

Iterator Protocol

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

Unknown directive type "highlight".

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

There are two functions specifically for working with iterators.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main) (Doc) (c-api) iter.rst, line 10)

Unknown directive type "c:function".

```
.. c:function:: int PyIter_Check(PyObject *o)
```

Return non-zero if the object **o* can be safely passed to :c:func:`PyIter_Next`, and ``0`` otherwise. This function always succeeds.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main) (Doc) (c-api) iter.rst, line 15)

Unknown directive type "c:function".

```
.. c:function:: int PyAIter_Check(PyObject *o)
```

Return non-zero if the object **o* provides the :class:`AsyncIterator` protocol, and ``0`` otherwise. This function always succeeds.

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\cpython-main) (Doc) (c-api) iter.rst, line 22)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyIter_Next(PyObject *o)
```

Return the next value from the iterator **o*. The object must be an iterator according to :c:func:`PyIter_Check` (it is up to the caller to check this). If there are no remaining values, returns ``NULL`` with no exception set. If an error occurs while retrieving the item, returns ``NULL`` and passes along the exception.

To write a loop which iterates over an iterator, the C code should look something like this:

```
PyObject *iterator = PyObject_GetIter(obj);
PyObject *item;
```

```
if (iterator == NULL) {
    /* propagate error */
}
```

```
while ((item = PyIter_Next(iterator))) {
    /* do something with item */
    ...
    /* release reference when done */
    Py_DECREF(item);
}
```

```
Py_DECREF(iterator);
```

```
if (PyErr_Occurred()) {
    /* propagate error */
}
else {
    /* continue doing useful work */
}
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\ (cpython-main) (Doc) (c-api) iter.rst, line 57)

Unknown directive type "c:type".

```
.. c:type:: PySendResult
```

The enum value used to represent different results of :c:func:`PyIter_Send`.

```
.. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\ (cpython-main) (Doc) (c-api) iter.rst, line 64)

Unknown directive type "c:function".

```
.. c:function:: PySendResult PyIter_Send(PyObject *iter, PyObject *arg, PyObject **presult)
```

Sends the **arg** value into the iterator **iter**. Returns:

- ``PYGEN_RETURN`` if iterator returns. Return value is returned via **presult**.
- ``PYGEN_NEXT`` if iterator yields. Yielded value is returned via **presult**.
- ``PYGEN_ERROR`` if iterator has raised an exception. **presult** is set to ``NULL``.

```
.. versionadded:: 3.10
```