

```
static PyMethodDef module_methods[] = {
    { "func", (PyCFunction)module_func, METH_NOARGS, NULL },
    { NULL, NULL, 0, NULL }
};
```

## The Initialization Function

The last part of your extension module is the initialization function. This function is called by the Python interpreter when the module is loaded. It is required that the function be named **initModule**, where *Module* is the name of the module.

The initialization function needs to be exported from the library you are building. The Python headers define PyMODINIT\_FUNC to include the appropriate incantations for that to happen for the particular environment in which we're compiling. All you have to do is use it when defining the function.

Your C initialization function generally has the following overall structure:

```
PyMODINIT_FUNC initModule() {
    Py_InitModule3(func, module_methods, "docstring...");
}
```

Here is the description of *Py\_InitModule3* function:

- 146.**            **func:** This is the function to be exported.
- 147.**            **module\_methods:** This is the mapping table name defined above.
- 148.**            **docstring:** This is the comment you want to give in your extension.

Putting this all together looks like the following:

```
#include <Python.h>

static PyObject *module_func(PyObject *self, PyObject *args) {
    /* Do your stuff here. */
    Py_RETURN_NONE;
```