

【Python】输入输出 input().split(),map(),str.format()

一、输入

分开输入：

```
>>> a,b= input().split()
12 34
>>> a
'12'
>>> b
'34'
```

显示提示信息：

```
>>> n = int(input("请输入一个整型值N:"))
请输入一个整型值N:6
>>> n
6
```

CSDN @fftx_00

二、输出

```
>>> print("hello")
hello
>>> a=1
>>> print(a)
1
>>> b=2
>>> print(a, b)
1 2
```

end参数

```
>>> for i in range(5):
    print(i)

0
1
2
3
4
>>> for i in range(5):
    print(i, end=',')

01234
```

CSDN @fftx_00

三、一行输入输出多个变量

```
a, b, c, d = input().split() # 没法每次都分割出来的string再int()
# 因此只能map一下int()
print(a, b, c, d)
```

CSDN @fftx_00

因为int()只能执行一次返回一个值

```
x = int(input()) # input()产生的Iterable可迭代对象传递给int()函数

# map(function,iterable,.....)
a, b, c, d = map(int, input().split())

print(a, b, c, d)
```

CSDN @fftx_00

四、str.format()

format()函数

基本格式: str.format()

```
>>>x=3.14159
```

```
>>>y=2*x*3
```

```
>>>print("{0:.2f} {1:.2f}".format(x,y))
3.14 18.85
```

0和1表示format函数中的第一和第二个参数

.2f 表示小数部分保留两位，四舍五入

CSDN @fftx_00

```
>>> N = int(input())
10
>>> a = [index/(2*index-1) if index % 2 == 1 else -index/(2*index-1)
...      for index in range(1, N+1)]
>>> res = sum(a)
>>> "{:.3f}".format(res)
'0.380'
>>>
```

CSDN @fftx_00

0表示format()方法的参数下标，对应于第一个参数

```
print('{0}'.format(3.555))
```

.4f表示格式化为实数，保留4位小数

```
print('{0:.4f}'.format(10/3))
```

格式化为百分数字符串，总宽度为10，保留2位小数，>表示右对齐

```
print('{0:>10.2%}'.format(1/3))
```

逗号表示在数字字符串中插入逗号作为千分符，#x表示格式化为十六进制数

```
print("{0:,} in hex is: {0:#x}, in oct is {0:#o}".format(5555555))
```

可以先格式化下标为1的参数，再格式化下标为0的参数

o表示八进制数，但不带前面的引导符0o

```
print("{1} in hex is: {1:#x}, {0} in oct is {0:o}".format(6666, 66666))
```

_表示在数字中插入下划线作为千分符，#x表示格式化为十六进制数

```
print('{0:_},{0:#_x}'.format(10000000))
```

字符串前面加字符f，Python 3.6之后的版本支持这种用法

```
1 | width = 8 2 | height = 6
3 |
4 | print(f'Rectangle of {width}*{height}\nArea:{width*height}')
```