

Code Objects

Code objects are a low-level detail of the CPython implementation. Each one represents a chunk of executable code that hasn't yet been bound into a function.

PyCodeObject

The C structure of the objects used to describe code objects. The fields of this type are subject to change at any time.

PyTypeObject PyCode_Type

This is an instance of `PyTypeObject` representing the Python `code` type.

`int PyCode_Check(PyObject *co)`

Return true if `co` is a `code` object.

`int PyCode_GetNumFree(PyCodeObject *co)`

Return the number of free variables in `co`.

`PyCodeObject* PyCode_New(int argcount, int kwonlyargcount, int nlocals, int stacksize, int flags, PyObject *code, PyObject *consts, PyObject *names, PyObject *varnames, PyObject *freevars, PyObject *cellvars, PyObject *filename, PyObject *name, int firstlineno, PyObject *notab)`

Return a new code object. If you need a dummy code object to create a frame, use `PyCode_NewEmpty()` instead. Calling `PyCode_New()` directly can bind you to a precise Python version since the definition of the bytecode changes often.

`PyCodeObject* PyCode_NewEmpty(const char *filename, const char *funcname, int firstlineno)`

Return a new empty code object with the specified filename, function name, and first line number. It is illegal to `exec()` or `eval()` the resulting code object.