



پایتون

مقدماتی



فهرست:

- ۱ دستورات شرط
- ۲ دستور if
- ۲ نکات مهم در نوشتن دستور if
- ۵ دستورات شرط ساده و مرکب
- ۵ آشنایی با دستور if ... else
- ۶ نکات مهمی در رابطه با if ... else
- ۷ کاربرد دیگری از تابع int()
- ۸ استفاده از تابع int() برای تعیین بخش پذیری



دستورات شرط

در تمام برنامه‌هایی که تا به حال با هم نوشتیم مثلاً برنامه **دست گرمی دوم** در فاز قبل، با فشار کلید F5، اجرای دستورات از سطر اول شروع شده و یکی پس از دیگری به ترتیب و بدون هیچ قید و شرطی اجرا می‌شوند.

```

dastGarmy2Ph2.py
File Edit Format Run Options Window Help

a = input("Enter a chracter : ")
b = input("Enter a number :")
print( int(b) * a )

Ln: 7 Col: 0

```

اما در مواردی لازم است یک دستور تنها در صورتی اجرا شود که شرطی برقرار باشد و در صورت عدم برقراری آن شرط، دستور مورد نظر ما اجرا نخواهد شد.

یک مثال غیر کامپیوتری



معمولاً پدر «پایسان» در زمان خروج او از منزل برای رفتن به مدرسه می‌گوید:

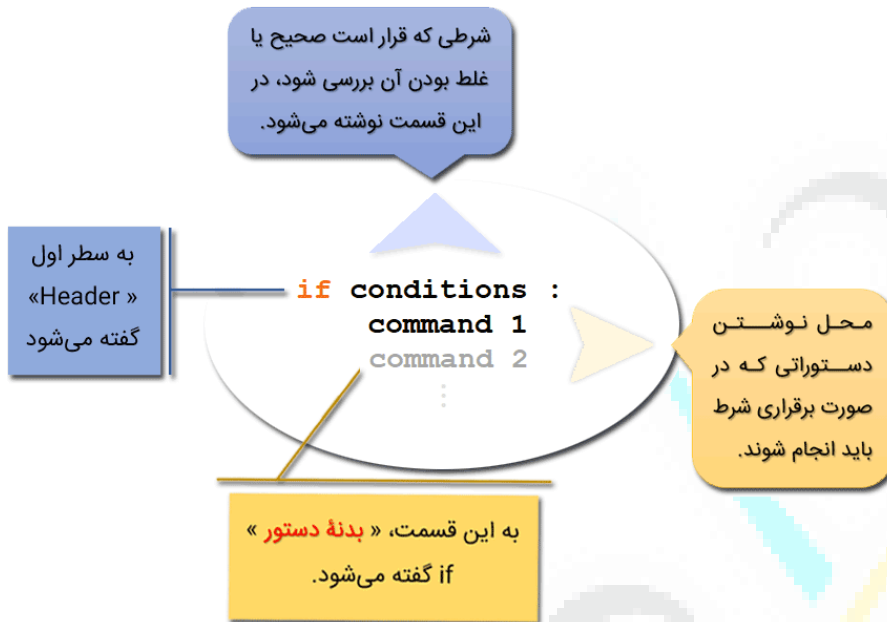
- کیف خود را بردار
- ظرف غذایت را بردار
- اگر هوا سرد است، یک لباس گرم هم بردار

در توصیه‌های پدر کاملاً مشخص است که پایسان قبل از خروج از منزل باید کیف و ظرف غذای خود را بردارد. اما برداشتن یک لباس گرم تنها در صورتی لازم است که هوا سرد باشد و در غیر این صورت یعنی گرم بودن هوا، لازم نیست پایسان لباس گرم همراه داشته باشد. عبارت سوم در واقع یک عبارت شرطی است و دستور موجود در آن تنها در صورت برقراری شرط انجام می‌شود.



دستور if

در زبان برنامه نویسی پایتون مانند سایر زبان‌های برنامه نویسی، از دستور if برای نوشتن عبارت شرطی استفاده می‌شود. ساختار کلی این دستور به صورت مقابل است. مشاهده می‌کنید که دستور if، یک دستور چند سطر است.



نکات مهم در نوشتن دستور if

- این دستور مانند سایر دستورات با حروف کوچک نوشته می‌شود.
- سطر اول که Header نام دارد، با علامت « : » خاتمه پیدا می‌کند.
- دستوراتی که در سطر دوم به

بعد یعنی در قسمت بدنه if نوشته می‌شوند، نسبت به سطر اول دارای یک تورفتگی یا Indent هستند و این تورفتگی نشان دهنده وابستگی این دستورات به سطر اول هستند.



- میزان تورفتگی یا Indent دستورات موجود در بدنه if به میزان یک Tab است. (Tab فاصله‌ای است که با فشار کلیدی به همین نام در صفحه کلید ایجاد می‌شود.) میزان این فاصله بسیار بسیار مهم است.





- در انتهای سطر اول از دستور `if` بعد از نوشتن علامت «:»، فشار کلید `Enter` باعث رفتن به سطر بعد می‌شود. این رفتن به سطر بعد هوشمندانه است یعنی `IDLE` به صورت خودکار فاصله مورد نیاز برای دستور سطر دوم را در نظر می‌گیرد.

یک مثال: می‌خواهیم برنامه‌ای بنویسیم که یک عدد دریافت کرده و بررسی کند که آیا این عدد از صفر بزرگتر است یا خیر.

```

17.py
File Edit Format Run Options Window Help
1 a = int(input("یک عدد وارد کنید: "))
2 if a > 0 :
3     print("بله")
4     print("عدد شما مثبت است")
5 print("*****")
Ln: 12 Col: 0

Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
>>>
>>> 10 : یک عدد وارد کنید
بله
عدد شما مثبت است
*****
>>>
>>> -10 : یک عدد وارد کنید
بله
عدد شما مثبت است
*****
>>>
Ln: 13 Col: 4
  
```

بررسی خطوط برنامه :

این دستور در داخل دستور if قرار ندارد.

- ① عددی را از کاربر گرفته و در متغیر `a` قرار می‌دهد.
- ② سطر اول از دستور `if` است و بزرگتر بودن مقدار `a` را از صفر بررسی می‌کند.
- ③ کلمه « بله » را در صورت مثبت بودن مقدار `a` چاپ می‌کند. این دستور قسمتی از دستور `if` است و تنها در صورت برقراری شرط اجرا می‌شود.
- ④ عبارت « عدد شما مثبت است » را در صورت مثبت بودن مقدار `a` چاپ می‌کند. این دستور نیز قسمتی از دستور `if` است و تنها در صورت برقراری شرط اجرا می‌شود.
- ⑤ این دستور تعدادی ستاره را بدون هیچ قید و شرطی چاپ می‌کند. یعنی دستوری است که بعد از دستور `if` قرار دارد. به همین دلیل نیز در امتداد دستور `if` نوشته شده است و تورفتگی یا `Indent` ندارد. این دستور، چه شرط دستور `if` صحیح باشد چه غلط باشد، اجرا خواهد شد چون خارج از بدنه `if` قرار دارد.

همانطور که در شکل قبل مشاهده می‌کنید، این مثال را دو بار اجرا کرده‌ایم. یک بار با عدد 10 که بزرگتر از صفر است و یک بار با عدد -10 که کوچکتر از صفر است. خروجی آنها را به دقت با یکدیگر مقایسه نمایید.



مثال دوم:

پایسان در بازار کامپیوتر یک لپ تاپ می‌بیند که دقیقاً برای نیاز او یعنی "برنامه نویسی" بسیار مناسب است. او ۴ میلیون تومان پس انداز کرده است، برنامه‌ای بنویسید که پایسان قیمت لپ تاپ را از فروشنده بپرسد و اگر قیمت آن مناسب بود، عبارت « **لطفاً یک لپ تاپ به من بدهید** » را نشان دهد.

```

18.py
File Edit Format Run Options Window Help
1 laptop = int(input("بخشید قیمت این لپ تاپ چقدر است "))
2 if laptop < 4000000 :
3     print("لطفاً یک لپ تاپ به من بدهید")
4     print("متشکرم")
Ln: 9 Col: 0

```

① قیمت لپ تاپ را دریافت کرده و در متغیر laptop قرار می‌دهد.

② سطر اول از دستور

if است و مقدار متغیر laptop را با عدد بودجه ای که داشتیم مقایسه می‌کند.

③ عبارت « **لطفاً یک لپ تاپ به من بدهید** » را در صورت ارزان تر بودن قیمت لپ تاپ از مبلغ ۴ میلیون تومان چاپ می‌کند. این دستور در واقع بدنه دستور if است و تنها در صورت برقراری شرط اجرا می‌شود.

④ عبارت « **متشکرم** » را بدون هیچ قید و شرطی چاپ می‌کند. یعنی دستوری است که بعد از دستور if قرار دارد. به همین دلیل نیز در امتداد دستور if نوشته شده است و تورفتگی یا Indent ندارد.

در شکل مقابل نتیجه دو بار اجرای مختلف این برنامه را می‌بینید:

```

Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
بخشید قیمت این لپ تاپ چقدر است 3400000
    لطفاً یک لپ تاپ به من بدهید
    متشکرم
>>>
بخشید قیمت این لپ تاپ چقدر است 5650000
    متشکرم
>>>
Ln: 12 Col: 4

```

در اجرای اول، قیمت لپ تاپ کمتر از ۴ میلیون است و پایسان درخواست یک لپ تاپ می‌کند و پس از آن عبارت « **متشکرم** » که دستور بعد از if است ظاهر می‌شود.

اما در اجرای دوم قیمت لپ تاپ بیشتر از ۴ میلیون است و پایسان دیگر درخواست خرید لپ تاپ نمی‌کند و تنها با عبارت « **متشکرم** » به این مکالمه خاتمه می‌دهد.



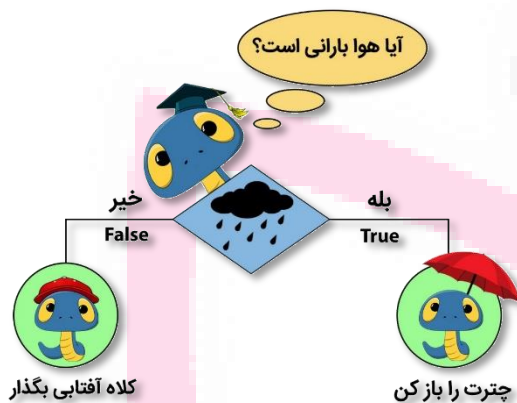
دستورات شرط ساده و مرکب

به دستورات شرطی که تا به حال بررسی کردیم، شرط ساده می‌گوییم. در این نوع شرط‌ها اگر شرط برقرار باشد، عمل یا عملیاتی انجام خواهد شد و در غیر این صورت، عملی انجام نمی‌شود و تنها اجرای برنامه به دستور بعد منتقل می‌شود.

مثلاً می‌گوییم: اگر هوا بارانی است چتر بردار!

اما دستور شرط مرکب دستوری است که اگر شرط برقرار باشد، عمل یا عملیاتی انجام می‌شود و در صورت برقرار نبودن شرط عمل یا عملیات دیگری انجام می‌شود. یعنی در این نوع دستور چه شرط درست باشد چه غلط باشد، عمل یا عملیاتی انجام می‌شود.

مثلاً می‌گوییم: اگر هوا بارانی است چترت را بردار **وگرنه** کلاهت را بردار



در این مثال اگر شرط برقرار باشد یعنی هوا بارانی باشد، باید چتر بردارد و اگر شرط برقرار نباشد، یعنی هوا بارانی نباشد، باید کلاه بردارد.

به کلمه **وگرنه** در این دستور توجه نمایید. این کلمه در دستورات شرط ساده وجود ندارد.

آشنایی با دستور `if ... else`

در دستورات شرط مرکب همانطور در شکل زیر مشاهده می‌کنید، دستور از دو قسمت اصلی تشکیل می‌شود. قسمت اول که کاملاً مشابه دستورات شرط ساده نوشته می‌شود و در صورت

برقرار بودن شرط انجام می‌شوند و قسمت دوم دستور که در صورت برقرار نبودن شرط انجام می‌شوند. این قسمت با عبارت « `else` » آغاز می‌شود و دستورات مربوط به آن در خطوط بعد و با یک تورفتگی یا `Indent` نوشته می‌شوند.

if conditions :

command

command

command

else :

command

command

command

در صورت صحیح
بودن
شرط انجام می‌شوند

در صورت صحیح
نبودن
شرط انجام می‌شوند

نکات مهمی در رابطه با `if ... else`

- این دستور باید در امتداد یا در راستای دستور `if` نوشته شود.
- در انتهای دستور `else` باید علامت `:` قرار داده شود.
- دستوراتی که در سطرهای بعد از `else` نوشته می‌شوند و به آن مربوط هستند، باید دارای یک تورفتگی یا `Indent` باشند تا محدوده این دستور مشخص شود.

یک مثال:

```
19.py
File Edit Format Run Options Window Help

a = int(input("یک عدد وارد کنید: "))
if a > 0 :
    print("عدد شما مثبت است")
else :
    print("عدد شما مثبت نیست")

Ln: 14 Col: 0
```

در شکل مقابل یک عدد از کاربر دریافت می‌شود.

اگر عدد بزرگتر است صفر باشد، عبارت « عدد شما مثبت است » چاپ می‌شود.

وگرنه عبارت « عدد شما مثبت نیست » چاپ می‌شود.

یک مثال دیگر: فرض کنید با تعدادی از همکلاسی‌های خود که همه هم‌سن هستید، می‌خواهید به شهربازی بروید. بلیت شهربازی برای هر نفر ۲۰ هزار تومان است اما برای افراد زیر ۱۲ سال نیم‌بها است یعنی بلیت آنها ۱۰ هزار تومان است. برنامه‌ای بنویسید که تعداد شما و همکلاسی‌هایتان که می‌خواهید وارد شهربازی شوید را دریافت کند، عدد سن شما را که باهم برابر است را هم دریافت کند، سپس جمع هزینه ورود شما به شهربازی را محاسبه نموده و به شما اعلام کند.

تعداد افراد را دریافت و در متغیر `n` قرار می‌دهد

سن افراد را دریافت و در متغیر `age` قرار می‌دهد

```
File Edit Format Run Options Window Help

n = int(input("شما چند نفر هستید: "))
age = int(input("شما چند سال دارید: "))

if age >= 12 :
    hazine = n * 20000
else :
    hazine = n * 10000

print("Hazine = ",hazine)
```

سن افراد را بررسی می‌کند که آیا بزرگتر یا مساوی ۱۲ هستند یا نه

در صورتی که سن افراد بزرگتر یا مساوی ۱۲ باشد اجرا شده و تعداد افراد را در هزینه هر بلیت که ۲۰ هزار تومان است ضرب می‌کند.

این دستور هزینه محاسبه شده را چاپ می‌کند. چه برای افراد بزرگتر از ۱۲ سال چه کوچکتر از ۱۲ سال. این دستور بعد از دستور `if` قرار دارد.

در صورتی که سن افراد کمتر از ۱۲ باشد اجرا شده و تعداد افراد را در هزینه هر بلیت که ۱۰ هزار تومان است ضرب می‌کند.



کاربرد دیگری از تابع `int()`

تا به حال از تابع `int()` برای تبدیل یک دادهٔ متنی که از ارقام تشکیل شده بود به یک دادهٔ عددی از نوع صحیح استفاده می‌کردیم.

```
Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
>>> 2 + int("20")
22
>>>
```

مانند مثال مقابل که با استفاده از تابع `int()` دادهٔ متنی "20" را به دادهٔ عددی 20 تبدیل کرده و آن را با عدد 2 جمع می‌کنیم.

و یا با استفاده از تابع `int()` عبارت دریافت شده توسط دستور `input()` که همواره یک عبارت متنی است را به یک دادهٔ عددی تبدیل می‌کنیم.

```
21.py
File Edit Format Run Options Window Help
a = input("Enter a number : ")
print( 3 * a )

a = int(input("Enter a number : "))
print( 3 * a )
```

```
Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
Enter a number : 10
101010

Enter a number : 10
30
>>>
```

در تمام مثال‌های قبل یک دادهٔ متنی به یک دادهٔ عددی تبدیل می‌شود.

اما دستور `int()` یک کاربرد دیگر نیز دارد و آن تبدیل یک دادهٔ عددی اعشاری به یک دادهٔ عددی صحیح است.

```
Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
>>> int(2.5)
2

>>> 10 / 3
3.3333333333333333

>>> int( 10 / 3 )
3
```

اگر داخل تابع `int()` یک عدد اعشاری نوشته شود، این تابع آن را به یک عدد صحیح تبدیل می‌کند. مانند دستور اول در شکل مقابل که عدد اعشاری 2.5 را به عدد صحیح 2 تبدیل کرده است.

اگر در داخل تابع `int()` یک عبارت محاسباتی نوشته شود که جواب آن یک عدد اعشاری باشد، این تابع تنها قسمت صحیح آن را نشان می‌دهد. به عنوان مثال جواب عبارت $10/3$ برابر با 3.3333333333333333 است اما اگر مانند شکل فوق، آن را در تابع `int()` قرار دهیم، تنها قسمت صحیح آن یعنی عدد 3 نمایش داده خواهد شد.



استفاده از تابع `int()` برای تعیین بخش پذیری

با استفاده از تابع `int()` به سادگی می‌توان بخش پذیر بودن یک عدد را بر عدد دیگر تعیین نمود. با دقت به مثال زیر توجه کنید:

چون دو طرف این عملگر با یکدیگر برابر است، عبارت `True` چاپ می‌شود.

```
Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
>>> 15 / 3
5.0
>>> int(15 / 3)
5
>>> 15 / 3 == int(15 / 3)
True
Ln: 18 Col: 4
```

یادآوری: این علامت یک عملگر مقایسه است که برابری عبارات دو طرف خود را با یکدیگر مقایسه می‌کند.

```
Python 3.6.6 Shell
File Edit Shell Debug Options Window Help
>>> 5 / 2
2.5
>>> int(5/2)
2
>>> 5/2 == int(5/2)
False
Ln: 18 Col: 4
```

چون دو طرف این عملگر با یکدیگر برابر نیست، عبارت `False` چاپ می‌شود.

مثال:

برنامه‌ای بنویسید که دو عدد صحیح دریافت کند، در متغیرهای `a` و `b` قرار دهد و مشخص نماید آیا `a` بر `b` بخش پذیر است یا خیر؟

دو عدد دریافت و در متغیرهای `a` و `b` قرار می‌دهد.

```
22.py
File Edit Format Run Options Window Help
a = int(input("Enter a number for a : "))
b = int(input("Enter a number for b : "))

if a / b == int(a / b):
    print(" Yes ")
else:
    print(" No ")
Ln: 13 Col: 0
```

با استفاده از تابع `int()` مشخص می‌نماید `a` بر `b` بخش پذیر است یا خیر

دست گرمی

برنامه‌ای بنویسید که یک عدد از کاربر دریافت کرده و مشخص نماید آن عدد زوج است یا فرد. در صورت زوج بودن عبارت `Even` بمعنی زوج و در غیر این صورت، عبارت `Odd` بمعنی فرد چاپ شود.