Parsing arguments and building values

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

Unknown directive type "highlight".

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

These functions are useful when creating your own extensions functions and methods. Additional information and examples are available in :ref:`extending-index`.

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

Unknown interpreted text role "ref".

The first three of these functions described, :c:func:`PyArg_ParseTuple`, :c:func:`PyArg_ParseTupleAndKeywords`, and :c:func:`PyArg_Parse`, all use *format strings* which are used to tell the function about the expected arguments. The format strings use the same syntax for each of these functions.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 12); *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)arg.rst, line 12); *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)arg.rst, line 12); *backlink*

Unknown interpreted text role "c:func".

Parsing arguments

A format string consists of zero or more "format units." A format unit describes one Python object; it is usually a single character or a parenthesized sequence of format units. With a few exceptions, a format unit that is not a parenthesized sequence normally corresponds to a single address argument to these functions. In the following description, the quoted form is the format unit; the entry in (round) parentheses is the Python object type that matches the format unit; and the entry in [square] brackets is the type of the C variable(s) whose address should be passed.

Strings and buffers

These formats allow accessing an object as a contiguous chunk of memory. You don't have to provide raw storage for the returned unicode or bytes area.

In general, when a format sets a pointer to a buffer, the buffer is managed by the corresponding Python object, and the buffer shares the lifetime of this object. You won't have to release any memory yourself. The only exceptions are `es`, `es#`, `et` and `et#`.

However, when a :c:type:`Py_buffer` structure gets filled, the underlying buffer is locked so that the caller can subsequently use the buffer even inside a :c:type:`Py_BEGIN_ALLOW_THREADS` block without the risk of mutable data being resized or destroyed. As a result, **you have to call** :c:func:`PyBuffer_Release` after you have finished processing the data (or in any early abort case).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 42); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 42); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 42); *backlink*

Unknown interpreted text role "c:func".

Unless otherwise stated, buffers are not NUL-terminated.

Some formats require a read-only :term:`bytes-like object`, and set a pointer instead of a buffer structure. They work by checking that the object's :c:member:`PyBufferProcs.bf_releasebuffer` field is `NULL`, which disallows mutable objects such as :class:`bytearray`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 51); *backlink*

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 51); *backlink*

Unknown interpreted text role "c:member".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 51); *backlink*

Unknown interpreted text role "class".

Note

For all `#` variants of formats (`s#`, `y#`, etc.), the macro :c:macro:`PY_SSIZE_T_CLEAN` must be defined before including :file:`Python.h`. On Python 3.9 and older, the type of the length argument is :c:type:`Py_ssize_t` if the :c:macro:`PY_SSIZE_T_CLEAN` macro is defined, or int otherwise.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 58); *backlink*

Unknown interpreted text role "c:macro".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 58); *backlink*

Unknown interpreted text role "file".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 58); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 58); *backlink*

Unknown interpreted text role "c:macro".

`s` (:class:`str`) [const char *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 83); *backlink*

Unknown interpreted text role "class".

Convert a Unicode object to a C pointer to a character string. A pointer to an existing string is stored in the character pointer variable whose address you pass. The C string is NUL-terminated. The Python string must not contain embedded null code points; if it does, a :exc:`ValueError` exception is raised. Unicode objects are converted to C strings using `'utf-8'` encoding. If this conversion fails, a :exc:`UnicodeError` is raised.

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

Unknown interpreted text role "exc".

Note

This format does not accept :term:`bytes-like objects <bytes-like object>`. If you want to accept filesystem paths and convert them to C character strings, it is preferable to use the `O&` format with :c:func:`PyUnicode_FSConverter` as *converter*.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 75); *backlink*

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 75); *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)arg.rst, line 81)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.5
```

Previously, :exc:`TypeError` was raised when embedded null code points were encountered in the Python string.

`s*` (:class:`str` or :term:`bytes-like object`) [Py_buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 89); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 89); backlink

Unknown interpreted text role "term".

This format accepts Unicode objects as well as bytes-like objects. It fills a :c:type:`Py_buffer` structure provided by the caller. In this case the resulting C string may contain embedded NUL bytes. Unicode objects are converted to C strings using `'utf-8'` encoding.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 86); backlink

Unknown interpreted text role "c:type".

`s#` (:class:`str`, read-only :term:`bytes-like object`) [const char *, :c:type:`Py_ssize_t`]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 96); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 96); backlink

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 96); backlink

Unknown interpreted text role "c:type".

Like `s*`, except that it doesn't accept mutable objects. The result is stored into two C variables, the first one a pointer to a C string, the second one its length. The string may contain embedded null bytes. Unicode objects are converted to C strings using `'utf-8'` encoding.

`z` (:class:`str` or `None`) [const char *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 100); backlink

Unknown interpreted text role "class".

Like `s`, but the Python object may also be `None`, in which case the C pointer is set to `NULL`.

`z*` (:class:`str`, :term:`bytes-like object` or `None`) [Py_buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 104); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 104); backlink

Unknown interpreted text role "term".

Like `s*`, but the Python object may also be `None`, in which case the `buf` member of the :c:type:`Py_buffer` structure is set to `NULL`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 103); backlink

Unknown interpreted text role "c:type".

`z#` (:class:`str`, read-only :term:`bytes-like object` or `None`) [const char *, :c:type:`Py_ssize_t`]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 108); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 108); *backlink*

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 108); *backlink*

Unknown interpreted text role "c:type".

Like `s#`, but the Python object may also be `None`, in which case the C pointer is set to `NULL`.

`y` (read-only :term:`bytes-like object`) [const char *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 118); *backlink*

Unknown interpreted text role "term".

This format converts a bytes-like object to a C pointer to a character string; it does not accept Unicode objects. The bytes buffer must not contain embedded null bytes; if it does, a :exc:`ValueError` exception is raised.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 111); *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)arg.rst, line 116)

Unknown directive type "versionchanged".

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

`y*` (:term:`bytes-like object`) [Py_buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 123); *backlink*

Unknown interpreted text role "term".

This variant on `s*` doesn't accept Unicode objects, only bytes-like objects. **This is the recommended way to accept binary data.**

`y#` (read-only :term:`bytes-like object`) [const char *, :c:type:`Py_ssize_t`]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 127); *backlink*

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 127); *backlink*

Unknown interpreted text role "c:type".

This variant on `s#` doesn't accept Unicode objects, only bytes-like objects.

`S` (:class:`bytes`) [PyBytesObject *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 132); *backlink*

Unknown interpreted text role "class".

Requires that the Python object is a :class:`bytes` object, without attempting any conversion. Raises :exc:`TypeError` if the object is not a bytes object. The C variable may also be declared as :c:type:`PyObject*`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 130); *backlink*

Unknown interpreted text role "class".

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

Unknown interpreted text role "c:type".

`Y` (:class:`bytearray`) [PyByteArrayObject *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 137); backlink

Unknown interpreted text role "class".

Requires that the Python object is a :class:`bytearray` object, without attempting any conversion. Raises :exc:`TypeError` if the object is not a :class:`bytearray` object. The C variable may also be declared as :c:type:`PyObject*`.

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

Unknown interpreted text role "class".

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

Unknown interpreted text role "class".

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

Unknown interpreted text role "c:type".

`u` (:class:`str`) [const Py_UNICODE *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 154); backlink

Unknown interpreted text role "class".

Convert a Python Unicode object to a C pointer to a NUL-terminated buffer of Unicode characters. You must pass the address of a :c:type:`Py_UNICODE` pointer variable, which will be filled with the pointer to an existing Unicode buffer. Please note that the width of a :c:type:`Py_UNICODE` character depends on compilation options (it is either 16 or 32 bits). The Python string must not contain embedded null code points; if it does, a :exc:`ValueError` exception is raised.

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

Unknown interpreted text role "c:type".

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

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 140); 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)arg.rst, line 148)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.5
```

```
Previously, :exc:`TypeError` was raised when embedded null code points
were encountered in the Python string.
```

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

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.3 3.12
   Part of the old-style :c:type:`Py_UNICODE` API; please migrate to using
   :c:func:`PyUnicode_AsWideCharString`.
```

`u#` (:class:`str`) [const Py_UNICODE *, :c:type:`Py_ssize_t`]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 163); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 163); *backlink*

Unknown interpreted text role "c:type".

This variant on `u` stores into two C variables, the first one a pointer to a Unicode data buffer, the second one its length. This variant allows null code points.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 161)

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.3 3.12
   Part of the old-style :c:type:`Py_UNICODE` API; please migrate to using
   :c:func:`PyUnicode_AsWideCharString`.
```

`Z` (:class:`str` or `None`) [const Py_UNICODE *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 171); *backlink*

Unknown interpreted text role "class".

Like `u`, but the Python object may also be `None`, in which case the :c:type:`Py_UNICODE` pointer is set to `NULL`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 166); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 169)

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.3 3.12
   Part of the old-style :c:type:`Py_UNICODE` API; please migrate to using
   :c:func:`PyUnicode_AsWideCharString`.
```

`Z#` (:class:`str` or `None`) [const Py_UNICODE *, :c:type:`Py_ssize_t`]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 179); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 179); *backlink*

Unknown interpreted text role "c:type".

Like `u#`, but the Python object may also be `None`, in which case the :c:type:`Py_UNICODE` pointer is set to `NULL`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 174); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 177)

Unknown directive type "deprecated-removed".

```
.. deprecated-removed:: 3.3 3.12
   Part of the old-style :c:type:`Py_UNICODE` API; please migrate to using
   :c:func:`PyUnicode_AsWideCharString`.
```

`U` (:class:`str`) [PyObject *]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 184); *backlink*

Unknown interpreted text role "class".

Requires that the Python object is a Unicode object, without attempting any conversion. Raises :exc:`TypeError` if the object is not a Unicode object. The C variable may also be declared as :c:type:`PyObject*`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 182); *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)arg.rst, line 182); *backlink*

Unknown interpreted text role "c:type".

`w*` (read-write :term:`bytes-like object`) [Py_buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 190); *backlink*

Unknown interpreted text role "term".

This format accepts any object which implements the read-write buffer interface. It fills a :c:type:`Py_buffer` structure provided by the caller. The buffer may contain embedded null bytes. The caller have to call :c:func:`PyBuffer_Release` when it is done with the buffer.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 187); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 187); *backlink*

Unknown interpreted text role "c:func".

`es` (:class:`str`) [const char *encoding, char **buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 207); *backlink*

Unknown interpreted text role "class".

This variant on `s` is used for encoding Unicode into a character buffer. It only works for encoded data without embedded NUL bytes.

This format requires two arguments. The first is only used as input, and must be a :c:type:`const char*` which points to the name of an encoding as a NUL-terminated string, or `NULL`, in which case `'utf-8'` encoding is used. An exception is raised if the named encoding is not known to Python. The second argument must be a :c:type:`char**`; the value of the pointer it references will be set to a buffer with the contents of the argument text. The text will be encoded in the encoding specified by the first argument.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 196); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 196); *backlink*

Unknown interpreted text role "c:type".

:c:func:`PyArg_ParseTuple` will allocate a buffer of the needed size, copy the encoded data into this buffer and adjust **buffer* to reference the newly allocated storage. The caller is responsible for calling :c:func:`PyMem_Free` to free the allocated buffer after use.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 204); 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)arg.rst, line 204); backlink

Unknown interpreted text role "c:func".

`et` (:class:`str`, :class:`bytes` or :class:`bytearray`) [const char *encoding, char **buffer]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 212); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 212); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 212); backlink

Unknown interpreted text role "class".

Same as `es` except that byte string objects are passed through without recoding them. Instead, the implementation assumes that the byte string object uses the encoding passed in as parameter.

`es#` (:class:`str`) [const char *encoding, char **buffer, :c:type:`Py_ssize_t` *buffer_length]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 243); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 243); backlink

Unknown interpreted text role "c:type".

This variant on `s#` is used for encoding Unicode into a character buffer. Unlike the `es` format, this variant allows input data which contains NUL characters.

It requires three arguments. The first is only used as input, and must be a :c:type:`const char*` which points to the name of an encoding as a NUL-terminated string, or `NULL`, in which case `'utf-8'` encoding is used. An exception is raised if the named encoding is not known to Python. The second argument must be a :c:type:`char**`; the value of the pointer it references will be set to a buffer with the contents of the argument text. The text will be encoded in the encoding specified by the first argument. The third argument must be a pointer to an integer; the referenced integer will be set to the number of bytes in the output buffer.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 219); backlink

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 219); backlink

Unknown interpreted text role "c:type".

There are two modes of operation:

If **buffer* points a `NULL` pointer, the function will allocate a buffer of the needed size, copy the encoded data into this buffer and set **buffer* to reference the newly allocated storage. The caller is responsible for calling :c:func:`PyMem_Free` to free the allocated buffer after usage.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 231); backlink

Unknown interpreted text role "c:func".

If **buffer* points to a non-`NULL` pointer (an already allocated buffer), :c:func:`PyArg_ParseTuple` will use this location as the buffer and interpret the initial value of **buffer_length* as the buffer size. It will then copy the encoded data into the buffer and NUL-terminate it. If the buffer is not large enough, a :exc:`ValueError` will be set.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 236); 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)arg.rst, line 236); backlink

Unknown interpreted text role "exc".

In both cases, **buffer_length* is set to the length of the encoded data without the trailing NUL byte.

`et#` (:class:`str`, :class:`bytes` or :class:`bytearray`) [const char *encoding, char **buffer, :c:type:`Py_ssize_t` *buffer_length]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 248); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 248); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 248); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 248); backlink

Unknown interpreted text role "c:type".

Same as `es#` except that byte string objects are passed through without recoding them. Instead, the implementation assumes that the byte string object uses the encoding passed in as parameter.

Numbers

`b` (:class:`int`) [unsigned char]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 255); backlink

Unknown interpreted text role "class".

Convert a nonnegative Python integer to an unsigned tiny int, stored in a C :c:type:`unsigned char`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 254); backlink

Unknown interpreted text role "c:type".

`B` (:class:`int`) [unsigned char]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 259); backlink

Unknown interpreted text role "class".

Convert a Python integer to a tiny int without overflow checking, stored in a C :c:type:`unsigned char`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 258); backlink

Unknown interpreted text role "c:type".

`h` (:class:`int`) [short int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 262); backlink

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`short int`.

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

Unknown interpreted text role "c:type".

`H` (:class:`int`) [unsigned short int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 266); *backlink*

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`unsigned short int`, without overflow checking.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 265); *backlink*

Unknown interpreted text role "c:type".

`i` (:class:`int`) [int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 269); *backlink*

Unknown interpreted text role "class".

Convert a Python integer to a plain C :c:type:`int`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 269); *backlink*

Unknown interpreted text role "c:type".

`I` (:class:`int`) [unsigned int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 273); *backlink*

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`unsigned int`, without overflow checking.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 272); *backlink*

Unknown interpreted text role "c:type".

`l` (:class:`int`) [long int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 276); *backlink*

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`long int`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 276); *backlink*

Unknown interpreted text role "c:type".

`k` (:class:`int`) [unsigned long]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 280); *backlink*

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`unsigned long` without overflow checking.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 279); *backlink*

Unknown interpreted text role "c:type".

`L` (:class:`int`) [long long]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 283); backlink

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`long long`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 283); backlink

Unknown interpreted text role "c:type".

`K` (:class:`int`) [unsigned long long]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 287); backlink

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`unsigned long long` without overflow checking.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 286); backlink

Unknown interpreted text role "c:type".

`n` (:class:`int`) [Py_ssize_t]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 290); backlink

Unknown interpreted text role "class".

Convert a Python integer to a C :c:type:`Py_ssize_t`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 290); backlink

Unknown interpreted text role "c:type".

`c` (:class:`bytes` or :class:`bytearray` of length 1) [char]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 297); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 297); backlink

Unknown interpreted text role "class".

Convert a Python byte, represented as a :class:`bytes` or :class:`bytearray` object of length 1, to a C :c:type:`char`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 293); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 293); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 293); backlink

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 296)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.3
   Allow :class:`bytearray` objects.
```

`c` (:class:`str` of length 1) [int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 301); backlink

Unknown interpreted text role "class".

Convert a Python character, represented as a :class:`str` object of length 1, to a C :c:type:`int`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 300); backlink

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 300); backlink

Unknown interpreted text role "c:type".

`f` (:class:`float`) [float]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 304); backlink

Unknown interpreted text role "class".

Convert a Python floating point number to a C :c:type:`float`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 304); backlink

Unknown interpreted text role "c:type".

`d` (:class:`float`) [double]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 307); backlink

Unknown interpreted text role "class".

Convert a Python floating point number to a C :c:type:`double`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 307); backlink

Unknown interpreted text role "c:type".

`D` (:class:`complex`) [Py_complex]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 310); backlink

Unknown interpreted text role "class".

Convert a Python complex number to a C :c:type:`Py_complex` structure.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 310); backlink

Unknown interpreted text role "c:type".

Other objects

`O` (object) [PyObject *]

Store a Python object (without any conversion) in a C object pointer. The C program thus receives the actual object that was passed. The object's reference count is not increased. The pointer stored is not `NULL`.

`O!` (object) [*typeobject*, PyObject *]

Store a Python object in a C object pointer. This is similar to `O`, but takes two C arguments: the first is the address of a Python type object, the second is the address of the C variable (of type :c:type:`PyObject*`) into which the object pointer is stored. If the Python object does not have the required type, :exc:`TypeError` is raised.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 321); backlink

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 321); backlink

Unknown interpreted text role "exc".

`O&` (object) [*converter*, *anything*]

Convert a Python object to a C variable through a *converter* function. This takes two arguments: the first is a function, the second is the address of a C variable (of arbitrary type), converted to :c:type:`void *`. The *converter* function in turn is called as follows:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 330); backlink

Unknown interpreted text role "c:type".

```
status = converter(object, address);
```

where *object* is the Python object to be converted and *address* is the :c:type:`void*` argument that was passed to the :c:func:`PyArg_Parse\*` function. The returned *status* should be `1` for a successful conversion and `0` if the conversion has failed. When the conversion fails, the *converter* function should raise an exception and leave the content of *address* unmodified.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 337); backlink

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 337); backlink

Unknown interpreted text role "c:func".

If the *converter* returns `Py_CLEANUP_SUPPORTED`, it may get called a second time if the argument parsing eventually fails, giving the converter a chance to release any memory that it had already allocated. In this second call, the *object* parameter will be `NULL`; *address* will have the same value as in the original call.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 349)

Unknown directive type "versionchanged".

```
.. versionchanged:: 3.1
   ``Py_CLEANUP_SUPPORTED`` was added.
```

`p` (:class:`bool`) [int]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 359); backlink

Unknown interpreted text role "class".

Tests the value passed in for truth (a boolean **p**redicate) and converts the result to its equivalent C true/false integer value. Sets the int to `1` if the expression was true and `0` if it was false. This accepts any valid Python value. See :ref:`truth` for more information about how Python tests values for truth.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 353); backlink

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 359)

Unknown directive type "versionadded".

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

`(items)` (:class:`tuple`) [*matching-items*]

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 364); backlink

Unknown interpreted text role "class".

The object must be a Python sequence whose length is the number of format units in *items*. The C arguments must correspond to the individual format units in *items*. Format units for sequences may be nested.

It is possible to pass "long" integers (integers whose value exceeds the platform's :const:`LONG_MAX`) however no proper range checking is done --- the most significant bits are silently truncated when the receiving field is too small to receive the value (actually, the semantics are inherited from downcasts in C --- your mileage may vary).

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 366); *backlink***
>
> Unknown interpreted text role "const".

A few other characters have a meaning in a format string. These may not occur inside nested parentheses. They are:

`|`

Indicates that the remaining arguments in the Python argument list are optional. The C variables corresponding to optional arguments should be initialized to their default value --- when an optional argument is not specified, :c:func:`PyArg_ParseTuple` does not touch the contents of the corresponding C variable(s).

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 376); *backlink***
>
> Unknown interpreted text role "c:func".

`$`

:c:func:`PyArg_ParseTupleAndKeywords` only: Indicates that the remaining arguments in the Python argument list are keyword-only. Currently, all keyword-only arguments must also be optional arguments, so `|` must always be specified before `$` in the format string.

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 383); *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)arg.rst, line 389)**
>
> Unknown directive type "versionadded".
>
> ```
> .. versionadded:: 3.3
> ```

`:`

The list of format units ends here; the string after the colon is used as the function name in error messages (the "associated value" of the exception that :c:func:`PyArg_ParseTuple` raises).

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 392); *backlink***
>
> Unknown interpreted text role "c:func".

`;`

The list of format units ends here; the string after the semicolon is used as the error message *instead* of the default error message. `:` and `;` mutually exclude each other.

Note that any Python object references which are provided to the caller are *borrowed* references; do not decrement their reference count!

Additional arguments passed to these functions must be addresses of variables whose type is determined by the format string; these are used to store values from the input tuple. There are a few cases, as described in the list of format units above, where these parameters are used as input values; they should match what is specified for the corresponding format unit in that case.

For the conversion to succeed, the *arg* object must match the format and the format must be exhausted. On success, the :c:func:`PyArg_Parse\*` functions return true, otherwise they return false and raise an appropriate exception. When the :c:func:`PyArg_Parse\*` functions fail due to conversion failure in one of the format units, the variables at the addresses corresponding to that and the following format units are left untouched.

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 410); *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)arg.rst, line 410); *backlink***
>
> Unknown interpreted text role "c:func".

API Functions

> **System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 421)**
>
> Unknown directive type "c:function".
>
> ```
> .. c:function:: int PyArg_ParseTuple(PyObject *args, const char *format, ...)
>
>    Parse the parameters of a function that takes only positional parameters into
>    local variables.  Returns true on success; on failure, it returns false and
>    raises the appropriate exception.
> ```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 428)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_VaParse(PyObject *args, const char *format, va_list vargs)

   Identical to :c:func:`PyArg_ParseTuple`, except that it accepts a va_list rather
   than a variable number of arguments.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 434)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_ParseTupleAndKeywords(PyObject *args, PyObject *kw, const char *format, char *keywords[],

   Parse the parameters of a function that takes both positional and keyword
   parameters into local variables.  The *keywords* argument is a
   ``NULL``-terminated array of keyword parameter names.  Empty names denote
   :ref:`positional-only parameters <positional-only_parameter>`.
   Returns true on success; on failure, it returns false and raises the
   appropriate exception.

   .. versionchanged:: 3.6
      Added support for :ref:`positional-only parameters
      <positional-only_parameter>`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 448)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_VaParseTupleAndKeywords(PyObject *args, PyObject *kw, const char *format, char *keywords

   Identical to :c:func:`PyArg_ParseTupleAndKeywords`, except that it accepts a
   va_list rather than a variable number of arguments.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 454)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_ValidateKeywordArguments(PyObject *)

   Ensure that the keys in the keywords argument dictionary are strings.  This
   is only needed if :c:func:`PyArg_ParseTupleAndKeywords` is not used, since the
   latter already does this check.

   .. versionadded:: 3.2
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 464)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_Parse(PyObject *args, const char *format, ...)

   Function used to deconstruct the argument lists of "old-style" functions ---
   these are functions which use the :const:`METH_OLDARGS` parameter parsing
   method, which has been removed in Python 3.  This is not recommended for use
   in parameter parsing in new code, and most code in the standard interpreter
   has been modified to no longer use this for that purpose.  It does remain a
   convenient way to decompose other tuples, however, and may continue to be
   used for that purpose.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 475)

Unknown directive type "c:function".

```
.. c:function:: int PyArg_UnpackTuple(PyObject *args, const char *name, Py_ssize_t min, Py_ssize_t max, ...)

   A simpler form of parameter retrieval which does not use a format string to
   specify the types of the arguments.  Functions which use this method to retrieve
   their parameters should be declared as :const:`METH_VARARGS` in function or
   method tables.  The tuple containing the actual parameters should be passed as
   *args*; it must actually be a tuple.  The length of the tuple must be at least
   *min* and no more than *max*; *min* and *max* may be equal.  Additional
   arguments must be passed to the function, each of which should be a pointer to a
   :c:type:`PyObject*` variable; these will be filled in with the values from
   *args*; they will contain :term:`borrowed references <borrowed reference>`.
   The variables which correspond
   to optional parameters not given by *args* will not be filled in; these should
   be initialized by the caller. This function returns true on success and false if
   *args* is not a tuple or contains the wrong number of elements; an exception
   will be set if there was a failure.
```

```
This is an example of the use of this function, taken from the sources for the
:mod:`_weakref` helper module for weak references::

   static PyObject *
   weakref_ref(PyObject *self, PyObject *args)
   {
       PyObject *object;
       PyObject *callback = NULL;
       PyObject *result = NULL;

       if (PyArg_UnpackTuple(args, "ref", 1, 2, &object, &callback)) {
           result = PyWeakref_NewRef(object, callback);
       }
       return result;
   }

The call to :c:func:`PyArg_UnpackTuple` in this example is entirely equivalent to
this call to :c:func:`PyArg_ParseTuple`::

   PyArg_ParseTuple(args, "O|O:ref", &object, &callback)
```

Building values

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst, line 518)

Unknown directive type "c:function".

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

   Create a new value based on a format string similar to those accepted by the
   :c:func:`PyArg_Parse\*` family of functions and a sequence of values.  Returns
   the value or ``NULL`` in the case of an error; an exception will be raised if
   ``NULL`` is returned.

   :c:func:`Py_BuildValue` does not always build a tuple.  It builds a tuple only if
   its format string contains two or more format units.  If the format string is
   empty, it returns ``None``; if it contains exactly one format unit, it returns
   whatever object is described by that format unit.  To force it to return a tuple
   of size 0 or one, parenthesize the format string.

   When memory buffers are passed as parameters to supply data to build objects, as
   for the ``s`` and ``s#`` formats, the required data is copied.  Buffers provided
   by the caller are never referenced by the objects created by
   :c:func:`Py_BuildValue`.  In other words, if your code invokes :c:func:`malloc`
   and passes the allocated memory to :c:func:`Py_BuildValue`, your code is
   responsible for calling :c:func:`free` for that memory once
   :c:func:`Py_BuildValue` returns.

   In the following description, the quoted form is the format unit; the entry in
   (round) parentheses is the Python object type that the format unit will return;
   and the entry in [square] brackets is the type of the C value(s) to be passed.

   The characters space, tab, colon and comma are ignored in format strings (but
   not within format units such as ``s#``).  This can be used to make long format
   strings a tad more readable.

   ``s`` (:class:`str` or ``None``) [const char \*]
      Convert a null-terminated C string to a Python :class:`str` object using ``'utf-8'``
      encoding. If the C string pointer is ``NULL``, ``None`` is used.

   ``s#`` (:class:`str` or ``None``) [const char \*, :c:type:`Py_ssize_t`]
      Convert a C string and its length to a Python :class:`str` object using ``'utf-8'``
      encoding. If the C string pointer is ``NULL``, the length is ignored and
      ``None`` is returned.

   ``y`` (:class:`bytes`) [const char \*]
      This converts a C string to a Python :class:`bytes` object.  If the C
      string pointer is ``NULL``, ``None`` is returned.

   ``y#`` (:class:`bytes`) [const char \*, :c:type:`Py_ssize_t`]
      This converts a C string and its lengths to a Python object.  If the C
      string pointer is ``NULL``, ``None`` is returned.

   ``z`` (:class:`str` or ``None``) [const char \*]
      Same as ``s``.

   ``z#`` (:class:`str` or ``None``) [const char \*, :c:type:`Py_ssize_t`]
      Same as ``s#``.

   ``u`` (:class:`str`) [const wchar_t \*]
      Convert a null-terminated :c:type:`wchar_t` buffer of Unicode (UTF-16 or UCS-4)
      data to a Python Unicode object.  If the Unicode buffer pointer is ``NULL``,
      ``None`` is returned.

   ``u#`` (:class:`str`) [const wchar_t \*, :c:type:`Py_ssize_t`]
      Convert a Unicode (UTF-16 or UCS-4) data buffer and its length to a Python
      Unicode object.   If the Unicode buffer pointer is ``NULL``, the length is ignored
      and ``None`` is returned.

   ``U`` (:class:`str` or ``None``) [const char \*]
      Same as ``s``.

   ``U#`` (:class:`str` or ``None``) [const char \*, :c:type:`Py_ssize_t`]
      Same as ``s#``.

   ``i`` (:class:`int`) [int]
      Convert a plain C :c:type:`int` to a Python integer object.

   ``b`` (:class:`int`) [char]
      Convert a plain C :c:type:`char` to a Python integer object.

   ``h`` (:class:`int`) [short int]
```

```
   Convert a plain C :c:type:`short int` to a Python integer object.

``l`` (:class:`int`) [long int]
   Convert a C :c:type:`long int` to a Python integer object.

``B`` (:class:`int`) [unsigned char]
   Convert a C :c:type:`unsigned char` to a Python integer object.

``H`` (:class:`int`) [unsigned short int]
   Convert a C :c:type:`unsigned short int` to a Python integer object.

``I`` (:class:`int`) [unsigned int]
   Convert a C :c:type:`unsigned int` to a Python integer object.

``k`` (:class:`int`) [unsigned long]
   Convert a C :c:type:`unsigned long` to a Python integer object.

``L`` (:class:`int`) [long long]
   Convert a C :c:type:`long long` to a Python integer object.

``K`` (:class:`int`) [unsigned long long]
   Convert a C :c:type:`unsigned long long` to a Python integer object.

``n`` (:class:`int`) [Py_ssize_t]
   Convert a C :c:type:`Py_ssize_t` to a Python integer.

``c`` (:class:`bytes` of length 1) [char]
   Convert a C :c:type:`int` representing a byte to a Python :class:`bytes` object of
   length 1.

``C`` (:class:`str` of length 1) [int]
   Convert a C :c:type:`int` representing a character to Python :class:`str`
   object of length 1.

``d`` (:class:`float`) [double]
   Convert a C :c:type:`double` to a Python floating point number.

``f`` (:class:`float`) [float]
   Convert a C :c:type:`float` to a Python floating point number.

``D`` (:class:`complex`) [Py_complex \*]
   Convert a C :c:type:`Py_complex` structure to a Python complex number.

``O`` (object) [PyObject \*]
   Pass a Python object untouched (except for its reference count, which is
   incremented by one).  If the object passed in is a ``NULL`` pointer, it is assumed
   that this was caused because the call producing the argument found an error and
   set an exception. Therefore, :c:func:`Py_BuildValue` will return ``NULL`` but won't
   raise an exception.  If no exception has been raised yet, :exc:`SystemError` is
   set.

``S`` (object) [PyObject \*]
   Same as ``O``.

``N`` (object) [PyObject \*]
   Same as ``O``, except it doesn't increment the reference count on the object.
   Useful when the object is created by a call to an object constructor in the
   argument list.

``O&`` (object) [*converter*, *anything*]
   Convert *anything* to a Python object through a *converter* function.  The
   function is called with *anything* (which should be compatible with :c:type:`void*`)
   as its argument and should return a "new" Python object, or ``NULL`` if an
   error occurred.

``(items)`` (:class:`tuple`) [*matching-items*]
   Convert a sequence of C values to a Python tuple with the same number of items.

``[items]`` (:class:`list`) [*matching-items*]
   Convert a sequence of C values to a Python list with the same number of items.

``{items}`` (:class:`dict`) [*matching-items*]
   Convert a sequence of C values to a Python dictionary.  Each pair of consecutive
   C values adds one item to the dictionary, serving as key and value,
   respectively.

If there is an error in the format string, the :exc:`SystemError` exception is
set and ``NULL`` returned.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\(cpython-main)(Doc)(c-api)arg.rst`, line 672)

Unknown directive type "c:function".

```
.. c:function:: PyObject* Py_VaBuildValue(const char *format, va_list vargs)

   Identical to :c:func:`Py_BuildValue`, except that it accepts a va_list
   rather than a variable number of arguments.
```