

```
In [1]: 7+8
```

```
Out[1]: 15
```

```
print('hello')
```

```
In [ ]: hello
```

```
In [3]: i=32  
i
```

```
Out[3]: 32
```

```
In [4]: type(i)
```

```
Out[4]: int
```

```
In [5]: f=110.32  
f
```

```
Out[5]: 110.32
```

```
In [7]: f1=1e0  
f1
```

```
Out[7]: 1.0
```

```
In [8]: f2=1e1  
f2
```

```
Out[8]: 10.0
```

```
In [9]: f3=1e2  
f3
```

```
Out[9]: 100.0
```

```
In [11]: a=20  
b=30  
a+b
```

```
Out[11]: 50
```

```
In [12]: a-b
```

```
Out[12]: -10
```

```
In [13]: print(a+b)  
print(a-b)  
print(a*b)
```

```
print(a/b)
print(a%b)
```

```
50
-10
600
0.6666666666666666
20
```

```
In [25]: num1=30
        num2=40
        add=num1+num2
        print('the addition of',num1, 'and',num2,'is', add)
```

the addition of 30 and 40 is 70

```
In [30]: num1=30
        num2=40
        add=num1+num2
        print('the addition of {} and {} is {}'.format (num1,num2, add))
```

the addition of 30 and 40 is 70

```
In [42]: num1=30
        num2=40
        num3=50
        add=num1+num2+num3
        print('the addition of {}, {} and {} is {}'.format(num1,num2,num3,add))
```

the addition of 30, 40 and 50 is 120

```
In [43]: num1=30
        num2=40
        num3=50
        add=num2+num3
        print('the addition of {},{} is {}'.format(num2, num3,add))
```

the addition of 40,50 is 90

```
In [45]: c= 2+3j
        c
```

Out[45]: (2+3j)

```
In [47]: type(c)
```

Out[47]: complex

```
In [48]: c=2+3j
        d=5+7j
        print(c+d)
        print(c-d)
```

```
(7+10j)
(-3-4j)
```

```
In [53]: import keyword
        keyword.kwlist
```

```
Out[53]: ['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 [55]: b1=False  
b1
```

```
Out[55]: False
```

```
In [56]: int(True)
```

```
Out[56]: 1
```

```
In [57]: int(False)
```

```
Out[57]: 0
```

```
In [58]: True+True
```

```
Out[58]: 2
```

```
In [59]: True+False
```

Out[59]: 1

In [60]: `False+False`

Out[60]: 0

In [1]: `True-False`

Out[1]: 1

In [2]: `False-True`

Out[2]: -1

In [3]: `True-True*False+False`

Out[3]: 1

In [61]: `p=q=r=30`
`q`

Out[61]: 30

In [62]: `p=q=r=30`
`r`

Out[62]: 30

In [63]: `print(id(p))`
`print(id(d))`
`print(id(r))`

140722614119768
2370479171504
140722614119768

In []: