

# المتغيرات العددية Numbers

## أنواع الأعداد المتوفرة

int مثلاً 5  
long مثلاً 51924361L  
float مثلاً 14.20  
complex مثلاً 61.j

نستطيع أن نغير أي نوع عددي إلى نوع آخر بإستعمال الدالات التالية (إسم الدالة هو نفسه نوع العدد):

int() تحول أي متغير عددي إلى عدد صحيح  
مثلاً

```
print int(4.5) # gives 4
```

long() تحول أي متغير عددي إلى عدد طويل  
مثلاً

```
x = long(5)  
print type(x) # gives <type 'long'>
```

float() تحول أي متغير عددي إلى عدد عشري  
مثلاً

```
print float(4) # gives 4.0
```

لإيجاد نتيجة أي عملية مثلاً عملية القسمة بالنوع العشري يجب تغيير أحد أطراف العملية إلى عدد عشري

```
print 5/ float(2) # gives 2.5
```

complex() تحول أي متغير عددي إلى عدد مركب  
مثلاً

```
print complex(5) # gives (5+0j)
```

للتأكد من نوع أي متغير (جميع المتغيرات وليس الأعداد فقط ) إستعمل دائماً `type()` مثلاً

`print type(2.5)`

تعطينا

`<type 'float'>`

## الأدوات الحسابية

العلامة	المعنى	مثال
+	الجمع	<code>print 5 + 2 # gives 7</code>
-	الطرح	<code>print 5 - 2 # gives 3</code>
/	القسمة الصحيحة (تعطي عدد صحيح بدون فاصل إذا كان كل العددين صحيحين )	<code>print 5/2 # gives 2</code> لكن <code>print 5/float(2) # gives 2.5</code>
%	باقي القسمة	<code>print 5%2 # gives 1</code>
*	الضرب	<code>print 5*2 # gives 10</code>
//	القسمة مع التخلص بما وراء الفاصل في حالة كان أحد الأعداد عدد عشري	<code>print 5/2.0 # 2.5</code> Vs <code>print 5//2.0 # gives 2.0</code>
**	تعطي القوة	<code>print 5**2 # gives 25</code> نفس الشيء مثل <code>print pow(5,2)</code>

## أدوات المقارنة

بعد إستخدام أحد هذه الأدوات بين أي عددين النتيجة سو تكون منطقية `boolean: True /False`

العلامة	المعنى	مثال
==	تعطي صحيح في حالة التساوي	<code>print 5==4 # False</code> <code>print 5==5 # True</code>
!=	في حالة الاختلاف	<code>print 5!=4 # True</code>
<	في حالة العدد الأول أصغر من الثاني	<code>print 5 &lt; 4 # False</code>
>	في حالة العدد الأول أكبر من الثاني	<code>print 5 &gt; 4 # True</code>
<=	في حالة العدد الأول أصغر أو يساوي من الثاني	<code>print 5 &lt;= 4 # False</code> <code>print 5 &lt;= 5 # True</code>
>=	في حالة العدد الأول أكبر أو يساوي من الثاني	<code>print 5 &gt;= 4 # True</code> <code>print 5 &gt;= 5 # True</code>

هنالك نوع آخر من الأدوات وهو ما يسمى: bitwise operators لن أضعه هنا لأنه قد يربكم

## دالات تستخدم للقيام بعمليات حسابية

العملية	المعنى	مثال								
cmp()	تقارن بين عددين وترجع 1 إذا كان العدد الأول أكبر ، -1 إذا كان العدد الأول أصغر ، 0 إذا كان العددين متساويان	<pre>print cmp(4,3) # gives 1 print cmp(3,4) # gives -1 print cmp(4,4) # gives 0</pre> <table><tr><th>x ? y</th><th>Output</th></tr><tr><td>x &lt; y</td><td>-1</td></tr><tr><td>x == y</td><td>0</td></tr><tr><td>x &gt; y</td><td>1</td></tr></table>	x ? y	Output	x < y	-1	x == y	0	x > y	1
x ? y	Output									
x < y	-1									
x == y	0									
x > y	1									
ceil()	تعطي أقرب أكبر عدد صحيح. إذا كان مابعد الفاصل هو 0 ترجع العدد الصحيح من المدخل	<pre>import math print math.ceil(5.1)# gives 6 print math.ceil(5.0)# gives 5</pre>								
abs()	تعطي القيمة المطلقة من عدد ( نوع المتغير يبقى هو نفسه )	<pre>print abs(-4) # gives 4 print abs(-4.4) # gives 4.4</pre>								
min()	تعطي أصغر عدد من جميع المدخلات	<pre>print min(3,4,9,2,8) # gives 2</pre>								
max()	تعطي أكبر عدد من جميع المدخلات	<pre>print max(3,4,9,2,8) # gives 9</pre>								
pow()	القوة من عدد	<pre>print pow(2,3) # gives 2*2*2 = 8</pre>								
floor()	تعطي أقرب أصغر عدد صحيح لكن يبقى نوع المتغير عشري ( تتخلص من الفاصل )	<pre>import math print math.floor(4) # gives 4.0 print math.floor(4.5) #gives 4.0 print math.floor(4.9)# gives 4.0</pre>								
round()	تحول إلى أقرب عدد صحيح	<pre>print round(4) #gives 4.0 print round(4.1)# gives 4.0 print round(4.5)# gives 5.0 print round(4.9) #gives 5.0</pre>								
sqrt()	تعطي الجذر التربيعي من عدد	<pre>import math print math.sqrt(4) #gives 2.0</pre>								