Floating Point Objects

`PyFloatObject`

This subtype of `PyObject` represents a Python floating point object.

PyTypeObject **`PyFloat_Type`**

This instance of `PyTypeObject` represents the Python floating point type. This is the same object as `float` in the Python layer.

int **`PyFloat_Check`**(PyObject **p*)

Return true if its argument is a `PyFloatObject` or a subtype of `PyFloatObject`.

int **`PyFloat_CheckExact`**(PyObject **p*)

Return true if its argument is a `PyFloatObject`, but not a subtype of `PyFloatObject`.

PyObject* **`PyFloat_FromString`**(PyObject **str*)

Return value: New reference.

Create a `PyFloatObject` object based on the string value in *str*, or *NULL* on failure.

PyObject* **`PyFloat_FromDouble`**(double *v*)

Return value: New reference.

Create a `PyFloatObject` object from *v*, or *NULL* on failure.

double **`PyFloat_AsDouble`**(PyObject **pyfloat*)

Return a C `double` representation of the contents of *pyfloat*. If *pyfloat* is not a Python floating point object but has a `__float__()` method, this method will first be called to convert *pyfloat* into a float. This method returns `-1.0` upon failure, so one should call `PyErr_Occurred()` to check for errors.

double **`PyFloat_AS_DOUBLE`**(PyObject **pyfloat*)

Return a C `double` representation of the contents of *pyfloat*, but without error checking.

PyObject* **`PyFloat_GetInfo`**(void)

Return a structseq instance which contains information about the precision, minimum and maximum values of a float. It's a thin wrapper around the header file `float.h`.

double **`PyFloat_GetMax`**()

Return the maximum representable finite float *DBL_MAX* as C `double`.

double **PyFloat_GetMin**()

Return the minimum normalized positive float *DBL_MIN* as C `double`.

int **PyFloat_ClearFreeList**()

Clear the float free list. Return the number of items that could not be freed.