

# Objects for Type Hinting

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

Unknown directive type "highlight".

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

Various built-in types for type hinting are provided. Currently, two types exist -- `:ref:`GenericAlias <types-genericalias>`` and `:ref:`Union <types-union>``. Only `GenericAlias` is exposed to C.

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] typehints.rst, line 8); [backlink](#)**

Unknown interpreted text role "ref".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] typehints.rst, line 8); [backlink](#)**

Unknown interpreted text role "ref".

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] typehints.rst, line 12)**

Unknown directive type "c:function".

```
.. c:function:: PyObject* Py_GenericAlias(PyObject *origin, PyObject *args)
```

Create a `:ref:`GenericAlias <types-genericalias>`` object. Equivalent to calling the Python class `:class:`types.GenericAlias``. The `*origin*` and `*args*` arguments set the `__GenericAlias__`'s `__origin__` and `__args__` attributes respectively. `*origin*` should be a `:c:type:`PyTypeObject*``, and `*args*` can be a `:c:type:`PyTupleObject*`` or any `PyObject*`. If `*args*` passed is not a tuple, a 1-tuple is automatically constructed and `__args__` is set to `(args,)`. Minimal checking is done for the arguments, so the function will succeed even if `*origin*` is not a type. The `__GenericAlias__`'s `__parameters__` attribute is constructed lazily from `__args__`. On failure, an exception is raised and `NULL` is returned.

Here's an example of how to make an extension type generic::

```
...
static PyMethodDef my_obj_methods[] = {
    // Other methods.
    ...
    {"__class_getitem__", Py_GenericAlias, METH_O|METH_CLASS, "See PEP 585"}
    ...
}

.. seealso:: The data model method :meth:`__class_getitem__`.

.. versionadded:: 3.9
```

**System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main] [Doc] [c-api] typehints.rst, line 42)**

Unknown directive type "c:var".

```
.. c:var:: PyTypeObject Py_GenericAliasType
```

The C type of the object returned by `:c:func:`Py_GenericAlias``. Equivalent to `:class:`types.GenericAlias`` in Python.

```
.. versionadded:: 3.9
```