

Concrete Objects Layer

The functions in this chapter are specific to certain Python object types. Passing them an object of the wrong type is not a good idea; if you receive an object from a Python program and you are not sure that it has the right type, you must perform a type check first; for example, to check that an object is a dictionary, use `PyDict_Check()`. The chapter is structured like the “family tree” of Python object types.

Warning: While the functions described in this chapter carefully check the type of the objects which are passed in, many of them do not check for *NULL* being passed instead of a valid object. Allowing *NULL* to be passed in can cause memory access violations and immediate termination of the interpreter.

Fundamental Objects

This section describes Python type objects and the singleton object `None`.

- [Type Objects](#)
- [The None Object](#)

Numeric Objects

- [Integer Objects](#)
- [Boolean Objects](#)
- [Floating Point Objects](#)
- [Complex Number Objects](#)
 - [Complex Numbers as C Structures](#)
 - [Complex Numbers as Python Objects](#)

Sequence Objects

Generic operations on sequence objects were discussed in the previous chapter; this section deals with the specific kinds of sequence objects that are intrinsic to the Python language.

- [Bytes Objects](#)
- [Byte Array Objects](#)
 - [Type check macros](#)
 - [Direct API functions](#)
 - [Macros](#)
- [Unicode Objects and Codecs](#)

- [Unicode Objects](#)
 - [Unicode Type](#)
 - [Unicode Character Properties](#)
 - [Creating and accessing Unicode strings](#)
 - [Deprecated Py_UNICODE APIs](#)
 - [Locale Encoding](#)
 - [File System Encoding](#)
 - [wchar_t Support](#)
- [Built-in Codecs](#)
 - [Generic Codecs](#)
 - [UTF-8 Codecs](#)
 - [UTF-32 Codecs](#)
 - [UTF-16 Codecs](#)
 - [UTF-7 Codecs](#)
 - [Unicode-Escape Codecs](#)
 - [Raw-Unicode-Escape Codecs](#)
 - [Latin-1 Codecs](#)
 - [ASCII Codecs](#)
 - [Character Map Codecs](#)
 - [MBCS codecs for Windows](#)
 - [Methods & Slots](#)
- [Methods and Slot Functions](#)
- [Tuple Objects](#)
- [Struct Sequence Objects](#)
- [List Objects](#)

Container Objects

- [Dictionary Objects](#)
- [Set Objects](#)

Function Objects

- [Function Objects](#)
- [Instance Method Objects](#)
- [Method Objects](#)
- [Cell Objects](#)
- [Code Objects](#)

Other Objects

- [File Objects](#)
- [Module Objects](#)
 - [Initializing C modules](#)

- Single-phase initialization
- Multi-phase initialization
- Low-level module creation functions
- Support functions
- Module lookup
- Iterator Objects
- Descriptor Objects
- Slice Objects
- Ellipsis Object
- MemoryView objects
- Weak Reference Objects
- Capsules
- Generator Objects
- Coroutine Objects
- DateTime Objects