

Bytes Objects

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

Unknown directive type "highlight".

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

These functions raise `exc:TypeError` when expecting a bytes parameter and called with a non-bytes parameter.

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

Unknown interpreted text role "exc".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 11)

Unknown directive type "index".

```
.. index:: object: bytes
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 14)

Unknown directive type "c:type".

```
.. c:type:: PyBytesObject
```

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

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 19)

Unknown directive type "c:var".

```
.. c:var:: PyTypeObject PyBytes_Type
```

This instance of `:c:type:PyTypeObject` represents the Python bytes type; it is the same object as `:class:bytes` in the Python layer.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 25)

Unknown directive type "c:function".

```
.. c:function:: int PyBytes_Check(PyObject *o)
```

Return true if the object `*o` is a bytes object or an instance of a subtype of the bytes type. This function always succeeds.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 31)

Unknown directive type "c:function".

```
.. c:function:: int PyBytes_CheckExact(PyObject *o)
```

Return true if the object `*o` is a bytes object, but not an instance of a subtype of the bytes type. This function always succeeds.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 37)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyBytes_FromString(const char *v)
```

Return a new bytes object with a copy of the string **v** as value on success, and ``NULL`` on failure. The parameter **v** must not be ``NULL``; it will not be checked.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 44)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyBytes_FromStringAndSize(const char *v, Py_ssize_t len)
```

Return a new bytes object with a copy of the string **v** as value and length **len** on success, and ``NULL`` on failure. If **v** is ``NULL``, the contents of the bytes object are uninitialized.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 51)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyBytes_FromFormat(const char *format, ...)
```

Take a C `:c:func:`printf`` -style **format** string and a variable number of arguments, calculate the size of the resulting Python bytes object and return a bytes object with the values formatted into it. The variable arguments must be C types and must correspond exactly to the format characters in the **format** string. The following format characters are allowed:

```
.. % XXX: This should be exactly the same as the table in PyErr_Format.
.. % One should just refer to the other.
.. % XXX: The descriptions for %zd and %zu are wrong, but the truth is complicated
.. % because not all compilers support the %z width modifier -- we fake it
.. % when necessary via interpolating PY_FORMAT_SIZE_T.
```

```
.. tabularcolumns:: |l|l|L|
```

Format Characters	Type	Comment
:attr:`%%`	*n/a*	The literal % character.
:attr:`%c`	int	A single byte, represented as a C int.
:attr:`%d`	int	Equivalent to ``printf("%d")``. [1]_
:attr:`%u`	unsigned int	Equivalent to ``printf("%u")``. [1]_
:attr:`%ld`	long	Equivalent to ``printf("%ld")``. [1]_
:attr:`%lu`	unsigned long	Equivalent to ``printf("%lu")``. [1]_
:attr:`%zd`	Py_ssize_t	Equivalent to ``printf("%zd")``. [1]_
:attr:`%zu`	size_t	Equivalent to ``printf("%zu")``. [1]_
:attr:`%i`	int	Equivalent to ``printf("%i")``. [1]_
:attr:`%x`	int	Equivalent to

		``printf("%x")``. [1]_	
attr:``%s``	const char*	A null-terminated C character array.	
attr:``%p``	const void*	The hex representation of a C pointer. Mostly equivalent to ``printf("%p")`` except that it is guaranteed to start with the literal ``0x`` regardless of what the platform's ``printf`` yields.	

An unrecognized format character causes all the rest of the format string to be copied as-is to the result object, and any extra arguments discarded.

.. [1] For integer specifiers (d, u, ld, lu, zd, zu, i, x): the 0-conversion flag has effect even when a precision is given.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 118)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyBytes_FromFormatV(const char *format, va_list args)
```

Identical to :c:func:`PyBytes_FromFormat` except that it takes exactly two arguments.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 124)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyBytes_FromObject(PyObject *o)
```

Return the bytes representation of object *o* that implements the buffer protocol.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 130)

Unknown directive type "c:function".

```
.. c:function:: Py_ssize_t PyBytes_Size(PyObject *o)
```

Return the length of the bytes in bytes object *o*.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 135)

Unknown directive type "c:function".

```
.. c:function:: Py_ssize_t PyBytes_GET_SIZE(PyObject *o)
```

Macro form of :c:func:`PyBytes_Size` but without error checking.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 140)

Unknown directive type "c:function".

```
.. c:function:: char* PyBytes_AsString(PyObject *o)
```

Return a pointer to the contents of *o*. The pointer

refers to the internal buffer of `*o*`, which consists of `len(o) + 1` bytes. The last byte in the buffer is always null, regardless of whether there are any other null bytes. The data must not be modified in any way, unless the object was just created using `PyBytes_FromStringAndSize(NULL, size)`. It must not be deallocated. If `*o*` is not a bytes object at all, `:c:func:PyBytes_AsString` returns `NULL` and raises `:exc:TypeError`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api]bytes.rst, line 152)

Unknown directive type "c:function".

```
.. c:function:: char* PyBytes_AS_STRING(PyObject *string)

    Macro form of :c:func:PyBytes_AsString but without error checking.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api]bytes.rst, line 157)

Unknown directive type "c:function".

```
.. c:function:: int PyBytes_AsStringAndSize(PyObject *obj, char **buffer, Py_ssize_t *length)

    Return the null-terminated contents of the object *obj*
    through the output variables *buffer* and *length*.

    If *length* is NULL, the bytes object
    may not contain embedded null bytes;
    if it does, the function returns -1 and a :exc:ValueError is raised.

    The buffer refers to an internal buffer of *obj*, which includes an
    additional null byte at the end (not counted in *length*). The data
    must not be modified in any way, unless the object was just created using
    PyBytes_FromStringAndSize(NULL, size). It must not be deallocated. If
    *obj* is not a bytes object at all, :c:func:PyBytes_AsStringAndSize
    returns -1 and raises :exc:TypeError.

    .. versionchanged:: 3.5
       Previously, :exc:TypeError was raised when embedded null bytes were
       encountered in the bytes object.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api]bytes.rst, line 178)

Unknown directive type "c:function".

```
.. c:function:: void PyBytes_Concat(PyObject **bytes, PyObject *newpart)

    Create a new bytes object in **bytes containing the contents of *newpart
    appended to *bytes; the caller will own the new reference. The reference to
    the old value of *bytes will be stolen. If the new object cannot be
    created, the old reference to *bytes will still be discarded and the value
    of *bytes will be set to NULL; the appropriate exception will be set.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main [Doc] [c-api]bytes.rst, line 187)

Unknown directive type "c:function".

```
.. c:function:: void PyBytes_ConcatAndDel(PyObject **bytes, PyObject *newpart)

    Create a new bytes object in **bytes containing the contents of *newpart
    appended to *bytes. This version decrements the reference count of
    *newpart.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api]bytes.rst, line 194)

Unknown directive type "c:function".

```
.. c:function:: int _PyBytes_Resize(PyObject **bytes, Py_ssize_t newsize)
```

A way to resize a bytes object even though it is "immutable". Only use this to build up a brand new bytes object; don't use this if the bytes may already be known in other parts of the code. It is an error to call this function if the refcount on the input bytes object is not one. Pass the address of an existing bytes object as an lvalue (it may be written into), and the new size desired. On success, `*bytes` holds the resized bytes object and `0` is returned; the address in `*bytes` may differ from its input value. If the reallocation fails, the original bytes object at `*bytes` is deallocated, `*bytes` is set to `NULL`, `:exc:MemoryError` is set, and `-1` is returned.