

# Integer Objects

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 1)**

Unknown directive type "highlight".

```
.. highlight:: c
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 8)**

Unknown directive type "index".

```
.. index:: object: long integer
          object: integer
```

All integers are implemented as "long" integer objects of arbitrary size.

On error, most `PyLong_As*` APIs return `(return type)-1` which cannot be distinguished from a number. Use `c:func:PyErr_Occurred` to disambiguate.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 13); [backlink](#)**

Unknown interpreted text role "c:func".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 16)**

Unknown directive type "c:type".

```
.. c:type:: PyLongObject

This subtype of :c:type:`PyObject` represents a Python integer object.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 21)**

Unknown directive type "c:var".

```
.. c:var:: PyTypeObject PyLong_Type

This instance of :c:type:`PyTypeObject` represents the Python integer type.
This is the same object as :class:`int` in the Python layer.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 27)**

Unknown directive type "c:function".

```
.. c:function:: int PyLong_Check(PyObject *p)

Return true if its argument is a :c:type:`PyLongObject` or a subtype of
:c:type:`PyLongObject`. This function always succeeds.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 33)**

Unknown directive type "c:function".

```
.. c:function:: int PyLong_CheckExact(PyObject *p)
```

Return true if its argument is a `:c:type:PyLongObject`, but not a subtype of `:c:type:PyLongObject`. This function always succeeds.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 39)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromLong(long v)
```

Return a new `:c:type:PyLongObject` object from `*v`, or `NULL` on failure.

The current implementation keeps an array of integer objects for all integers between `-5` and `256`. When you create an int in that range you actually just get back a reference to the existing object.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 48)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromUnsignedLong(unsigned long v)
```

Return a new `:c:type:PyLongObject` object from a C `:c:type:unsigned long`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 54)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromSsize_t(Py_ssize_t v)
```

Return a new `:c:type:PyLongObject` object from a C `:c:type:Py_ssize_t`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 60)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromSize_t(size_t v)
```

Return a new `:c:type:PyLongObject` object from a C `:c:type:size_t`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 66)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromLongLong(long long v)
```

Return a new `:c:type:PyLongObject` object from a C `:c:type:long long`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 72)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromUnsignedLongLong(unsigned long long v)
```

Return a new `:c:type:PyLongObject` object from a C `:c:type:unsigned long long`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 78)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromDouble(double v)
```

Return a new `:c:type:PyLongObject` object from the integer part of `*v`, or `NULL` on failure.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 84)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromString(const char *str, char **pend, int base)
```

Return a new `:c:type:PyLongObject` based on the string value in `*str`, which is interpreted according to the radix in `*base`. If `*pend` is non-`NULL`, `*pend` will point to the first character in `*str` which follows the representation of the number. If `*base` is `0`, `*str` is interpreted using the `:ref: integers` definition; in this case, leading zeros in a non-zero decimal number raises a `:exc: ValueError`. If `*base` is not `0`, it must be between `2` and `36`, inclusive. Leading spaces and single underscores after a base specifier and between digits are ignored. If there are no digits, `:exc: ValueError` will be raised.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 97)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromUnicodeObject(PyObject *u, int base)
```

Convert a sequence of Unicode digits in the string `*u` to a Python integer value.

```
.. versionadded:: 3.3
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 105)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyLong_FromVoidPtr(void *p)
```

Create a Python integer from the pointer `*p`. The pointer value can be retrieved from the resulting value using `:c:func:PyLong_AsVoidPtr`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 112)**

Unknown directive type "c:function".

```
.. c:function:: long PyLong_AsLong(PyObject *obj)
```

```
.. index::
    single: LONG_MAX
    single: OverflowError (built-in exception)
```

Return a C `:c:type:long` representation of `*obj`. If `*obj` is not an instance of `:c:type:PyLongObject`, first call its `:meth: __index__` method (if present) to convert it to a `:c:type:PyLongObject`.

Raise :exc:`OverflowError` if the value of \*obj\* is out of range for a :c:type:`long`.

Returns ``-1`` on error. Use :c:func:`PyErr\_Occurred` to disambiguate.

.. versionchanged:: 3.8  
Use :meth:`\_\_index\_\_` if available.

.. versionchanged:: 3.10  
This function will no longer use :meth:`\_\_int\_\_`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 134)**

Unknown directive type "c:function".

.. c:function:: long PyLong\_AsLongAndOverflow(PyObject \*obj, int \*overflow)

Return a C :c:type:`long` representation of \*obj\*. If \*obj\* is not an instance of :c:type:`PyLongObject`, first call its :meth:`\_\_index\_\_` method (if present) to convert it to a :c:type:`PyLongObject`.

If the value of \*obj\* is greater than :const:`LONG\_MAX` or less than :const:`LONG\_MIN`, set \*overflow\* to ``1`` or ``-1``, respectively, and return ``-1``; otherwise, set \*overflow\* to ``0``. If any other exception occurs set \*overflow\* to ``0`` and return ``-1`` as usual.

Returns ``-1`` on error. Use :c:func:`PyErr\_Occurred` to disambiguate.

.. versionchanged:: 3.8  
Use :meth:`\_\_index\_\_` if available.

.. versionchanged:: 3.10  
This function will no longer use :meth:`\_\_int\_\_`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 154)**

Unknown directive type "c:function".

.. c:function:: long long PyLong\_AsLongLong(PyObject \*obj)

.. index::  
single: OverflowError (built-in exception)

Return a C :c:type:`long long` representation of \*obj\*. If \*obj\* is not an instance of :c:type:`PyLongObject`, first call its :meth:`\_\_index\_\_` method (if present) to convert it to a :c:type:`PyLongObject`.

Raise :exc:`OverflowError` if the value of \*obj\* is out of range for a :c:type:`long long`.

Returns ``-1`` on error. Use :c:func:`PyErr\_Occurred` to disambiguate.

.. versionchanged:: 3.8  
Use :meth:`\_\_index\_\_` if available.

.. versionchanged:: 3.10  
This function will no longer use :meth:`\_\_int\_\_`.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 175)**

Unknown directive type "c:function".

.. c:function:: long long PyLong\_AsLongLongAndOverflow(PyObject \*obj, int \*overflow)

Return a C :c:type:`long long` representation of \*obj\*. If \*obj\* is not an instance of :c:type:`PyLongObject`, first call its :meth:`\_\_index\_\_` method (if present) to convert it to a :c:type:`PyLongObject`.

If the value of `*obj*` is greater than `:const:`LLONG_MAX`` or less than `:const:`LLONG_MIN``, set `*overflow*` to ``1`` or ``-1``, respectively, and return ``-1``; otherwise, set `*overflow*` to ``0``. If any other exception occurs set `*overflow*` to ``0`` and return ``-1`` as usual.

Returns ``-1`` on error. Use `:c:func:`PyErr_Occurred`` to disambiguate.

.. versionadded:: 3.2

.. versionchanged:: 3.8  
Use `:meth:`__index__`` if available.

.. versionchanged:: 3.10  
This function will no longer use `:meth:`__int__``.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]long.rst, line 197)**

Unknown directive type "c:function".

.. c:function:: Py\_ssize\_t PyLong\_AsSsize\_t(PyObject \*pylong)

.. index::  
single: PY\_SSIZE\_T\_MAX  
single: OverflowError (built-in exception)

Return a C `:c:type:`Py_ssize_t`` representation of `*pylong*`. `*pylong*` must be an instance of `:c:type:`PyLongObject``.

Raise `:exc:`OverflowError`` if the value of `*pylong*` is out of range for a `:c:type:`Py_ssize_t``.

Returns ``-1`` on error. Use `:c:func:`PyErr_Occurred`` to disambiguate.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]long.rst, line 212)**

Unknown directive type "c:function".

.. c:function:: unsigned long PyLong\_AsUnsignedLong(PyObject \*pylong)

.. index::  
single: ULONG\_MAX  
single: OverflowError (built-in exception)

Return a C `:c:type:`unsigned long`` representation of `*pylong*`. `*pylong*` must be an instance of `:c:type:`PyLongObject``.

Raise `:exc:`OverflowError`` if the value of `*pylong*` is out of range for a `:c:type:`unsigned long``.

Returns ``(unsigned long)-1`` on error.  
Use `:c:func:`PyErr_Occurred`` to disambiguate.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]long.rst, line 228)**

Unknown directive type "c:function".

.. c:function:: size\_t PyLong\_AsSize\_t(PyObject \*pylong)

.. index::  
single: SIZE\_MAX  
single: OverflowError (built-in exception)

Return a C `:c:type:`size_t`` representation of `*pylong*`. `*pylong*` must be an instance of `:c:type:`PyLongObject``.

Raise `:exc:`OverflowError`` if the value of `*pylong*` is out of range for a `:c:type:`size_t``.

Returns ``(size_t)-1`` on error.

Use `:c:func:`PyErr_Occurred`` to disambiguate.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 244)**

Unknown directive type "c:function".

```
.. c:function:: unsigned long long PyLong_AsUnsignedLongLong(PyObject *pylong)

.. index::
   single: OverflowError (built-in exception)

Return a C :c:type:`unsigned long long` representation of *pylong*. *pylong*
must be an instance of :c:type:`PyLongObject`.

Raise :exc:`OverflowError` if the value of *pylong* is out of range for an
:c:type:`unsigned long long`.

Returns ``(unsigned long long)-1`` on error.
Use :c:func:`PyErr_Occurred` to disambiguate.

.. versionchanged:: 3.1
   A negative *pylong* now raises :exc:`OverflowError`, not :exc:`TypeError`.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 262)**

Unknown directive type "c:function".

```
.. c:function:: unsigned long PyLong_AsUnsignedLongMask(PyObject *obj)

Return a C :c:type:`unsigned long` representation of *obj*. If *obj* is not
an instance of :c:type:`PyLongObject`, first call its :meth:`__index__`
method (if present) to convert it to a :c:type:`PyLongObject`.

If the value of *obj* is out of range for an :c:type:`unsigned long`,
return the reduction of that value modulo ``ULONG_MAX + 1``.

Returns ``(unsigned long)-1`` on error. Use :c:func:`PyErr_Occurred` to
disambiguate.

.. versionchanged:: 3.8
   Use :meth:`__index__` if available.

.. versionchanged:: 3.10
   This function will no longer use :meth:`__int__`.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api] long.rst, line 281)**

Unknown directive type "c:function".

```
.. c:function:: unsigned long long PyLong_AsUnsignedLongLongMask(PyObject *obj)

Return a C :c:type:`unsigned long long` representation of *obj*. If *obj*
is not an instance of :c:type:`PyLongObject`, first call its
:meth:`__index__` method (if present) to convert it to a
:c:type:`PyLongObject`.

If the value of *obj* is out of range for an :c:type:`unsigned long long`,
return the reduction of that value modulo ``ULLONG_MAX + 1``.

Returns ``(unsigned long long)-1`` on error. Use :c:func:`PyErr_Occurred`
to disambiguate.

.. versionchanged:: 3.8
   Use :meth:`__index__` if available.

.. versionchanged:: 3.10
   This function will no longer use :meth:`__int__`.
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 301)**

Unknown directive type "c:function".

```
.. c:function:: double PyLong_AsDouble(PyObject *pylong)
```

Return a C `:c:type: `double`` representation of `*pylong*`. `*pylong*` must be an instance of `:c:type: `PyLongObject``.

Raise `:exc: `OverflowError`` if the value of `*pylong*` is out of range for a `:c:type: `double``.

Returns ```-1.0``` on error. Use `:c:func: `PyErr_Occurred`` to disambiguate.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] long.rst, line 312)**

Unknown directive type "c:function".

```
.. c:function:: void* PyLong_AsVoidPtr(PyObject *pylong)
```

Convert a Python integer `*pylong*` to a C `:c:type: `void`` pointer. If `*pylong*` cannot be converted, an `:exc: `OverflowError`` will be raised. This is only assured to produce a usable `:c:type: `void`` pointer for values created with `:c:func: `PyLong_FromVoidPtr``.

Returns ```NULL``` on error. Use `:c:func: `PyErr_Occurred`` to disambiguate.