

The background of the slide is a light gray gradient, decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and subtle, scattered across the top and bottom edges of the frame.

# CS001 编程零基础 Python语言入门

第七讲

# 本讲内容

字符串格式化

字符串函数

字符串分析处理

## print函数

- 当输出多项时，自动添加分隔符，默认为空格（' '）。可以由参数**sep**指定分隔符。

```
#默认分隔符为空格
print('Hello', 'world')
#设置分隔符为逗号
print('Hello', 'world', sep=',')
#设置分隔符为空，即无分隔符
print('Hello', 'world', sep='')
```

## print函数

- 当输出完毕后，自动添加终结符，默认为换行。可以由参数end指定终结符。

```
#默认终结符为换行
print('Hello')
print('world')
#设置终结符为空格
print('Hello', end=' ')
print('world')
```

print2.py  
Print3.py

## 转义符

print4.py

以\开始的字符，有特殊含义

- \n: 新行符
- \t: 制表符
- \': 单引号
- \": 双引号
- \\: 反斜杠

## 格式字符串

在字符串中嵌入变量，使用格式符

- %d: 插入整数
- %f: 插入浮点数
- %s: 插入字符串



## 格式字符串： %d

print5.py

```
#print5.py
age=100
print("I'm %d years old"%age)
age=10
print("I'm %3d years old"%age)
print("I'm %-3d years old"%age)
```

```
I'm 100 years old
I'm 10 years old
I'm 10 years old
```

%d表示插入**整数**。

d前面的数字表示整数占据的**宽度**。若宽度超过实际需要，左边补空格。若宽度不足，按实际输出。宽度的默认值是0。

负号表示**左对齐**，右边补空格。

## 格式字符串: %f

print6.py

```
#print6.py
average=85.46
print("My score is %f"%average)
print("My score is %.1f"%average)
print("My score is %.0f"%average)
print("My score is %5.1f"%average)
print("My score is %3.1f"%average)
print(average)
```

```
My score is 85.460000
My score is 85.5
My score is 85
My score is 85.5
My score is 85.5
85.46
```

%f表示插入浮点数。

f前面的.n表示精度为n位，即四舍五入保留到小数点后n位。精度的默认值是6。

f前面也可以有表示宽度的数字，以及表示左对齐的负号。

当宽度和精度有矛盾时，保证精度。  
四舍五入不影响原来的数字。



## 格式字符串： %s

print7.py

```
#print7.py  
name='Tomas'  
print("I'm", name)  
print("I'm "+name)  
print("I'm %10s"%name)  
print("I'm %.3s"%name)
```

```
I'm Tomas  
I'm Tomas  
I'm      Tomas  
I'm Tom
```

%s表示插入字符串。

s前面也可以有表示宽度的数字，以及表示左对齐的负号。

s前面的.n表示取前n个字符。

## 格式字符串：多个变量及%%

print8.py

```
#print8.py
name='Tom'
age=10
print("I'm %s.\nI'm %d years old."%(name, age))
s="I'm older than %d%% of my classmates."%90
print(s)
```

转义符换行

多个变量必须  
放在圆括号里

格式字符串可以  
储存在变量里

%%表示一个%

```
I'm Tom.
I'm 10 years old.
I'm older than 90% of my classmates.
```

## 宽度可变

Pi.py

```
#Pi.py
import math
for i in range(10):
    print('%.*f' % (i, math.pi))
```

```
3
3.1
3.14
3.142
3.1416
3.14159
3.141593
3.1415927
3.14159265
3.141592654
```

## 打印三角形

```
#triangle2.py
n = 4
for i in range(n):
    s = '*' * (2*i+1)
    print('%*s' % (n+i, s))
```

#三角形的高度为n行  
#循环n遍，i从0到2  
#本行有2i+1个星号  
#输出字符串,宽度为n+i

```

    *
  ***
 *****
*****
```

# 测试函数

test.py

## 测试

- isalpha: 只包含字母
- isdigit: 只包含数字
- isalnum: 只包含字母或数字
- islower: 只包含小写字母
- isupper: 只包含大写字母
- isspace: 只包含空白字符（空格、制表符、换行符）

```
#test.py
print('Jack'.isalpha())
print('Jack Jones'.isalpha())
print('123'.isdigit())
print('123.4'.isdigit())
print('F22'.isalnum())
print('My phone number is 12345678.'.isalnum())
print('MIT'.isupper())
print('Japan'.islower())
```

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



# 大小写转换函数

case.py

## 大小写

- upper: 变大写
- lower: 变小写
- capitalize: 首字母大写
- title: 每个单词的首字母大写

```
#case.py  
s='hello, Tom.'  
print(s.upper())  
print(s.lower())  
print(s.capitalize())  
print(s.title())  
print(s)
```

```
HELLO, TOM.  
hello, tom.  
Hello, tom.  
Hello, Tom.  
hello, Tom.
```

原字符串  
并未改变

# 字符串搜索函数

## 搜索

- startswith
- endswith
- in
- index
- find

search.py

```
#search.py
s='peach, apple, banana, pear'
print(s.startswith('pe'))
print(s.endswith(','))
print('pe' in s)
print(s.index('pe'))
print(s.find('pe'))
print(s.find('pe', 1))
print(s.find('pe', 20))
```

```
True
False
True
0
0
19
-1
```

**index**和**find**都返回找到的位置（从0开始）。  
若找不到，**index**会出错，**find**则返回-1。

## 筛选出大写字母

string1.py

```
#string1.py
s = 'Tom studied in MIT. It\'s a famous university in USA.'
for c in s:
    if c.isupper():
        print(c, end='')
```

## 字符串中的数字

string2.py

```
#string2.py
list = []
s = input()
for c in s:
    if c.isdigit():
        list.append(c)
n=int(''.join(list))
print(n)
```

## 合并函数

join.py

- join: 将列表元素按指定分隔符合并成一个字符串

```
#join.py  
list = ['ab', 'c', 'def']  
print(' '.join(list))  
print(','.join(list))  
print(';'.join(list))
```

```
ab c def  
ab,c,def  
abcdef
```



## 拆分函数



- `split`: 将字符串按指定分隔符拆分成列表的多个元素

```
#split.py  
s='123.45,6.54'  
print(s.split(','))  
print(s.split('.'))  
print(s.split('45'))
```

```
['123.45', '6.54']  
['123', '45,6', '54']  
['123.', ',6.54']
```

## 最高分

Highest.py

```
#Highest.py
s = '80 Tom 90 Mike 85 Sam 90 Dora'
list = s.split(' ')           #按空格把字符串拆成列表
scores = list[::2]            #从0开始切片, 步长2, 得到成绩列表
names = list[1::2]            #从0开始切片, 步长2, 得到成绩列表
maxscore = max(scores)
print('Who got the highest score:')
for i in range(len(scores)):
    if scores[i] == maxscore:
        print(names[i])
```

# 复习阅读

- 课本第8章的8.1
- 课本第6章的6.4.1~6.4.4。

