out[96]: 48

In [97]:
```
1<<2
```

Out[97]: 4

In [98]:
```
10<<2
```

Out[98]: 40

In [99]:
```
10<<3
```

Out[99]: 80

In [100…
```
5<<2
```

Out[100… 20

In [101…
```
100<<3
```

Out[101… 800

In [ ]:

In [102…
```
10>>2
```

Out[102… 2

In [103…
```
100>>3
```

Out[103… 12

In [ ]:

swap two variables

In [104…
```
a,b=5,6
```

In [108…
```
print(a)
print(b)
```

```
5
6
```

In [107…
```
print(a), print(b)
```

```
5
6
```

Out[107… (None, None)

In [109…
```
temp=a
a=b
b=temp
```

In [110…
```
=print(a)
print(b)
```

```
6
5
```

In [111…
```
a=a+b
b=a-b
a=a-b
```

In [112…
```
print(a)
print(b)
```

```
5
6
```

In [ ]:

In [113…
```
a=a^b
b=a^b
a=a^b
```

In [114…
```
print(a)
print(b)
```

```
6
5
```

In [115…
```
a,b=b,a
```

In [116…
```
print(a)
print(b)
```

```
5
6
```

In [ ]:

15th dec

In [3]:
```
pi=3.14
```

In [5]:
```
pi=2.3
pi
```

Out[5]: 2.3

In [ ]:

In [6]:
```
x=sqrt(25)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[6], line 1
----> 1 x=sqrt(25)

NameError: name 'sqrt' is not defined
```

In [12]:
```
import math
x=math.sqrt(25)
```

In [13]:
```
import math
```

In [16]:
```
x=math.sqrt(15)
```

In [14]:
```
math.floor(3.9)
```

Out[14]: 3

In [15]:
```
x=math.sqrt(15)
```

In [17]:
```
import math
x=math.sqrt(25)
```

In [18]:
```
math.sqrt(25)
```

Out[18]: 5.0

In [19]:
```
x=sqrt(25)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[19], line 1
----> 1 x=sqrt(25)

NameError: name 'sqrt' is not defined
```

In [22]:
```
import math
x=math.sqrt(25)
print(x)
```

```
5.0
```

In [23]:
```
import math
x=math.sqrt(15)
```

In [24]:
```
print(x)
```

```
3.872983346207417
```

In [25]:
```
math.pow(3,2)
```

Out[25]: 9.0

In [26]:
```
3**2
```

Out[26]: 9

In [27]:
```
math.pi
```

Out[27]: 3.141592653589793

In [28]:
```
math.e
```

Out[28]: 2.718281828459045

In [ ]:

In [29]:
```
import math as m
m.floor(3.4)
```

Out[29]: 3

In [30]:
```
from math import pow,floor,ceil
print(m.floor(2.3))
print(m.ceil(2.3))
print(m.pow(2,3))
```

```
2
3
8.0
```

In [ ]:

Input() functions

In [31]:
```
x=input()
x
```

Out[31]: 'HELLO TEAM'

In [ ]:

In [34]:
```
x=input()
y=input()
z=x+y
print(z)
```

```
22
```

In [ ]:

input() with numbers

In [36]:
```
x1=int(input('enter the 1st number:'))
y1=int(input('enter the 2nd number:'))
z1=x1+y1
print(z1)
```

```
30
```

In [ ]:

In [37]:
```
x1=(input('enter the 1st name:'))
y1=(input('enter the 2nd name:'))
z1=x1+y1
print(z1)
```

```
sunithavenu
```

In [ ]:

input() with string with index

In [38]:
```
st=input('enter a string')
print(st)
```

```
hello
```

In [39]:
```
print(st[0])
```

```
h
```

In [40]:
```
print(st[0:2])
```

```
he
```

In [41]:
```
print(st[1])
```

```
e
```

In [42]:
```
print(st[-1])
```

```
o
```

input() with eval()

In [ ]:
```
result=input('enter an expr')
print(result)
```

In [ ]:
```
result=eval(input('enter an expr'))
print(result)
```

In [ ]: