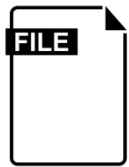# Python

Combining scripts and modules

Thanks to all contributors:

Alison Pamment, Sam Pepler, Ag Stephens, Stephen Pascoe, Kevin Marsh, Anabelle Guillory, Graham Parton, Esther Conway, Eduardo Damasio Da Costa, Wendy Garland, Alan Iwi, Matt Pritchard and Tommy Godfrey.

A simple python module/script

In Python you will often want to write a module where most of your code is held and then use a separate script to interact with it.

In this contrived example we have:

`greetings.py` (module)

`greeter.py` (script)

Centre for Environmental Data Analysis
SCIENCE AND TECHNOLOGY FACILITIES COUNCIL
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Atmospheric Science
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Earth Observation
NATURAL ENVIRONMENT RESEARCH COUNCIL

How will it work?

When written, the script will be called like this:

```
$ python greeter.py
Nobody to greet!


$ python greeter.py Greta
Hello Greta


$ python greeter.py Harpo Chico Zeppo
Hello Harpo
Hello Chico
Hello Zeppo
```

The "greetings.py" module

greetings.py

(module)

Holds the function that actual does something:

```
def greet(someone):
    print("Hello {0}".format(someone))
```

Centre for Environmental Data Analysis
SCIENCE AND TECHNOLOGY FACILITIES COUNCIL
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Atmospheric Science
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Earth Observation
NATURAL ENVIRONMENT RESEARCH COUNCIL

The "greeter.py" script

FILE

greeter.py
(script)

• defines the interaction between the "greetings.py" module and user input (from the command-line).

```python
import greetings
import sys


if len(sys.argv) == 1:
    print("Nobody to greet!")
else:
    for person in sys.argv[1:]:
        greetings.greet(person)
```
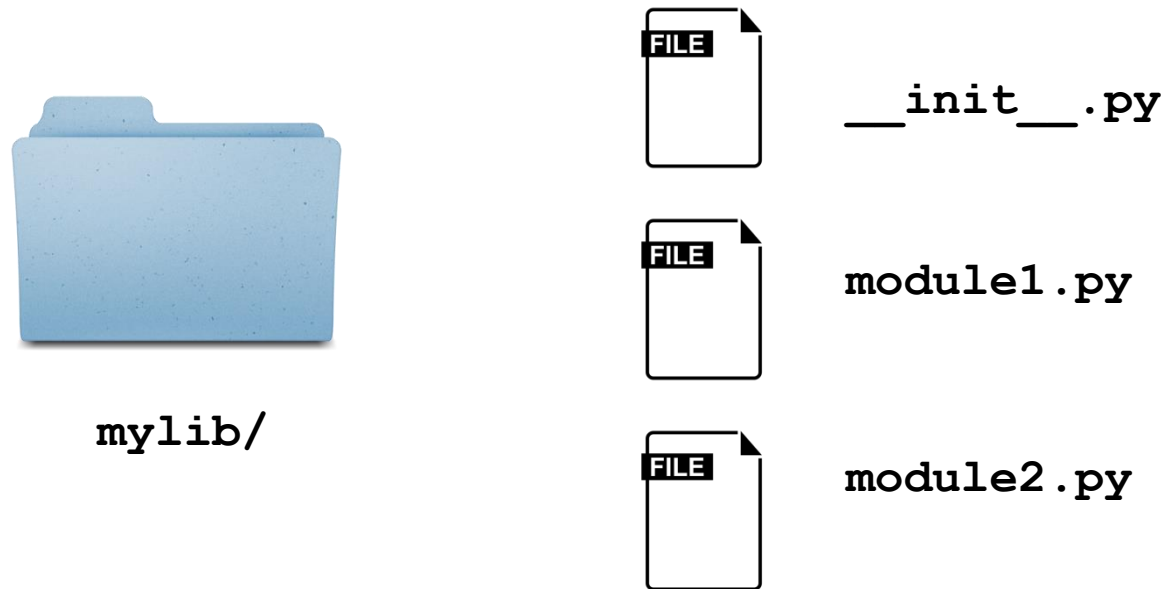
A python "package"

In Python you will often want to group a set of modules into a **package** or **library.**

On the file system a library might look like this:

mylib/

`__init__.py`

`module1.py`

`module2.py`

What does __init__.py do?

mylib/   **FILE** __init__.py

The "__init__.py" module is run when you import the name of the directory. It tells python that this directory is a Python *package*.

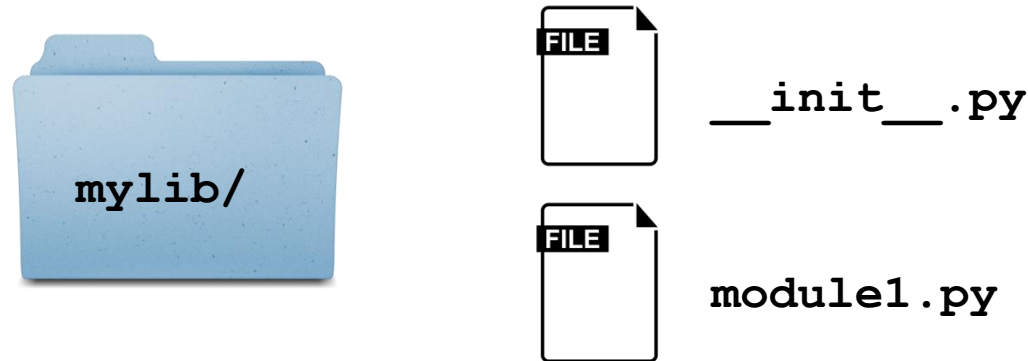In this case it is called "mylib" so you would type:

```
>>> import mylib    # runs content of mylib/__init__.py
```

If "__init__.py" contained the line "**print**(10)" you would see:

```
>>> import mylib
10
```

Importing a package module

mylib/

FILE __init__.py

FILE module1.py

The existence of the "__init__.py" module allows you to import modules within the package with:

```
>>> import mylib.module1
>>> mylib.module1.runSomething(1, 2, 3)
```