

Byte Array Objects

PyByteArrayObject

This subtype of [PyObject](#) represents a Python bytearray object.

PyTypeObject PyByteArray_Type

This instance of [PyTypeObject](#) represents the Python bytearray type; it is the same object as [bytearray](#) in the Python layer.

Type check macros

int PyByteArray_Check([PyObject](#) *o)

Return true if the object *o* is a bytearray object or an instance of a subtype of the bytearray type.

int PyByteArray_CheckExact([PyObject](#) *o)

Return true if the object *o* is a bytearray object, but not an instance of a subtype of the bytearray type.

Direct API functions

[PyObject](#)* PyByteArray_FromObject([PyObject](#) *o)

Return a new bytearray object from any object, *o*, that implements the [buffer protocol](#).

[PyObject](#)* PyByteArray_FromStringAndSize(const char *string, Py_ssize_t len)

Create a new bytearray object from *string* and its length, *len*. On failure, *NULL* is returned.

[PyObject](#)* PyByteArray_Concat([PyObject](#) *a, [PyObject](#) *b)

Concat bytearrays *a* and *b* and return a new bytearray with the result.

Py_ssize_t PyByteArray_Size([PyObject](#) *bytearray)

Return the size of *bytearray* after checking for a *NULL* pointer.

char* PyByteArray_AsString([PyObject](#) *bytearray)

Return the contents of *bytearray* as a char array after checking for a *NULL* pointer. The returned array always has an extra null byte appended.

int PyByteArray_Resize([PyObject](#) *bytearray, Py_ssize_t len)

Resize the internal buffer of *bytearray* to *len*.

Macros

These macros trade safety for speed and they don't check pointers.

`char* PyByteArray_AS_STRING(PyObject *btearray)`

Macro version of `PyByteArray_AsString()`.

`Py_ssize_t PyByteArray_GET_SIZE(PyObject *btearray)`

Macro version of `PyByteArray_Size()`.