

# Mapping Protocol

See also [PyObject\\_GetItem\(\)](#), [PyObject\\_SetItem\(\)](#) and [PyObject\\_DelItem\(\)](#).

int **PyMapping\_Check**(PyObject \*o)

Return 1 if the object provides mapping protocol or supports slicing, and 0 otherwise. Note that it returns 1 for Python classes with a [\\_\\_getitem\\_\\_\(\)](#) method since in general case it is impossible to determine what the type of keys it supports. This function always succeeds.

Py\_ssize\_t **PyMapping\_Size**(PyObject \*o)

Py\_ssize\_t **PyMapping\_Length**(PyObject \*o)

Returns the number of keys in object o on success, and -1 on failure. This is equivalent to the Python expression `len(o)`.

PyObject\* **PyMapping\_GetItemString**(PyObject \*o, const char \*key)

*Return value: New reference.*

Return element of o corresponding to the string key or NULL on failure. This is the equivalent of the Python expression `o[key]`. See also [PyObject\\_GetItem\(\)](#).

int **PyMapping\_SetItemString**(PyObject \*o, const char \*key, PyObject \*v)

Map the string key to the value v in object o. Returns -1 on failure. This is the equivalent of the Python statement `o[key] = v`. See also [PyObject\\_SetItem\(\)](#).

int **PyMapping\_DelItem**(PyObject \*o, PyObject \*key)

Remove the mapping for the object key from the object o. Return -1 on failure. This is equivalent to the Python statement `del o[key]`. This is an alias of [PyObject\\_DelItem\(\)](#).

int **PyMapping\_DelItemString**(PyObject \*o, const char \*key)

Remove the mapping for the string key from the object o. Return -1 on failure. This is equivalent to the Python statement `del o[key]`.

int **PyMapping\_HasKey**(PyObject \*o, PyObject \*key)

Return 1 if the mapping object has the key key and 0 otherwise. This is equivalent to the Python expression `key in o`. This function always succeeds.

int **PyMapping\_HasKeyString**(PyObject \*o, const char \*key)

Return 1 if the mapping object has the key key and 0 otherwise. This is equivalent to the Python expression `key in o`. This function always succeeds.

**PyObject\*** **PyMapping\_Keys**(PyObject \*o)

*Return value: New reference.*

On success, return a list or tuple of the keys in object o. On failure, return *NULL*.

**PyObject\*** **PyMapping\_Values**(PyObject \*o)

*Return value: New reference.*

On success, return a list or tuple of the values in object o. On failure, return *NULL*.

**PyObject\*** **PyMapping\_Items**(PyObject \*o)

*Return value: New reference.*

On success, return a list or tuple of the items in object o, where each item is a tuple containing a key-value pair. On failure, return *NULL*.