

برنامه نویسی به زبان

# پایتون

مرجع کامل

تالیف

مهندس هادی کیامرثی

تمام مثال های موجود در این کتاب با کامپیوتر تست شده اند تا از هر گونه خطا  
مبرا باشند با این حال ممکن است باز هم خطاهایی در آن وجود داشته باشد از  
کلیه خوانندگان این کتاب ، اساتید و دانشجویان محترم خواهشمندم برای مطلع  
کردن مولف از این خطا ها لطفا با ایمیل آدرس زیر تماس بگیرید

hadikiamarsi@gmail.com

لازم به ذکر است کلیه حقوق مادی و معنوی این اثر برای مولف محفوظ می  
باشد و هرگونه کپی برداری و استفاده از محتویات این کتاب به هر نوعی تحت  
پیگرد قانونی قرار می گیرد

کلیه حقوق  
مدیریتی

# فصل سوم

فادی کیپیامدثی

## در این فصل مطالب زیر را خواهید آموخت

حلقه while در پایتون ( python )

حلقه بی نهایت

استفاده از دستور else در حلقه while

حلقه while یک خطی

حلقه for در پایتون ( python )

تکرار با استفاده از اندیس مرتب

استفاده از دستور else در حلقه for

حلقه های تو در تو در پایتون ( python )

دستور break در پایتون ( python )

دستور continue در پایتون ( python )

در برنامه نویسی موقعیت هایی پیش می آید که نیاز می باشد که اجرای یک کد تکرار گردد برای تکرار اجرای یک کد یا بلاکی از کدها در زبان های برنامه نویسی از حلقه ها استفاده می گردد . لازم به ذکر است که حلقه ها به صورت درونی دارای دستورات تصمیم گیری یا شرطی نیز می باشند . بنابراین با حلقه ها می توان یک دستور شرطی یا تصمیم گیری را پیاده سازی نمود . این فصل تمام آنچه نیاز دارید درباره حلقه ها در زبان برنامه نویسی پایتون ( python ) بدانید را توضیح می دهد

## حلقه while در پایتون ( python )

حلقه while در زبان برنامه نویسی پایتون ( python ) یک دسته کد را تا زمانی که شرط حلقه برقرار باشد تکرار می نماید

### راهنمای نحوی

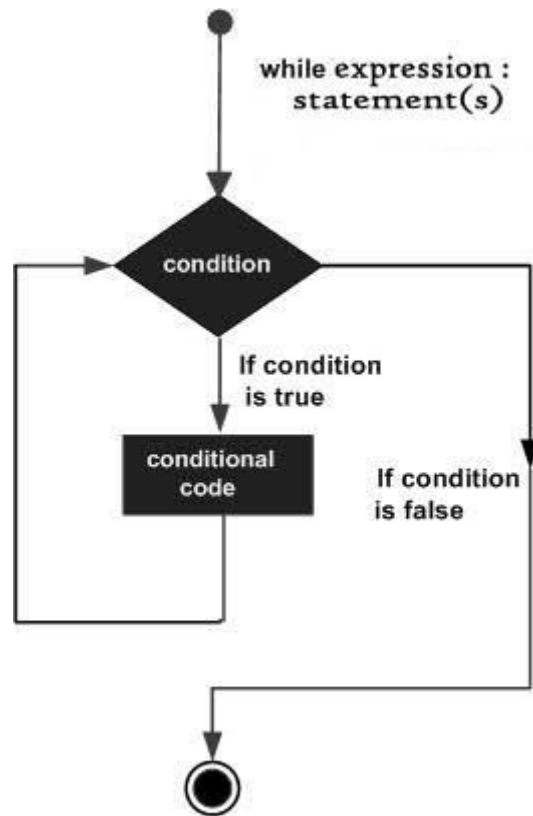
راهنمای نحوی حلقه while در زیر آورده شده است

```
while expression:  
    statement(s)
```

در راهنمای نحوی بالا (s) statement می تواند یک خط کد یا بلاکی از کدها باشد expression می تواند یک عبارت شرطی باشد و تکرار حلقه تا زمانی که عبارت شرطی برقرار باشد ادامه می یابد

زمانی که شرط حلقه نادرست ( false ) بشود روند اجرای برنامه از حلقه خارج شده و به اولین کد بعد از حلقه منتقل می گردد

در زبان برنامه نویسی پایتون ( python ) بلاک کدهای حلقه بوسیله تو رفتگی ( indentation ) مشخص می گردد در زبان برنامه نویسی پایتون ( python ) خط کدهایی که دارای تو رفتگی ( indentation ) هم اندازه باشند تشکیل یک بلاک کد می دهند



در حلقه ها زمانی که شرط حلقه برقرار نباشد ( false ) حلقه متوقف می گردد و روند اجرای کد برنامه به اولین خط کد بعد از حلقه انتقال می یابد.

```

#!/usr/bin/python

count = 0
while (count < 9):
    print 'The count is:', count
    count = count + 1

print "Good bye!"
    
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```

The count is: 0
The count is: 1
The count is: 2
The count is: 3
The count is: 4
The count is: 5
The count is: 6
The count is: 7
    
```

```
The count is: 8
Good bye!
```

در مثال بالا یک عبارت 9 بار بوسیله یک حلقه در خروجی نمایش داده می شود و زمانی حلقه متوقف می گردد که مقدار متغیر count یا مساوی و یا بزرگتر از 9 بشود و در هر بار اجرای حلقه مقدار متغیر count یک عدد افزایش می یابد .

## حلقه بی نهایت

یک حلقه زمانی تبدیل به حلقه بی نهایت می شود که عبارت شرطی حلقه همیشه برقرار باشد یا هیچوقت اشتباه ( false ) نباشد یک چنین حلقه ای هیچوقت متوقف نمی شود و تا بی نهایت ادامه می یابد

حلقه های بی نهایت کاربردهای زیادی در زمینه های ایجاد برنامه های سیستم عامل ، شبکه و برنامه نویسی سوکت ( سرور های تحت شبکه مانند ftp ، telnet ) و سرویسها دارند

```
#!/usr/bin/python

var = 1
while var == 1 : # This constructs an infinite loop
    num = raw_input("Enter a number :")
    print "You entered: ", num

print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
Enter a number :20
You entered: 20
Enter a number :29
You entered: 29
Enter a number :3
You entered: 3
Enter a number between :Traceback (most recent call last):
  File "test.py", line 5, in <module>
    num = raw_input("Enter a number :")
KeyboardInterrupt
```

در مثال بالا برای متوقف کردن حلقه بی نهایت و خروج از برنامه از کلیدهای CTRL+C استفاده می گردد

## استفاده از دستور else در حلقه while

زبان برنامه نویسی پایتون ( python ) از دستور else در حلقه ها پشتیبانی می نماید .

اگر در حلقه for از دستور else استفاده گردد زمانی کدهای بلاک else اجرا می گردد که روند اجرای برنامه از حلقه خارج بشود یا تکرار حلقه متوقف گردد .

اگر دستور else در حلقه while بکار رود بلاک دستورات else زمانی اجرا می گردند که عبارت شرطی حلقه برقرار نباشد

در مثال زیر نحوه استفاده از دستور else در حلقه while نشان داده شده است

```
#!/usr/bin/python

count = 0
while count < 5:
    print count, " is less than 5"
    count = count + 1
else:
    print count, " is not less than 5"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
0 is less than 5
1 is less than 5
2 is less than 5
3 is less than 5
4 is less than 5
5 is not less than 5
```

## حلقه while یک خطی

همانند دستور if اگر بلاک حلقه while تنها شامل یک خط کد باشد می توان آن را در یک خط پیاده سازی کرد

در زیر مثالی از حلقه while یک خطی آورده شده است

```
#!/usr/bin/python

flag = 1
while (flag): print 'Given flag is really true!'
print "Good bye!"
```

مثال بالا در یک حلقه بی نهایت می افتد و برای متوقف کردن آن و خروج از برنامه باید کلید های CTRL+C را فشار داده

## حلقه for در پایتون ( python )

دستور for هم یکی از دستورات تکرارساز یا حلقه ساز در زبان برنامه نویسی پایتون ( python ) می باشد

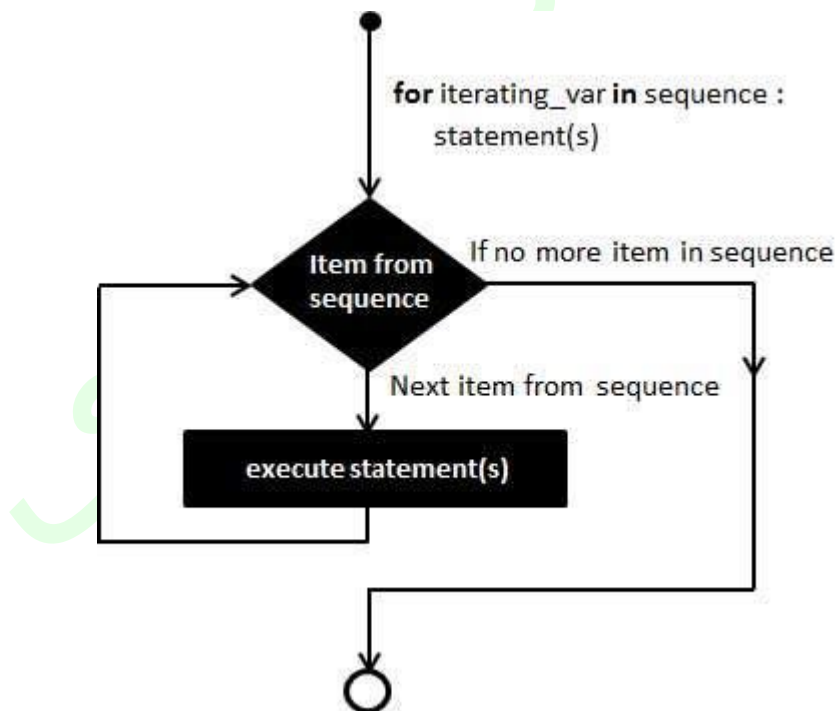


## راهنمای نحوی

```
for iterating_var in sequence:  
    statements(s)
```

در راهنمای نحوی بالا sequence یک لیست ( لیست ها در فصل های آینده توضیح داده خواهند شد ) می باشد که عناصر آن دونه به دونه در متغیر iterating\_var قرار می گیرد و حلقه تا زمانی که تمام عناصر لیست در متغیر iterating\_var قرار بگیرد تکرار می گردد و بعد از آن متوقف می گردد .

نمودار راهنما



```
#!/usr/bin/python

for letter in 'Python':    # First Example
    print 'Current Letter :', letter

fruits = ['banana', 'apple', 'mango']
for fruit in fruits:    # Second Example
    print 'Current fruit :', fruit

print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
Current Letter : P
Current Letter : y
Current Letter : t
Current Letter : h
Current Letter : o
Current Letter : n
Current fruit : banana
Current fruit : apple
Current fruit : mango
Good bye!
```

## تکرار با استفاده از اندیس مرتب

این قسمت نیازی به توضیح ندارد تنها برای روشن شدن این درس به مثال زیر توجه نمایید

```
#!/usr/bin/python

fruits = ['banana', 'apple', 'mango']
for index in range(len(fruits)):
    print 'Current fruit :', fruits[index]

print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
Current fruit : banana
Current fruit : apple
Current fruit : mango
Good bye!
```

در مثال بالا تابع `len` یک تابع پیش ساخته برای دریافت اندازه یک تاپل می باشد که در فصل های بعد بیشتر توضیح داده می شود و تابع `range` نیز برای درست کردن یک اندیس مرتب بکار رفته است .

## استفاده از دستور else در حلقه for

در زبان برنامه نویسی پایتون ( python ) حلقه for می تواند شامل دستور else باشد

اگر دستور else به همراه دستور for بکار رود بلاک دستوری else زمانی اجرا می شود که حلقه متوقف شده باشد .

اگر دستور else در حلقه while بکار رود بلاک دستورات else زمانی اجرا می گردند که عبارت شرطی حلقه برقرار نباشد

در مثال زیر نحوه استفاده از دستور else در حلقه for نشان داده شده است

```
#!/usr/bin/python
for num in range(10,20):      #to iterate between 10 to 20
    for i in range(2,num):    #to iterate on the factors of the number
        if num%i == 0:       #to determine the first factor
            j=num/i           #to calculate the second factor
            print '%d equals %d * %d' % (num,i,j)
            break #to move to the next number, the #first FOR
        else:                 # else part of the loop
            print num, 'is a prime number'
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
10 equals 2 * 5
11 is a prime number
12 equals 2 * 6
13 is a prime number
14 equals 2 * 7
15 equals 3 * 5
16 equals 2 * 8
17 is a prime number
18 equals 2 * 9
19 is a prime number
```

## حلقه های تو در تو در پایتون ( python )

در زبان برنامه نویسی ( python ) اجازه دارید یک حلقه را درون حلقه ی دیگری بکار ببرید به این حالت حلقه های تو در تو گفته می شود .

راهنمای نحوی

راهنمای نحوی حلقه تو در تو بوسیله for در زیر آورده شده است

```
for iterating_var in sequence:
    for iterating_var in sequence:
        statements(s)
statements(s)
```

راهنمای نحوی حلقه تو در تو بوسیله while در زیر آورده شده است

```
while expression:
    while expression:
        statement(s)
statement(s)
```

نکته آخر در مورد حلقه های تو در تو این می باشد که شما در زبان برنامه نویسی پایتون ( python ) اجازه دارید حلقه های تو در تو را از انواع مختلفی دستور حلقه بسازید . به عنوان نمونه می توانید حلقه while را درون حلقه for بکار ببرید

برای آشنایی بیشتر با نحوه تعریف و کاربرد حلقه های تو در تو به مثال زیر توجه نمایید

```
#!/usr/bin/python

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, " is prime"
    i = i + 1

print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
2 is prime
3 is prime
5 is prime
7 is prime
11 is prime
13 is prime
17 is prime
19 is prime
23 is prime
29 is prime
31 is prime
37 is prime
41 is prime
43 is prime
```

```
47 is prime
53 is prime
59 is prime
61 is prime
67 is prime
71 is prime
73 is prime
79 is prime
83 is prime
89 is prime
97 is prime
Good bye!
```

## دستور break در پایتون ( python )

در زبان برنامه نویسی پایتون ( python ) دستور Break برای متوقف کردن روند اجرای یک حلقه استفاده می گردد . دستور break برای هر دو حلقه for و while بکار می رود .

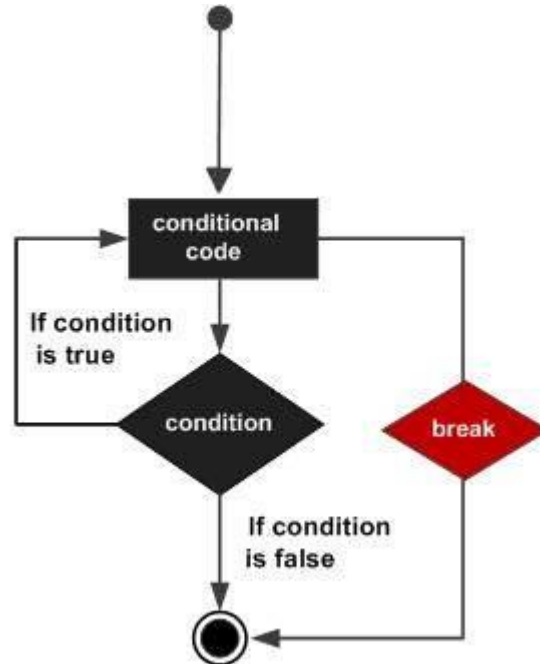
لازم به ذکر است که اگر دستور break را در یک حلقه تو در تو بکار ببرید ابتدا داخلی ترین حلقه را متوقف می نماید

### راهنمای نحوی

راهنمای نحوی دستور break در زیر آورده شده است

```
break
```

## نمودار راهنما



```
#!/usr/bin/python

for letter in 'Python':      # First Example
    if letter == 'h':
        break
    print 'Current Letter :', letter

var = 10                      # Second Example
while var > 0:
    print 'Current variable value :', var
    var = var - 1
    if var == 5:
        break

print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
Current Letter : P
Current Letter : y
Current Letter : t
Current variable value : 10
Current variable value : 9
Current variable value : 8
Current variable value : 7
Current variable value : 6
Good bye!
```

# دستور continue در پایتون ( python )

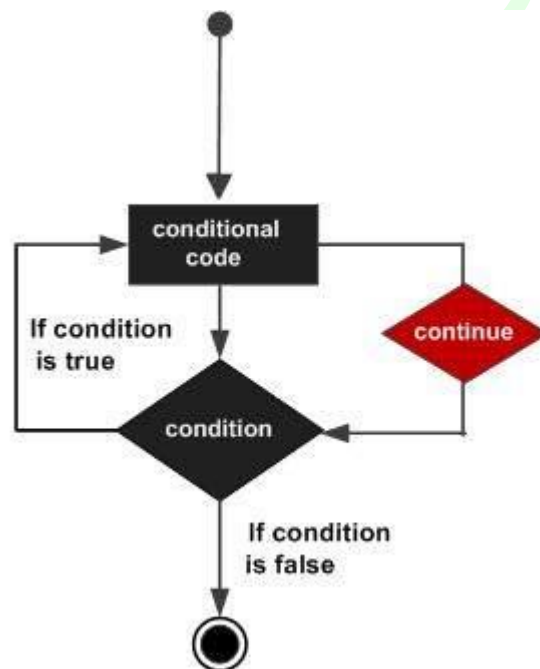
این دستور کنترل روند اجرای برنامه را به ابتدای حلقه منتقل می نماید . دستور continue کلیه دستورات توقف حلقه پیش از خود را خنثی می نماید .

دستور continue برای هر دو حلقه while و for بکار می رود

## راهنمای نحوی

```
continue
```

## نمودار راهنما



```
#!/usr/bin/python

for letter in 'Python':    # First Example
    if letter == 'h':
        continue
    print 'Current Letter :', letter
```

```
var = 10                                # Second Example
while var > 0:
    var = var -1
    if var == 5:
        continue
    print 'Current variable value :', var
print "Good bye!"
```

اجرای کد بالا نتیجه زیر را در صفحه خروجی ظاهر خواهد نمود

```
Current Letter : P
Current Letter : y
Current Letter : t
Current Letter : o
Current Letter : n
Current variable value : 9
Current variable value : 8
Current variable value : 7
Current variable value : 6
Current variable value : 4
Current variable value : 3
Current variable value : 2
Current variable value : 1
Current variable value : 0
Good bye!
```

پیامدستی