

```
support.print_func("Zara")
```

When the above code is executed, it produces the following result:

```
Hello : Zara
```

A module is loaded only once, regardless of the number of times it is imported. This prevents the module execution from happening over and over again if multiple imports occur.

The *from...import* Statement

Python's *from* statement lets you import specific attributes from a module into the current namespace. The *from...import* has the following syntax:

```
from modname import name1[, name2[, ... nameN]]
```

For example, to import the function *fibonacci* from the module *fib*, use the following statement:

```
from fib import fibonacci
```

This statement does not import the entire module *fib* into the current namespace; it just introduces the item *fibonacci* from the module *fib* into the global symbol table of the importing module.

The *from...import ** Statement:

It is also possible to import all names from a module into the current namespace by using the following import statement:

```
from modname import *
```

This provides an easy way to import all the items from a module into the current namespace; however, this statement should be used sparingly.

Locating Modules:

When you import a module, the Python interpreter searches for the module in the following sequences:

- The current directory.