Data Science in Python (DSiP) - Programming Basics

Zime, Songbian PhD

September 24, 2023

Python Files and Modules

Structure of the Code File

- Code files end with "‘.py‘":

`myprogram.py`

- Every line is a Python statement (or part thereof). Comment lines start with #

Running Python Programs from the Command Line

Execute with the Python interpreter

```
$ python myprogram.py
```

Direct use on UNIX systems

- Define the path to the interpreter on the first line of the program as a comment:

```
#!/usr/bin/env python
```

- If setting the executable flag of the file, we can run the program directly in the shell:

```
$ myprogram.py
```

Python Modules

Overview

- **Modules** group Python functions
- *Modules are files* containing Python code (e.g., myprogram.py)
- Using Modules requires importing them first using the `import` statement
- **Scope:** Import a module to the current namespace, defined in its own namespace, or used as `module.function`

The Python Standard Library is a large collection of modules that provides *cross-platform* implementations of common facilities such as access to the operating system, file I/O, string management, network communication, and much more.

References

- The Python Language Reference: `http://docs.python.org/2/reference/index.html`
- The Python Standard Library: `http://docs.python.org/2/library/`

Module Import

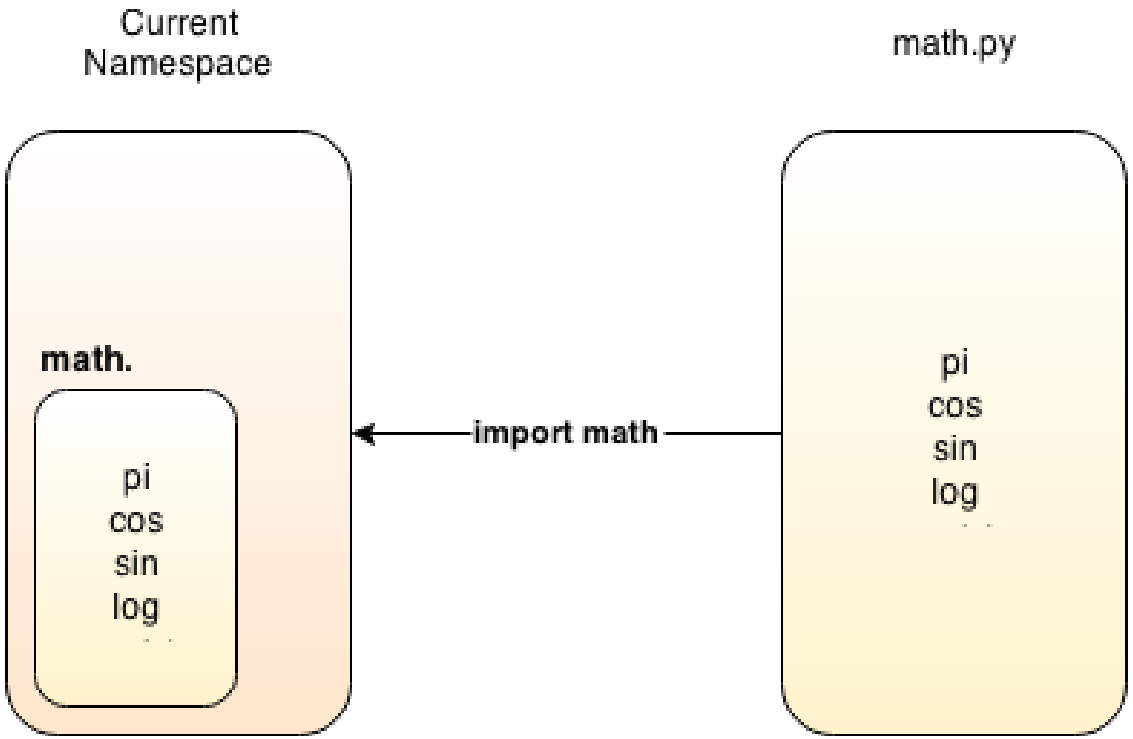
For example, to import the module `math`, which contains many standard mathematical functions, we can do:

```
import math

x = math.cos(2 * math.pi)

print(x)
 1.0
```

Python Namespaces



Note

Warning

Contrary to Java, a Python file can contain classes, variables, functions, and main code. The filename defines the module name and the namespace to access the functions.

Import a Module in Current Namespace

- Goal: avoid writing the prefix `module.` by importing all symbols (functions and variables) of one module into the current namespace.