Maximum and Minimum values for ints

Asked 10 years, 10 months ago Modified 3 months ago Viewed 1.6m times 1088

159

How do I represent minimum and maximum values for integers in Python? In Java, we have Integer.MIN_VALUE and Integer.MAX_VALUE. python integer

<mark>Share</mark> Follow

edited Apr 10 at 12:22 -Mateen Ulhaq **21.9k** • 16 • 83 • 124 asked Sep 30, 2011 at 1:01

bdhar 20.2k • 16 • 68 • 85

20

Note that in Python 3 the int type is basically the same as the long type in Python 2, so the idea of a maximum or minimum int disappears completely. It's basically irrelevant even on Python 2. – agf

Sep 30, 2011 at 1:42 23

@agf: it can be relevant in various way. For instance in any algorithm that require to save the min value found (like a sorting algorithm). The min value could be initialized at sys.maxint so it guarantees that any first value found is taken as min – Basile Perrenoud

Jan 9, 2014 at 15:30 2

@Toaster except that you can have a list where all values are greater than sys.maxint since it's only the maximum for the int type on Python 2, which Python will silently promote to a long. – agf

Jan 9, 2014 at 16:17 🧪 50

If you need to use "a very large value" in an algorithm, e.g. finding minimum or maximum of a generic collection, float('inf') or float('-inf') can be quite helpful. - geoff Oct 25, 2015 at 7:49

@geoff true, but one caveat for modern code is that floats can't be used as Literal in type hints. So you can't say that a list can contain Union[int, Literal[-inf]] even though that might be exactly what might be needed for a given application :/ – Christian Jul 13, 2020 at 16:57

 \mathbf{T}

9 Answers Sorted by:

Highest score (default)





Ð Python 3

In Python 3, this question doesn't apply. The plain int type is unbound.

However, you might actually be looking for information about the current interpreter's *word size*, which will be the same as the machine's word size in most cases. That information is still available in Python 3 as sys.maxsize, which is the maximum value representable by a signed word. Equivalently, it's the size of the largest possible list or in-memory <u>sequence</u>.

Generally, the maximum value representable by an unsigned word will be sys.maxsize * 2 + 1, and the number of bits in a word will be math.log2(sys.maxsize * 2 + 2). See this answer for more information.

Python 2

In Python 2, the maximum value for plain int values is available as sys.maxint :

>>> sys.maxint 9223372036854775807

You can calculate the minimum value with -sys.maxint - 1 as shown here.

Python seamlessly switches from plain to long integers once you exceed this value. So most of the time, you won't need to know it. Share

Follow edited Mar 10 at 12:12
 Neuron

 4,506 • 4 • 32 • 53
 answered Sep 30, 2011 at 1:03 0 senderle

138k • 35 • 205 • 231

26 This number may appear to be arbitrary, but it isn't. 9223372036854775807 is exactly 2^63 - 1, so you've got a 64-bit int. In general, an n-bit integer has values ranging from -2^(n-1) to 2^(n-1) - 1 . - NullUserException

Sep 30, 2011 at 1:18 29

Note that if you're using a 32-bit Python runtime, sys.maxint will return 2^31 - 1, even though Python will jump to 64-bit seamlessly with the long datatype.

– Scott Stafford Feb 26, 2014 at 16:19 🧪

29

Use sys.maxsize instead, as suggested by @Akash Rana. It is present also in Python 2, as sys docs say. This will make the code more compatible with both Python versions. -0_ Oct 27, 2015 at 20:54 🧪

)

You and I have different interpretations of that line from the docs. The replacement in 2to3 is a fine quick-and-dirty heuristic that won't break anything most of the time -- but the difference between these two values matters. The best practice is to use the value you actually mean to use. If you *truly need* sys.maxint in Python 2, you won't need it anymore in Python 3, and it should really be removed entirely, not changed to sys.maxsize . – senderle

Nov 1, 2015 at 12:51

minsize - Multiplying with Bitwise min operator gives minsize ~sys.maxsize - om471987

Oct 14, 2016 at 15:33

428 ▼ •

If you just need a number that's bigger than all others, you can use

float('inf')

in similar fashion, a number smaller than all others: float('-inf')

This works in both python 2 and 3. Share Follow

answered May 18, 2016 at 12:58 Melle 6,701 • 1 • 28 • 31

6 Just a note tho (as irrelevant it is, but still): float('inf') > float('inf') results in 'false'. Infinite number should be bigger than another infinite number :-D ... *mind snaps* – Scre

Jun 28, 2017 at 14:49

31

@Scre What else would you expect? x > x is usually False , and infinity should be no exception. (float('NaN) , on the other hand...) – jamesdlin

S	ep 6, 2017 at 2:03
1 T - Se	his actually doesn't apply for int cauze cannot convert infinite float to intbut works for most cases Leighton ep 23, 2017 at 23:16
} @ D	Scre "In comparison operations, positive infinity is larger than all values except itself and NaN, and negative infinity is smaller than all values except itself and NaN." hu.org/software/libc/manual/html_node/Infinity-and-NaN.html Nathan ec 7, 2017 at 5:38
13 T - N	his is not an answer to the OP question <mark>ghosh</mark> ov 28, 2019 at 12:28 🎤
28	5
Tł	A sys.maxint constant has been removed from Python 3.0 onward, instead use sys.maxsize .
	 PEP 237: Essentially, long renamed to int. That is, there is only one built-in integral type, named int; but it behaves mostly like the old long type. PEP 238: An expression like 1/2 returns a float. Use 1//2 to get the truncating behavior. (The latter syntax has existed for years, at least since Python 2.2.) The sys.maxint constant was removed, since there is no longer a limit to the value of integers. However, sys.maxsize can be used as an integer larger than any practical list or string index. It conforms to the implementation's "natural" integer size and is typically the same as sys.maxint in previous releases on the same platform (assuming the same build options). The repr() of a long integer doesn't include the trailing L anymore, so code that unconditionally strips that character will chop off the last digit instead. (Use str() instead.) Octal literals are no longer of the form 0720; use 0o720 instead.
Re <mark>Sh</mark> Fo	efer : <u>https://docs.python.org/3/whatsnew/3.0.html#integers</u> are llow
e	dited Dec 17, 2019 at 2:02 martineau 115k • 24 • 159 • 280
a 1	nswered Sep 30, 2014 at 3:12 Akash Rana 3,447 • 2 • 18 • 28
} D	orrect. Indeed, from help(sys) : <i>maxsize the largest supported length of containers</i> . This should be the accepted answer. <mark>Marco Sulla</mark> ec 29, 2019 at 21:53
I ; - M	guess that the correct answer depends on the use case: in my casa (a default for a limiting parameter in a function) this was indeed the best answer, YMMV. Francesco Marchetti-Stasi far 13 at 11:12
96	
Fc	or Python 3, it is
im ma mi <mark>Sh</mark>	port sys axSize = sys.maxsize nSize = -sys.maxsize - 1 are
Fo e	llow dited Oct 15, 2021 at 12:43 Felix An
a	135 • 2 • 10 nswered Mar 2, 2017 at 0:09
58	netskink 3,382 • 1 • 28 • 40
w <i>in</i> Ju	rell, python 3 does <i>exist</i> , thankfully(!); but sys.maxint doesn't exist in python 3 (tl;dr: <i>" sys.maxint constant was removed (in python3), since there is no longer a limit to the value of</i> ntegers. However, sys.maxsize can be used as an integer larger than any practical list or string index.") michael Il 16, 2017 at 0:33
' - Ju 3	Thy create variables that shadow builtins such as min() and max() ? <mark>RoadRunner</mark> Il 12, 2018 at 0:38
, 	ook up 2's compliment binary netskink un 6, 2019 at 14:11
r - N	nin = ~sys.maxsize Andrew ov 8, 2019 at 4:03
Tl m – M	hese would work great as sentinel values in algorithms, which is a common usage. Please don't delete this answer, just make it clear that it's the most practical one, even if not the most Nathematically correct one. Abhijit Sarkar Nar 7, 2021 at 11:41 🎤
90	
In <u>sy</u> 922 922 922 Fr	Python integers will automatically switch from a fixed-size int representation into a variable width <u>long</u> representation once you pass the value (x.maxint), which is either 2 ³¹ - 1 or 2 ⁶³ - 1 depending on your platform. Notice the L that gets appended here: > 9223372036854775807 > 9223372036854775808 23372036854775808 23372036854775808 23372036854775808L om the Python manual:
	Numbers are created by numeric literals or as the result of built-in functions and operators. Unadorned integer literals (including binary, hex, and octal numbers) yield plain integers unless the value they denote is too large to be represented as a plain integer, in which case they yield a long integer. Integer. Integer literals with an 'L' or 'l' suffix yield long integers ('L' is preferred because 11 looks too much like eleven!).
Py	thon tries very hard to pretend its integers are mathematical integers and are unbounded. It can, for instance, calculate a <u>googol</u> with ease:
100 Sh Fo e	are llow dited Sep 30, 2011 at 1:12
a	nswered Sep 30, 2011 at 1:07 John Kugelman 333k • 66 • 507 • 556
τς - Se 2	o add to the confusion, Python's long isn't like Java's long - it's rather closer to BigInteger . NullUserException ep 30, 2011 at 1:20 🎤
Ir – Fe	n python3, seems there is no L suffix, and it's just int , not long , no matter how large the number is. Eric eb 19, 2020 at 13:18 🎤

may i suggest editing your answer showing type(...) instead on relying on an L at the end which an have ambiguous meaning? – Coder

Jun 2, 2021 at 0:29 @JohnD What is the ambiguity? – John Kugelman Jun 2, 2021 at 12:04 28 ▼ • You may use 'inf' like this: import math bool_true = 0 < math.inf bool_false = 0 < -math.inf Refer: <u>math — Mathematical functions</u> Share Follow edited Sep 27, 2019 at 9:10 Georgy 10.4k • 7 • 59 • 67 answered Jul 28, 2019 at 10:01

499 • 8 • 11 Note that math.inf is equivalent to float('inf') – Georgy

Sep 27, 2019 at 9:10

Rahul Nimbal

The author questioned how to obtain the MAX and MIN int values. How does this answer relates to the question, since the results are True and False, not MAX and MIN? – Gilberto Albino Mar 28 at 11:07



If you want the max for array or list indices (equivalent to size_t in C/C++), you can use numpy: np.iinfo(np.intp).max

This is same as sys.maxsize however advantage is that you don't need import sys just for this. If you want max for native int on the machine:

np.iinfo(np.intc).max You can look at other available types in <u>doc</u>.

For floats you can also use sys.float_info.max .

Follow answered Jan 15, 2019 at 5:30 Shital Shah 57k • 12 • 222 • 178

e.g. for the maximum 32-bit int use: np.iinfo(np.int32).max – Contango Jun 6 at 9:46



Ð I rely heavily on commands like this.

python -c 'import sys; print(sys.maxsize)' Max int returned: 9223372036854775807

For more references for 'sys' you should access

https://docs.python.org/3/library/sys.html

<u>https://docs.python.org/3/library/sys.html#sys.maxsize</u> Share Follow edited Aug 19, 2018 at 12:21 Amit Dube 143 • 5 • 8

answered Jun 10, 2018 at 23:49

Wender 955 • 14 • 23

Apr 28, 2019 at 16:55

No - maxsize is simply the largest possible container index. Python will happily work with 100 digit integers and more – Tony Suffolk 66



2

Ð

sys.maxsize is not the actually the maximum integer value which is supported. You can double maxsize and multiply it by itself and it stays a valid and correct value. However, if you try sys.maxsize ** sys.maxsize , it will hang your machine for a significant amount of time. As many have pointed out, the byte and bit size does not seem to be relevant because it practically doesn't exist. I guess python just happily expands it's integers when it needs more memory space. So in general there is no limit.

Now, if you're talking about packing or storing integers in a safe way where they can later be retrieved with integrity then of course that is relevant. I'm really not sure about packing but I know python's pickle module handles those things well. String representations obviously have no practical limit.

So really, the bottom line is: what is your applications limit? What does it require for numeric data? Use that limit instead of python's fairly nonexistent integer limit. Share Follow

edited Jun 20, 2020 at 17:06
 Run_Script

 2,352
 2
 13
 26
 answered Jun 20, 2020 at 14:56

Ismael Harun **123** • 2 • 7

à Highly active question. Earn 10 reputation (not counting the association bonus) in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.



By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our Cookie Policy.

Accept all cookies Customize settings