```
>>> 0
>>> 10 + 20
>>> 10 + 2.0
>>> 10 * 20
>>> 10 * 2.0
>>> 10 - 20
>>> 20 / 5
>>> 20 / 5.0
>>> 20 / 7
>>> 20 / 7.0
>>>20 / 3.5
>>> 20 //  5
>>> 20 //  5.0
>>> 20 //  7
>>> 20 //  7.0
>>>20 // 3.5
>>>20 % 7
>>> 20 % 7.0
>>> 20 % 3.5
>>> 2 ** 4                    # exponent operator, 2 raised to 4
>>> 2 ** -2
>>> 2 ** -1
>>> divmod(10,3)              # (3, 1)
>>> divmod(10,2.5)            # (4.0, 0.0)
>>> divmod(15,4)              # (3, 3)
```

```
>>> int (4.5)

>>> int (‘4’)

>>> int (‘a’)        #error

>>> float(4.5)

>>> float(4)

>>> 4 == 4

>>> 4 == 5

>>> 4 is 4

>>>  4 is 5

>>>  1 == 1 == 1

>>>  1 == 1 == 2

>>>  A = 1 ==  1

>>>  A = 1 ==  2

>>>  True == 1

>>>  True == 0

>>>  False == 0

>>>  False == 1

>>>  1 != 2

>>>  1 > 2

>>>  1 >= 2

>>> 1 < 2 < 3

>>> 1 < 3 < 2

>>> 1 < 2 and 2 < 3

>>> 1 < 4 or 2 > 4

>>> any([1<2, 2<3])                          #True
```

```
>>> any([1<2, 2<1])              #True
>>> any([1<0, 2<1])              #False
>>> all([1<2, 2<3])              #True
>>> all([1<0, 2<3])              #False
>>> all([1<0, 2<1])              #False
>>> all([1<0, 2<3])              #False
>>> x=5
>>> 1 < x < 10                   #True
>>> 1 < x != 7                   #True
>>> 1 < x != 5                   #False
>>> a = 1
# check the values of a and b after executing each statement
>>> b=a++
>>> b = ++a
>>> b = a++1
>>> a+=1
>>> a-=1
>>> a*=2
>>> a/=2
>>> x = y = z = 10               # check values stored in x,y,z
>>> x, y, z = 10,20,'*'          # check values stored in x,y,z


>>> import string
>>> ord('a')                     # 97
>>> chr(97)                      # 'a'
```

```
>>> abs(-4)

>>> abs(5)

>>> round(1.5678,1)                #1.6

>>> round(1.5678,2)                #1.57

>>> round(1.5678)                  #2

>>> round(1.5678,6)                #1.5678

>>> round(1.5678, -1)              # 0.0

>>> round(1.5678, -2)              # 0.0

>>> round(15678.1234, -1)          # 15680.0

>>> round(15678.1234, -2)          # 15700.0

>>> round(15678.1234, -3)          # 16000.0

>>> round(15678.1234, -4)          # 20000.0
```