NHẬP MÔN LẬPTRÌNH CHO KHOA HỌC DỮ LIỆU

Bài 3. Các thao tác cơ bản trong Python

Nội dung

1. Lệnh rẽ nhánh
2. Lệnh lặp
3. Hàm trong python
4. Bài tập

Lệnh rẽ nhánh

Cấu trúc rẽ nhánh if-else

```
if expression:
    # If-block
```

```
if expression:
    # If-block
else:
    # else-block
```

```
if expression:
    # If-block
elif 2-expression:
    # 2-if-block
elif 3-expression:
    # 3-if-block
...
elif n-expression:
    # n-if-block
```

```
if expression:
    # If-block
elif 2-expression:
    # 2-if-block
...
elif n-expression:
    # n-if-block
else:
    # else-block
```

Lệnh rẽ nhánh

Cấu trúc rẽ nhánh if-else

- Ví dụ 3.1:

```
var1 = 100
if var1==100:
   print ("1 - Nhan mot gia tri true")
   print (var1)
else:
   print ("1 - Nhan mot gia tri false")
   print (var1)
----------------
var2 = 0
if var2==200:
   print ("2 - Nhan mot gia tri true")
   print (var2)
else:
   print ("2 - Nhan mot gia tri false")
   print(var2)
print ("Good bye!")
```

Lệnh rẽ nhánh

Cấu trúc rẽ nhánh if-else

- Ví dụ 3.2:

```
var = 100
if var == 200:
   print ("1 - Nhan mot gia tri true")
   print (var)
elif var == 150:
   print ("2 - Nhan mot gia tri true")
   print (var)
elif var == 100:
   print ("3 - Nhan mot gia tri true")
   print (var)
else:
   print ("4 - Nhan mot gia tri false")
   print (var)
print ("Good bye!")
```

Lệnh rẽ nhánh

Lệnh if lồng nhau

- Cú pháp:

```
if bieu_thuc1:
    cac_lenh
    if bieu_thuc2:
        cac_lenh
    elif bieu_thuc3:
        cac_lenh
    else
        cac_lenh
elif bieu_thuc4:
    cac_lenh
else:
    cac_lenh
```

- Ví dụ 3.3:

```
var = 100
if var < 200:
    print ("Gia tri bieu thuc la nho hon 200")
    if var == 150:
        print ("Do la 150")
    elif var == 100:
        print ("Do la 100")
    elif var == 50:
        print ("Do la 50")
elif var < 50:
    print ("Gia tri bieu thuc la nho hon 50")
else:
    print ("Khong tim thay bieu thuc true")
```

Lệnh rẽ nhánh

Phép toán if

- Python có cách sử dụng lệnh if khá khác biệt so với ngôn ngữ khác.
- Cú pháp: A if <điều kiện> else B
- Giải thích: Nếu điều kiện trả về giá trị đúng thì phép toán trả về giá trị A, nếu điều kiện trả về giá trị sai thì phép toán trả về giá trị B.
- Ví dụ 3.4: Tìm giá trị lớn nhất:

```
A=5
B=7
X = A if A >B else B
print ('Gia tri lon nhat la:',X)
```

Lệnh lặp

While

```
while expression:
    # while-block
```

```
while expression:
    #while-block-1
    continue
    #while-block-2
```

```
while expression:
    # while-block
else:
    # else-block
```

- Chú ý:
  - Lặp while trong python tương đối giống trong các ngôn ngữ khác
  - Trong khối lệnh while (lệnh lặp nói chung) có thể dùng continue hoặc break để về đầu hoặc cuối khối lệnh
  - Khối "else" sẽ được thực hiện sau khi toàn bộ vòng lặp đã chạy xong
    - Khối này sẽ không chạy nếu vòng lặp bị "break"

Lệnh lặp

While

- Ví dụ 3.5: while không có else

```
count = 0
while (count <= 9):
   print ('So thu tu cua ban la:', count)
   count = count + 1
```

- Ví dụ 3.6: while có else

```
count = 0
while count < 5:
   print (count, " la nho hon 5")
   count = count + 1
else:
   print (count, " la khong nho hon 5")
```

Lệnh lặp

For

```
for variable_1, variable_2, .. variable_n in sequence:
    # for-block
```

```
for variable_1, variable_2, .. variable_n in sequence:
    # for-block
else:
    # else-block
```

- Vòng lặp for sử dụng để duyệt danh sách, khối else làm việc tương tự như ở vòng lặp while
- Dùng hàm range(a, b) để tạo danh sách gồm các số từ a đến b-1, hoặc tổng quát hơn là range(a, b, c)
  - trong đó c là bước nhảy

Lệnh lặp

For

- Ví dụ 3.7: for không có else

```
for i in range (0,10):
    print ('So thu tu la:',i)
```

- Ví dụ 3.8: while có else

```
for i in range(0,10):
    print ('So thu tu la:',i)
else:
    print ('la so cuoi cung')
```

Lệnh lặp

Lặp lồng nhau

- Ví dụ 3.9: Tìm số nguyên tố nhỏ hơn 100

```
i = 2
while(i < 100):
   j = 2
   while(j <= (i/j)):
      if not(i%j): break
      j = j + 1
   if (j > i/j) : print (i, " la so nguyen to")
   i = i + 1
```

Hàm

- Cú pháp khai báo hàm rất đơn giản

```
def <tên-hàm>(danh-sách-tham-số):
    <lệnh 1>
    ...
    <lệnh n>
```

- Ví dụ 3.10: hàm tính tích 2 số

```
def tich(a,b):
    return a*b
```

- Hàm trả về kết quả bằng lệnh return, nếu không trả về thì coi như trả về None

Hàm

- Hàm có thể chỉ ra giá trị mặc định của tham số

```
def tich(a, b = 1):
    return a*b
```

- Như vậy với hàm trên ta có thể gọi thực hiện nó:

```
print(tich(10, 20))      # 200
print(tich(10)) #        10
print(tich(a=5))         # 5
print(tich(b=6, a=5))    # 30
```

- Chú ý: các tham số có giá trị mặc định phải đứng cuối danh sách tham số

Hàm cơ bản

Hàm toán học

Hàm	Miêu tả
abs(x)	Trị tuyệt đối của x
ceil(x)	Số nguyên nhỏ nhất mà không nhỏ hơn x
cmp(x, y)	Trả về -1 nếu x < y, trả về 0 nếu x == y, hoặc 1 nếu x > y
exp(x)	Trả về e^x
fabs(x)	Giá trị tuyệt đối của x
floor(x)	Số nguyên lớn nhất mà không lớn hơn x
log(x)	Trả về lnx, với x> 0

Hàm cơ bản

Hàm toán học

Hàm	Miêu tả
log10(x)	Trả về log10(x), với x> 0 .
max(x1, x2,...)	Trả về số lớn nhất
min(x1, x2,...)	Trả về số nhỏ nhất
modf(x)	Trả về phần nguyên và phần thập phân của x. Cả hai phần có cùng dấu với x và phần nguyên được trả về dưới dạng một số thực
pow(x, y)	Trả về giá trị của x**y.
round(x [,n])	Làm tròn x về n chữ số sau dấu thập phân. Python làm tròn theo cách sau: round(0.5) là 1.0 và round(-0.5) là -1.0
sqrt(x)	Trả về căn bậc hai của x, với x > 0

Hàm cơ bản

Hàm chuỗi ký tự

Hàm	Miêu tả
capitalize()	Viết hoa chữ cái đầu tiên của chuỗi
center(width, fillchar)	Trả về một chuỗi mới, trong đó chuỗi ban đầu đã được cho vào trung tâm và hai bên đó là các fillchar sao cho tổng số ký tự của chuỗi mới là width
count(str, beg= 0,end=len(string))	Đếm xem chuỗi str này xuất hiện bao nhiêu lần trong chuỗi string hoặc chuỗi con của string nếu bạn cung cấp chỉ mục ban đầu start và chỉ mục kết thúc end
decode(encoding='UTF-8',errors='strict')	Giải mã chuỗi bởi sử dụng encoding đã cho
encode(encoding='UTF-8',errors='strict')	Trả về phiên bản chuỗi đã được mã hóa của chuỗi ban đầu. Nếu có lỗi xảy ra, thì chương trình sẽ tạo một ValueError trừ khi các lỗi này được cung cấp với ignore hoặc replace
endswith(suffix, beg=0, end=len(string))	Xác định xem nếu chuỗi string hoặc chuỗi con đã cho của string (nếu bạn cung cấp chỉ mục bắt đầu beg và chỉ mục kết thúc end) kết thúc với hậu tố suffix thì trả về true, nếu không thì phương thức này trả về false
find(str, beg=0 end=len(string))	Xác định xem chuỗi str có xuất hiện trong chuỗi string hoặc chuỗi con đã cho của string (nếu bạn cung cấp chỉ mục bắt đầu beg và chỉ mục kết thúc end), nếu xuất hiện thì trả về chỉ mục của str, còn không thì trả về -1
index(str, beg=0, end=len(string)	Tương tự như find(), nhưng tạo ra một ngoại lệ nếu str là

https://vietjack.com/python/string_index_trong_python.jsp

Hàm cơ bản

Hàm chuỗi ký tự

Hàm	Miêu tả
isalnum()	Trả về true nếu chuỗi có ít nhất một ký tự và tất cả ký tự là chữ-số. Nếu không hàm sẽ trả về false
isalpha()	Trả về true nếu chuỗi có ít nhất 1 ký tự và tất cả ký tự là chữ cái. Nếu không phương thức sẽ trả về false
isdigit()	Trả về true nếu chuỗi chỉ chứa các chữ số, nếu không là false
islower()	Trả về true nếu tất cả ký tự trong chuỗi là ở dạng chữ thường, nếu không là false
isnumeric()	Trả về true nếu một chuỗi dạng Unicode chỉ chứa các ký tự số, nếu không là false
isspace()	Trả về true nếu chuỗi chỉ chứa các ký tự khoảng trắng whitespace, nếu không là false
isupper()	Trả về true nếu tất cả ký tự trong chuỗi là chữ hoa
join(seq)	Nối chuỗi các biểu diễn chuỗi của các phần tử trong dãy seq thành một chuỗi

Hàm cơ bản

Hàm chuỗi ký tự

Hàm	Miêu tả
len(string)	Trả về độ dài của chuỗi
lower()	Chuyển đổi tất cả chữ hoa trong chuỗi sang kiểu chữ thường
lstrip()	Xóa tất cả các khoảng trống trắng ban đầu (leading) trong chuỗi
max(str)	Trả về ký tự chữ cái lớn nhất từ chuỗi str đã cho
min(str)	Trả về ký tự chữ cái nhỏ nhất từ chuỗi str đã cho
replace(old, new [, max])	Thay thế tất cả sự xuất hiện của old trong chuỗi với new với số lần xuất hiện max (nếu cung cấp)
upper()	Chuyển đổi các chữ thường trong chuỗi thành chữ hoa
title()	Trả về một bản sao của chuỗi trong đó tất cả ký tự đầu tiên của tất cả các từ là ở kiểu chữ hoa.

Hàm cơ bản

Hàm chuỗi ký tự

Hàm	Miêu tả
len(string)	Trả về độ dài của chuỗi
lower()	Chuyển đổi tất cả chữ hoa trong chuỗi sang kiểu chữ thường
lstrip()	Xóa tất cả các khoảng trống trắng ban đầu (leading) trong chuỗi
max(str)	Trả về ký tự chữ cái lớn nhất từ chuỗi str đã cho
min(str)	Trả về ký tự chữ cái nhỏ nhất từ chuỗi str đã cho
replace(old, new [, max])	Thay thế tất cả sự xuất hiện của old trong chuỗi với new với số lần xuất hiện max (nếu cung cấp)
upper()	Chuyển đổi các chữ thường trong chuỗi thành chữ hoa
title()	Trả về một bản sao của chuỗi trong đó tất cả ký tự đầu tiên của tất cả các từ là ở kiểu chữ hoa.


LOGO
CẢM ƠN!